Coventry University

DOCTOR OF PHILOSOPHY

A new computational framework for the classification and function prediction of long non-coding RNAs

Deshpande, Sumukh

Award date:
2018

Awarding institution:
Coventry University

Link to publication

A new computational framework for the classification and function prediction of long non-coding RNAs

Sumukh Deshpande

A thesis submitted in partial fulfilment of the University's requirements for the Degree of Doctor of Philosophy

August 2018

Coventry University

Certificate of Ethical Approval

Applicant:

Sumukh Deshpande

Project Title:

A new computational framework for the classification and function prediction of long non-coding RNAs

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Low Risk

Date of approval:

30 June 2018

Project Reference Number:

P72741

ABSTRACT

Long non-coding RNAs (lncRNAs) are known to play a significant role in several biological processes. These RNAs possess sequence length greater than 200 base pairs (bp), and so are often misclassified as protein-coding genes. Most Coding Potential Computation (CPC) tools fail to accurately identify, classify and predict the biological functions of lncRNAs in plant genomes, due to previous research being limited to mammalian genomes.

In this thesis, an investigation and extraction of various sequence and codon-bias features for identification of lncRNA sequences has been carried out, to develop a new CPC Framework. For identification of essential features, the framework implements regularisation-based selection. A novel classification algorithm is implemented, which removes the dependency on experimental datasets and provides a coordinate-based solution for sub-classification of lncRNAs. For imputing the lncRNA functions, lncRNA-protein interactions have been first determined through co-expression of genes which were re-analysed by a sequence similarity-based approach for identification of novel interactions and prediction of lncRNA functions in the genome. This integrates a D3-based application for visualisation of lncRNA sequences and their associated functions in the genome.

Standard evaluation metrics such as accuracy, sensitivity, and specificity have been used for benchmarking the performance of the framework against leading CPC tools. Case study analyses were conducted with plant RNA-seq datasets for evaluating the effectiveness of the framework using a cross-validation approach. The tests show the framework can provide significant improvements on existing CPC models for plant genomes: 20-40% greater accuracy. Function prediction analysis demonstrates results are consistent with the experimentally-published findings.

ACKNOWLEDGEMENTS

My deepest and sincere appreciation goes to my Director of Studies Dr. James Shuttleworth whose tremendous support, commitments, suggestions, ideas, encouragement and mentorship helped shaped this work and moulded me. He contributed immensely to bring this work this far. I will forever remain grateful as I keep thanking my stars for having you with so much wealth of knowledge and experience as my Director of Studies.

My special thanks and appreciation also go to my project external supervisor Dr. Jianhua Yang who inspired me and this work in many ways. I am grateful for his support, suggestions, brilliant criticisms and corrections and for keeping me focused on the work.

I also wish to thank and appreciate my co-supervisors Dr. Sandy Taramonli and Dr. Matthew England whose early contributions helped point out a direction for the research. I also wish to extend my gratitude to Prof. Anne James whose contributions and observations especially during the PRPs contributed greatly to shape this work.

I owe a depth of gratitude to all the staff and research colleagues in Coventry University, words cannot express my gratitude for the role you all played at different stages of this work, I am deeply grateful.

My gratitude also goes to Coventry University for providing scholarships to pursue doctoral studies. I would like to thank administrative and technical staff members of the school who have been kind enough to advise and help in their respective roles.

Most importantly, I would like to thank my father Dr. S. D. Deshpande, my mother Dr. Sumedha Deshpande, my wife Mrs. Pooja Purohit Deshpande and my lovely daughter Swara Deshpande, for their love, patience, and understanding — they allowed me to spend most of the time on this thesis and for supporting me spiritually.

PUBLISHED ARTICLES

Journal Contributions

1. Deshpande, S., Yang, J., Shuttleworth, J., Taramonli, S., England, M. and James, A. (2018). 'LIFT: LncRNA Identification and Function-prediction Tool'. (*Submitted for review*).
2. Deshpande, S., James, A., Franklin, C. H., Leach, L. J. and Yang, J (2018). 'Identification of novel flowering genes using RNA-seq pipeline employing combinatorial approach in Arabidopsis thaliana time-series apical shoot meristem data', in *International Journal of Bioinformatics Research and Applications, Inderscience Publishers* (*in press*).

Conference Contributions

1. Deshpande, S., James, A., Franklin, C. H., Leach, L. J., Taramonli, S. and Yang, J. (2017) 'An RNA-seq Bioinformatics Pipeline for Data Processing of Arabidopsis Thaliana Datasets', in *Proceedings of the International Conference on Bioinformatics Research and Applications 2017*. New York, NY, USA: ACM (ICBRA 2017), pp. 1–8. doi: 10.1145/3175587.3175592.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

5mC	5-methyl cytosine
A3	Alternative 3-prime splice site
A5	Alternative 5-prime splice site
ACC	Accuracy
ANT	Adjoining Nucleotide Triplet
ANT	Antisense lncRNA
AOE	Antisense Overlapping Exonic
AOI	Antisense Overlapping Intronic
ATH	Arabidopsis Thaliana
ATSS	Alternative Transcription Start Site
ATTS	Alternative Transcription Termination Site
AUC	Area Under the ROC Curve
BAM	Binary Alignment Map
BC	Betweenness Centrality
BCV	Biological Coefficient of Variation
BDLDA	Block Diagonal Linear Discriminant Analysis
BMRF	Bayesian Markov Random Fields
BN	Bayesian Network
BNA	Brassica Napus
BOL	Brassica Oleracea
bp	base pairs
BP	Biological Process
BRAD	Brassica Database
BRF	Balanced Random Forest
BSR	Backward Stepwise Regression
CART	Classification and Regression Trees
CC	Cellular Component
CC	Closeness Centrality
CFS	Correlation-based Feature Selection
ChIP-Seq	Chromatin Immunoprecipitation Sequencing
CNCI	Coding-Non-Coding Index
COLDAIR	COLD ASSISTED INTRONIC NONCODING RNA

COOLAIR	COLD ASSISTED LONG ANTISENSE INTRAGENIC RNAs
CPAT	Coding Potential Assessment Tool
CPC	Coding Potential Calculator
CRF	Conditional Random Fields
CSF	Codon Substitution Frequencies
CSV	Comma Separated Values
CUB	Codon Usage Bias
CV	Cross Validation
DAG	Directed Acyclic Graph
DEG	Differentially Expressed Genes
DEG	Differentially Expressed Gene
DEMC	Differential Evolution Markov Chain
DFT	Discrete Fourier Transform
DGE	Differential Gene Expression
DNA	DeoxyriboNucleic Acid
dsRNA	double stranded RNA
EM	Expectation-Maximization
eRNA	enhancer ncRNA
ESI	Exon Skipping/Inclusion
EST	Expressed Sequence Tags
FDR	False Discovery Rate
FG	Feature Groups
FGN	Functional Grouped Network
FPF	False Positive Fraction
FPKM	Fragments of Per Kilobase of transcript per Million
FSR	Forward Stepwise Regression
GA-SVM	Genetic Algorithms with SVM
GBA	Greedy Backward Algorithm
GFS	Greedy Forward Selection
GLM	Generalized Linear Model
GluBP	Glucosinolate Biosynthetic Process
GluMP	Glucosinolate Metabolic Process
GlyBP	Glycosinolate Biosynthetic Process

GlyMP	Glycosinolate Metabolic Process
GO	Gene Ontology
GR	Gain Ratio
GS	Gibbs-Sampling
GTF	Gene Transfer Format
GUI	Graphical User Interface
HGLDA	Hyper Geometric Distribution of LncRNA-Disease Association
HMM	Hidden Markov Model
HS	Homo Sapiens
HSP	High-scoring Segment Pair
IDD	Intervertebral Disc Degeneration
IG	Information Gain
IGB	Integrated Genome Browser
INT	Intergenic lncRNA
iRF	iterative Random Forest
JPE	Joint Parameter Estimation
kB	kilo Basepair
KEGG	Kyoto Encylcopedia of Gene and Genome
KLR	Kernel Logistic Regression
LASSO	Least Absolute Shrinkage and Selection Operator
LFCs	Logarithmic Fold Change sizes
LincRNA	Long intergenic non-coding RNA
LiRF-FS	LASSO-iterative Random Forest-Feature Selection
lncRNA	long non-coding RNA
LPBNI	lncRNA-protein bipartite network inference
LPCS	LncRNA-Protein Co-expression Similarity
LPI	LncRNA-Protein Interaction
LPIHN	lncRNA-protein interaction prediction based on Heterogenous Network Model
LR	Logistic Regression
MCC	Matthews Correlation Coefficient
MCMC	Markov Chain Monte Carlo
MDA	Multimodal Deep Autoencoders

MeDIPSeq	Methylated DNA ImmunoPrecipitation Sequencing
MEE	Mutually Exclusive Exon
MF	Molecular Function
MI	Mutual Information
MID	Mutual Information Difference
miRNA	micro RNA
ML	Machine Learning
MLCDS	Most-Like Coding Sequences
MLE	Maximum Likelihood Estimate
MM	Mus Musculus
MRF	Markov Random Field
mRMR	minimum Redundancy Maximum Relevance
mRNA	messenger RNA
mtDNA	mitochondrial DNA
NA	Nucleic Acid
NAT	Natural Antisense Transcript
ncRNA	non-coding RNA
NGS	Next-Generation Sequencing
NPV	Negative Predictive Value
NRLMF	Neighbourhood Regularised Logistic Matrix Factorisation
OLS	Ordinary Least Squares
ORF	Open Reading Frame
OS	Oryza Sativa
PARs	promoter-associated RNA
PB	Protein Block
PBC	Position-Based Classification
PCC	Pearson Correlation Coefficient
PCR	Polymerase Chain Reaction
PDB	Protein Data Bank
piRNA	piwi-interacting RNA
PLEK	Predictor of long noncoding RNAs and messenger RNAs based on an improved k-mer scheme
PLF	Pseudo-Likelihood Function

PLS	Partial Least Squares
PPI	Protein-Protein Interactions
PPMI	Positive Pointwise Mutual Information
PPV	Positive Predictive Value
pre-mRNA	premature-mRNA
PRES	Precision
PT	Processed Transcript
PT-ANT	processed transcript antisense
PT-AOE	processed transcript antisense overlapping exonic
PT-AOI	processed transcript antisense overlapping intronic
PT-BDP	processed transcript bidirectional promoter
PT-INT	processed transcript intergenic
PT-SOE	processed transcript sense overlapping exonic
PT-SOI	processed transcript sense overlapping intronic
RCB	Relative Codon Bias
RF	Random Forest
RFE	Recursive Feature Elimination
RGE	Relative Gene Expression
RIT	Random Intersection Trees
RNA-seq	RNA Sequencing
ROC	Receiver Operating Characteristics
RPKM	Reads Per Kilobase of transcript per Million
RR	Ridge regression
rRNA	ribosome-associated RNA
RRS	Ribosome Release Score
RSCU	Relative Synonymous Codon Usage
RWR	Random Walk with Restart
SAM	Shoot Apical Meristem
SAM	Sequence Alignment Map
SCBP	Sulfur Compound Biosynthetic Process
SCMP	Sulfur Compound Metabolic Process
SCUO	Synonymous Codon Usage Order
SENS	Sensitivity

SFS	Sequential Forward Selection
SGS	Second Generation Sequencing
siRNA	small interfering RNA
SL	Solanum Lycopersicum
SMRT	Single Molecule Real-Time
snoRNA	small nucleolar RNA
SNPs	Single Nucleotide Polymorphisms
snRNA	small nuclear RNA
SOE	Sense Overlapping Exonic
SOI	Sense Overlapping Intronic
SPEC	Specificity
SSM	Sequence Similarity Match
ST	Solanum Tuberosum
SVM	Support Vector Machine
SVM-RFE	SVM-Recursive Feature Elimination
TEC	To be Experimentally Confirmed
TEC-ANT	TEC antisense
TEC-AOE	TEC antisense overlapping exonic
TEC-AOI	TEC antisense overlapping intronic
TEC-BDP	TEC bidirectional promoter
TEC-INT	TEC intergenic
TEC-SOE	TEC sense overlapping exonic
TEC-SOI	TEC sense overlapping intronic
TPF	True Positive Fraction
TPM	Transcripts Per Million
tRNA	transfer RNA
TS	Targeted Sequencing
UTR	untranslated region
VCF	Variant Calling Format
WES	Whole Exome Sequencing
WGCNA	Weighted Gene Co-expression Network Analysis
WGS	Whole-Genome Sequencing
ZM	Zea Mays

CHAPTER 1: INTRODUCTION

1.1 Introduction

Genomics is a field which focusses on studying the genome of organisms. The genome is made up of DNA sequence which codes for protein structures required for normal functioning of cells and tissues. Apart from these protein-coding sequences, DNA also consists of several other types of sequences which do not code for any protein structures but play a pivotal role in the gene regulation process. These non-coding RNA sequences have also been found to be associated with several cancer types (Huarte, 2015). They affect the proliferation, migration, survival, genomic stability and in the regulation of cellular homeostasis. Current advancements in genomics have enabled sequencing of coding and non-coding transcripts. To identify their potential functions in various biological processes, it is essential to accurately identify these sequences in the genome. Identification of these non-coding RNAs can be performed through computational algorithms which can predict these transcripts with certain degree of accuracy, however accurate computationally identification of these non-coding RNA sequences and their functions in plants is still an open problem. The research presented in this thesis addresses the gaps in computational genomics with application to RNA sequence analysis in plant datasets.

This chapter introduces the background to the problem and the motivation for undertaking the research. It then identifies the research questions recognized from the study of current research developments in identification, classification and function prediction of long non-coding RNA (lncRNA) sequences. This is followed by a description of the aims and objectives of the thesis. A short description of methods adopted in order to achieve the primary research goals is also provided, followed by a summary of the research contributions made in the project. This chapter also clarifies the scope of the research undertaken. A summary of the work is presented which is followed by the description of the thesis structure.

1.2 Background

DeoxyriboNucleic Acid (DNA) is the hereditary substance found in the nucleus of a cell in all cellular organisms. However, small amounts of DNA can also be found in the mitochondria, otherwise known as mitochondrial DNA (mtDNA). DNA is primarily made up of four chemical bases: Adenine (A), Cytosine (C), Guanine (G) and Thymine (T). These chemical bases pair up with each other to form base pairs (bp). For example, A pairs with T and G pairs with C

(Figure 1.1). Each base is attached to a phosphate and sugar molecule to form a nucleotide. These nucleotides are organized in two long DNA strands forming a double helix.

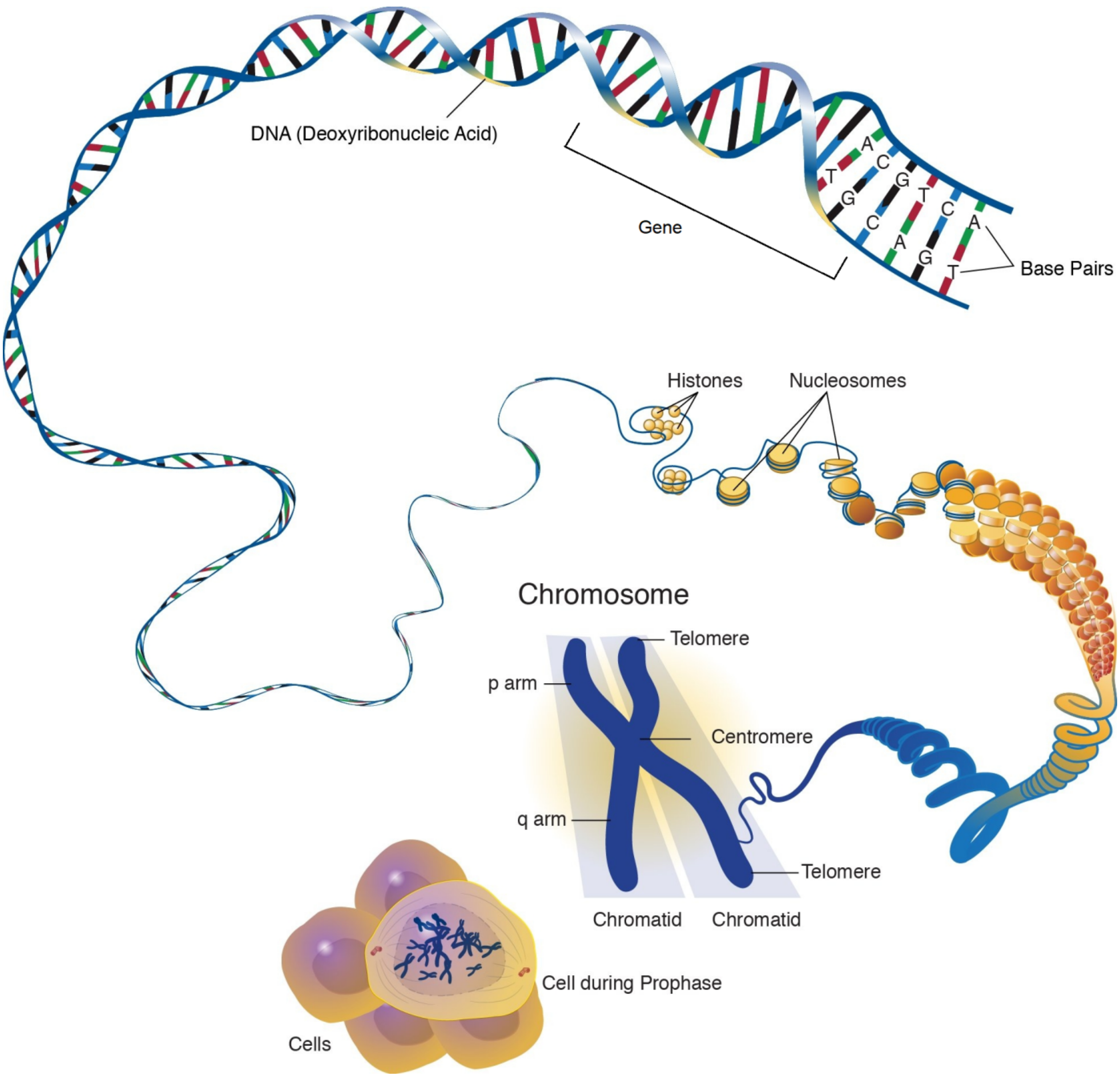


Figure 1.1: Graphical illustration of Chromosome, DNA and Gene in the nucleus of a cell. After (Genome.gov, 2014).

The DNA encodes a functional unit of heredity called gene (Figure 1.1) which is made up of DNA, acting as instructions for making molecules called proteins. The human genomic DNA is estimated to contain 25,000 – 30,000 genes encoding several different proteins required for cellular differentiation, repair and maintenance of a cell. These functions as antibodies binding to foreign particles such as bacteria and viruses, enzymes involved in undertaking several chemical reactions, messenger proteins involved in transmission of signals for coordinating biological processes, as structural components providing support and structure for the cells, and as transporters responsible for conduction of ions and small molecules across the cell. These protein structures are formed by the processes involving transcription and translation (Figure 1.2). Transcription involves transfer of information from DNA to messenger RNA (mRNA). The DNA serves as template for complementary base pairing through RNA polymerase enzyme which catalyses pre-mRNA molecule. The pre-mRNA undergoes processing inside the nucleus to form mature mRNA. The resulting mRNA then translocate out of the nucleus into the cytoplasm of the cell where it undergoes translation. The translation process involves conversion of mRNA encoded information into a polypeptide chain. The polypeptide chain undergoes folding which forms into a compact molecular structure called protein. The protein is transported to various locations in the cell for performing cellular function.

Figure 1.2: Expression of a gene through the processes of transcription and translation. (Clancy and Brown, 2008).

The genome of an organism comprises of a complete DNA sequence containing all the information needed for building and maintaining that organism. A human genome is made up of 3 billion bp which are distributed across 24 chromosomes residing in the nucleus of each cell. The genome encodes numerous protein-coding genes which are required for the normal functioning of cells and tissues in an organism. Genome sequencing has revealed that 98% of the genome which is transcribed into RNA has little or no protein-coding ability (Xie *et al.*, 2014a). This class of RNA is called non-coding RNAs (ncRNAs). Additionally, the majority of ncRNAs are expressed along with protein-coding genes thus displaying accurate localisation within the cell which mainly suggests their potential regulatory functions.

ncRNAs are functionally categorised into two types: housekeeping and regulatory ncRNAs. Housekeeping ncRNAs include transfer RNA (tRNA), ribosome-associated RNA (rRNA), small nuclear RNA (snRNA) and small nucleolar RNA (snoRNA) playing critical roles in cellular and processes and are involved in maintenance of cellular functions. While regulatory ncRNAs include micro RNA (miRNA), small interfering RNA (siRNA), long non-coding RNA (lncRNA), enhancer ncRNA (eRNA), promoter-associated RNA (PARs) and piwi-interacting RNA (piRNA).

Unlike other ncRNAs and protein-coding genes which are highly conserved across the genome, lncRNAs exhibit lower sequence conservation. The majority of lncRNAs possess sequence length greater than 200 bp whereas other classes of ncRNAs commonly occur with sequence lengths less than 200 bp. Due to their longer lengths, they are often misclassified as protein-coding genes. In contrast to the protein-coding sequences having experimentally-verified functions, only a small fraction of lncRNA sequences have been known to be associated with molecular function. However, through experimental studies, lncRNAs have been implicated in many crucial cellular and biological processes such as genomic imprinting, X-chromosome inactivation, centromere and telomere organisation, nuclear trafficking and sub-cellular organisation. Additionally, changes in the expression of lncRNAs have been reported in multiple diseases which makes them an important therapeutic target. The function of lncRNA in the genome generally depends on four factors: (1) Sequence composition, (2) Pattern of expression, (3) Interactions with protein-coding genes, and (4) Genomic localisation. To determine each of the factors, several methods have been evolved.

Since the 1970s, several technologies have emerged for sequencing DNA, including Sanger sequencing and Maxam-Gilbert sequencing, the first-generation sequencing technologies which initiated the sequencing of DNA through conventional methods (Heather and Chain, 2016; Mardis, 2017). However, due to the speed of analysis and cost involved, a second generation of sequencers emerged which included Illumina, Ion Torrent, Roche/454, ABI/SOLiD sequencers (Heather and Chain, 2016; Mardis, 2017).

The emergence of powerful machines revolutionised DNA sequencing and analysis. However, second generation sequencers involved a Polymerase Chain Reaction (PCR) amplification step which was expensive and took longer time. To remedy the problems caused by Second Generation Sequencing (SGS) machines, a third generation of sequencers often referred to as Next-Generation Sequencing (NGS) technology evolved which provided significant improvement over speed and cost (Heather and Chain, 2016; Mardis, 2017). This opened gateways to identification and analysis of various DNA sequences through bioinformatics approaches such as sequencing of the whole genome using Whole-Genome Sequencing (WGS), sequencing of protein-coding genes using Whole Exome Sequencing (WES), sequencing of RNAs using RNA sequencing (RNA-seq), identification protein-DNA interaction sites using Chromatin Immunoprecipitation sequencing (ChIP-Seq), identification of DNA methylation using MethylSeq (Buermans and den Dunnen, 2014).

NGS sequencing technologies such as RNA-seq not only allowed identification of protein-coding sequences but also allowed sequencing of lncRNA sequences. Through RNA sequencing, several lncRNAs have been catalogued in public databases such as GENCODE and NONCODE which have now been developed for storage of lncRNAs and protein-coding sequences (Harrow *et al.*, 2012; Zhao *et al.*, 2016b). Currently, a large number of the lncRNAs identified through RNA-seq and predicted through computational approaches have been reported in mammalian species. However, current knowledge on lncRNAs and their biological function in plants is still limited. Even though NGS techniques such as RNA-seq is actively used for identification and revelation of novel lncRNAs, accurately identifying the lncRNAs and determining their functions in model/non-model plant organisms is an area of open research.

Through computational and statistical approaches, a number of tools have been developed which provide identification of novel lncRNAs. However, most of them fail in accurately determining lncRNAs in plants. These tools are primarily based on machine learning methods which involves extraction and statistical analysis of various features from the sequences. Current tools and methods employed for computational prediction of lncRNA transcripts can be broadly classified into alignment-free and alignment-based methods. Alignment-based tools predicts the lncRNAs based on their alignment with protein-coding sequences deposited in the web databases. The transcript sequences are scored, and the prediction is obtained based on the degree of match with the protein-coding sequences. Alignment-free methods on the other hand derive the sequence characteristics features predominantly based on features such as sequence motifs (repetitive sequence patterns) and sequence length. Currently available methods do not provide information about the significance of each feature from the identification process. Determining feature importance, not only helps to accurately determine lncRNAs, but also highlights the crucial role of the features.

Additionally, current methods developed for identification of lncRNAs are focused on mammalian species which means that similar computational models fail to provide a reasonable accuracy for the identification of plant lncRNA sequences. These include non-availability of lncRNA sequences, known lncRNA-protein interaction data, protein-protein interaction data, lncRNA genomic annotation data for plant species. Since lncRNAs exhibit poor sequence conservation across several species unlike protein-coding genes which show high level of sequence conservation, identification of lncRNAs becomes even more challenging in plants.

One of the major bottlenecks is the accurate determination of lncRNA sub-classes (Ma, Bajic and Zhang, 2013; St.Laurent, Wahlestedt and Kapranov, 2015) among the overabundance of sequences. lncRNAs are generally categorised into four types depending upon their position in the genome: (1) Sense and Antisense lncRNAs, (2) Intronic or Exonic lncRNAs, (3) Intergenic lncRNAs and (4) Bidirectional lncRNAs. Currently available computational tools and databases provide limited resources of lncRNA sub-classes in plants, due to which the majority of the lncRNAs remains unclassified. Also, prediction of these sub-classes using machine learning methods often leads to poor accuracy due to data limitations. Therefore, a smarter approach is required, which removes the dependency on machine learning classifiers and also provides detailed and comprehensive classification of lncRNA sequences. Classifications of lncRNA sequences are of fundamental importance for lncRNA studies as it is helpful for formulation of new hypothesis based on different lncRNA features and exploration of functional mechanisms of lncRNAs.

Through experimental approaches, lncRNAs have been known to regulate several biological processes through interaction with protein-coding genes (Dykes and Emanueli, 2017). However, determination of functions of lncRNAs still remains challenging. Currently developed lncRNA function prediction tools rely on known lncRNA-protein interactions which are commonly available for mammalian species due to which the application of function prediction becomes limited and cannot be applied to other non-mammalian species (Hou *et al.*, 2016;

Wang *et al.*, 2016a; Zheng *et al.*, 2016; Zhou *et al.*, 2018). Furthermore, present research on lncRNA function imputation is predominantly based on co-expression of lncRNA and protein-coding genes and disease association in humans (Guo *et al.*, 2015; Hao *et al.*, 2015; Wang *et al.*, 2016b; Cagirici, Alptekin and Budak, 2017; Zhu *et al.*, 2017). Currently, elucidation of lncRNA function is still in its infancy, due to limitation of known lncRNA-protein interaction data, potential functions becomes difficult to impute.

1.3 Motivation for undertaking this work

LncRNA plays a significant role in the regulation of several biological processes but accurate determination of lncRNAs and their sub-classes remains a challenge (Ma, Bajic and Zhang, 2013). Current computational tools and methods developed for identification of lncRNAs are tailored for determining sequences derived from GENCODE, Refseq and NONCODE databases but fail to identify lncRNAs obtained from RNA-seq data (Wang *et al.*, 2013; Li, Zhang and Zhou, 2014; Zhao *et al.*, 2016a). A crucial step in lncRNA identification process requires extraction of relevant sequence features. These features are then used by Machine Learning (ML) methods such as Random Forest (RF), Support Vector Machines (SVM) and Logistic Regression (LR) for classification. Most of the currently developed ML-based methods derive features for classification based on sequence alignment-free and sequence alignment-based approaches. Usage of the alignment-based approach for classification often requires significant computational resources such as processing power for sequence alignment process and storage of large number of alignment data due to which usage of such tools becomes computationally impractical and therefore limits their usage. Whereas alignment-free methods extract the features which depends on relative oligonucleotide frequencies or *k*-mers. *K*-mer is a small DNA substring of length *k*. Given a DNA string of length L, there are L – *k* + 1 possible *k*-mers for a given DNA substring. *K*-mer dependent alignment-free methods require longer computation times and therefore becomes unsuitable. These tools also tend to generalise the features for different species such as mammals and plants due to which important information regarding potential selection of synonymous codons remain hidden and cannot be identified in individual species.

From various experimental studies (Harrow *et al.*, 2012; St.Laurent, Wahlestedt and Kapranov, 2015), lncRNAs have been known to belong to discrete categories based on their genomic position. Identification of various lncRNA sub-classes provides valuable insights into their sequence, structure, function and possible interactions with partner RNA sequences. Current methods and tools attempting to identify lncRNAs fail to classify the sequences. Additionally, currently available data on lncRNA sub-classes in public databases only covers mammalian genomic sequences. Therefore, ML-based tools developed for identification of these sub-classes are biased towards identification in mammalian species and a plethora of lncRNA sequences in plants are still waiting to be classified.

Experimental results show lncRNAs are often subjected to multiple regulatory and processing steps which are coordinated through interactions with DNA and RNA-binding proteins (RBPs) (Moore, 2005). These interactions often regulate nuclear export of mRNA to the cytoplasmic region, and maturation, providing stability and translation of mRNA to protein structures. Therefore, identification of lncRNA-protein interaction becomes crucial for understanding their function; however current tools for function prediction are particularly designed for predicting functions in mammalian species.

A wide range of studies have been conducted on optimisation of ML-based approaches for selection of optimal features but are commonly limited to application on microarray data. Therefore, considering the gaps in the literature, the main goal of this research is to address these limitations, by developing an approach for identification, classification and prediction of functions of lncRNA sequences. The proposed approach not only identifies the sequences but also determines essential features through a feature selection approach which provide insights into their sequence characteristics. The approach also implements a classification algorithm which removes the dependency from experimental datasets and therefore provides a position-based classification approach widely applicable to multiple plant and mammalian species. For function prediction of lncRNAs, the approach relies on the identification of novel lncRNA-protein interactions based on sequence similarity between plant and mammalian transcript sequences. Functions are determined based on the Bayesian inference from the lncRNA-protein regulatory network. This unified approach of LPI determination and probabilistic computation of lncRNA functions provide a reliable computational model for plant species.

1.4 Research questions

An in-depth review of the current state-of-the-art presents the following research questions:

1. Is it possible to computationally predict the molecular/regulatory functions of lncRNAs based on coexpression of lncRNAs and protein-coding genes?
2. Can the lncRNA and protein interactions be predicted based on known lncRNA-protein interactions from humans?

3. How can we improve the prediction accuracy for distinguishing lncRNA sequences from protein-coding sequences as well as identify various sub-classes of lncRNAs based on the genomic coordinates?
4. Can we observe an improvement in the prediction accuracy of lncRNA identification using sequence-based, ORF-based, and codon-biased features?

1.5 Aims and objectives

This proposed research aims to improve the currently developed computational approaches for lncRNA classification and function prediction in plant species.

From the above aim, the following research objectives emerge:

- Develop a computational workflow for identification of coding and non-coding regions in the DNA from reference and RNA-seq datasets.
- Derive and extract sequence-based features from the RNA-seq data using coding potential measures established through literature review.
- Develop an optimisation method by integrating a regression-based feature selection method with an iterative Random Forest (iRF) classifier for identifying and extracting optimal features in species-specific datasets.
- Classification of lncRNAs using a LASSO-iterative Random Forest-Feature Selection (*LiRF-FS*) approach for obtaining optimal features derived from Refseq and GENCODE databases in plants and mammalian species.
- Develop a pipeline for identification of lncRNAs from RNA-seq data using species-specific feature subset.
- Annotate lncRNAs into various sub-classes based on their genomic location.
- Benchmark performance of the model against currently developed lncRNA identification methods.
- Prediction of lncRNA and protein interactions in plant species.
- Predict functions of lncRNAs based on lncRNA-protein inteactions and protein-protein interactions.
- Visualisation of diverse lncRNA sequence transcripts annotated with function information.

1.6 Research methodology

The initial stages of this work were focused on methods to identify the differentially expressed (DE) mRNAs from *Arabidopsis thaliana* time-series RNA-seq datasets. Subsequent part of this work involved development of the framework and demonstration on plant time-series datasets for performance evaluation of classification and function prediction methods. The following steps were implemented to achieve the goals enlisted in the thesis:

a) Identification of differentially expressed genes (DEGs)

We obtained the RNA-seq data of the *A. thaliana* apical-shoot dataset originally deposited by Klepikova et al. (A. V. Klepikova *et al.*, 2015) and constructed a customized pipeline which involved several data pre-processing and post-processing steps for identification of DEGs. Pre-processing steps involved include format conversion, quality checks before and after data cleaning, reference alignment, transcript identification and quantification, merging quantified transcripts from control and cases samples and identification of DEGs using multiple methods. Post-processing steps included identifying DEGs by intersection of results from several different approaches, gene ontology enrichment analysis, pathway analysis, protein-protein interaction network analysis and alternative splicing analysis.

b) Datasets

For testing and implementation of the computational model, FASTA sequences of protein-coding and long non-coding sequences were obtained from reference Refseq and GENCODE databases as well as from RNA-seq datasets. FASTA sequences of eight plant species were derived from the Refseq database whereas FASTA transcript sequences of two mammalian species (humans and mouse) were obtained from the GENCODE database. We obtained time-series datasets of *A. thaliana* and *Z. mays* species from the NCBI SRA database to implement the model already tested and benchmarked on reference datasets.

c) Data preparation

For feature extraction and classification of lncRNAs, lncRNA FASTA sequences extracted from reference databases were filtered using a cutoff value ≥ 200bp. An equal number of protein-coding sequences matching number of filtered lncRNA sequences were extracted to create a balanced dataset.

d) Feature generation and extraction

To extract several different features from the FASTA sequences for classification analysis, scripts for extraction of features were constructed in Javascript. 7 sequence-based and 66 codon-biased based features were extracted from protein-coding and lncRNA FASTA sequences. A feature matrix was constructed with 73 features which was normalized by scaling the values in all the columns between 0 and 1. Using normalised feature matrix, training and test sets were generated by dividing the feature set into 70% training set and 30% test set which was applied similarly to all species.

e) lncRNA identification using all features

Once the data is normalized and separated into training and test sets, the Random Forest classifier from Python module “scikit-learn” (Pedregosa and Varoquaux, 2011) and the iRF classifier (Basu *et al.*, 2018) were applied for identifying the lncRNA sequences in the test set, as well as to test the prediction accuracy of the lncRNA sequences using all features.

f) Feature selection

Feature selection was to extract relevant features using the LASSO regression method, which was combined with the iRF classifier to build an integrative approach for finding the optimal feature set in order to produce a higher accuracy. Several feature selection methods were tested against the LASSO method to benchmark its performance.

g) Annotation of lncRNA sequences based on their genomic location

Since lncRNAs can be found in several different regions in the genome, their genomic location determines their sub-type which can be categorised into sense, antisense, bidirectional and intergenic. Using FASTA sequences of lncRNAs, transcript sequences can be classified based on unique mapping algorithm.

h) Identification of lncRNA-protein interaction pairs

To identify potential lncRNA and protein interaction pairs, sequence similarity between lncRNA-lncRNA and protein-protein FASTA sequences was performed. FASTA sequences of plants were matched against FASTA sequences of human species having confirmed interaction as reported in NPInter database. Using NRLMF approach, sequences similarity matrices and adjacency matrix were used for obtaining novel interactions between lncRNA and proteins of plant sequences. LncRNA-protein interaction pairs were retained having Pearson Correlation Coefficient (PCC) ≥ -0.5.

i) Identification of protein-protein interaction pairs

To identify the protein-protein interaction pairs, protein-protein interaction data was retrieved from the STRING database (Szklarczyk *et al.*, 2015). Furthermore, mRNA associated Gene Ontology (GO) terms were also extracted.

j) Functional prediction of lncRNAs

Using lncRNA-protein interaction pairs, protein-protein interaction pairs and protein-GOterm pairs, function prediction of lncRNAs was performed using BMRF method. Potential lncRNAs having probability ≥ 0.8 were retained and were annotated with molecular function using GOterms.

k) Construction of visualisation framework

By analyzing the results obtained from the above methods, a visualisation report was constructed using D3.js library which provides a graphical interactive interface to the results obtained from lncRNA identification, lncRNA sub-type annotation and lncRNA function prediction.

1.7 Research scope

The proposed computational method can be widely applied for the identification of lncRNA sequences and the selection of the optimal features in plant RNA-seq datasets. The approach depends on data processing using bioinformatics and statistical tools. It uses multiple sequence features for classification which can be widely used for feature extraction in multiple species. However, the method does not restrict the application of these features in lncRNA identification. The feature selection approach developed can be used for identification of essential features from FASTA-based datasets. The algorithm developed for sub-classification of lncRNAs relies on the availability of the FASTA sequences and the coordinates of lncRNAs and protein-coding transcripts. Therefore, the approach is purely sequence and position-based and cannot be applied on data with missing coordinates or sequences.

The computational approach developed has been tested on two plant time-series RNA-seq datasets for identifying known lncRNA sequences and predicting functions based on time-series expression. For determining lncRNA-protein interactions, the approach relies on the availability of FASTA sequences for computing sequence similarities based on known and confirmed experimental lncRNA-protein interactions. The scope of the project is to provide a computational method for identification, annotation and functional prediction of lncRNAs in plants based on known lncRNA and mRNA data available from web-based genomic databases. Genome-wide exploration and function prediction of lncRNAs across several plant species is beyond the scope of this work. The proposed computational method can be widely applicable on several plant species for lncRNA prediction and function prediction.

This research only focusses on partial and full-length lncRNA and protein-coding transcript sequences. It filters out the rest of the RNA sequences as the scope of the project is limited to the analysis of lncRNA and protein-coding sequences only. Each component of the computational framework can be individually applied on reference and RNA-seq datasets.

1.8 Research contributions

The contributions of this thesis are briefly outlined below:

1. The thesis has adopted a novel approach for identification of lncRNAs which uses an ensemble of 73 carefully selected features that not only includes sequence-based features but also takes advantage of the codon-biased features to increase discriminative power.
2. The thesis has implemented LASSO-based feature selection in combination with the iRF classification for the selection of the optimal features from the reference datasets of plants and mammalian species with higher prediction accuracy and Area Under the ROC Curve (AUC) scores. This approach selects optimal features based on the training and validation datasets, which can be widely implemented on test set data. It not only provides an optimal and informative set of features but also delivers a list of the higher-order feature combinations which can be used to confirm the results obtained though *LiRF-FS* implementation. Implementation of *LiRF-FS* approach with codon-biased features promotes elucidation of potential regulatory motifs or codons which provide insights into distribution of codons in mRNA and lncRNA transcripts.
3. The development of the coordinate-based mapping algorithm for sub-classification of lncRNAs removes dependency over machine learning methods for prediction.
4. Demonstration of function prediction of lncRNA sequences in plant species through combinatorial approach utilizing NRLMF-derived lncRNA-protein interactions and determination of lncRNA functions using probabilistic Bayesian approach to Markov Random Fields. This provides accurate function predictions of unannotated lncRNAs using regulatory network and gene ontology data.

5. Plant-specific lncRNA prediction tool provides a useful resource for understanding lncRNA biology in plants.
6. The developed computational methods provide valuable functional and mechanistic insight into lncRNAs which are crucial for informing subsequent functional studies.
7. The lncRNA-protein interactions uncover relationships between lncRNA and protein in model and non-model plant species which help in determining potential functions of lncRNAs.
8. Implementation of lncRNA prediction, feature selection using LASSO and iRF, lncRNA sub-classification, lncRNA-protein interaction prediction and lncRNA function annotation as a computational framework will provide a useful bioinformatics resource for biomedical research studies.
9. Implementation of *LiRF-FS* approach with codon-biased features promote elucidation of potential regulatory motifs or codons which provide insights into distribution of codons in mRNA and lncRNA transcripts.

1.9 Alternative splicing and translation processes

Once the transcription is completed (as described in Section 1.2), the pre-mRNA or premature-mRNA sequences undergo alternative splicing before translation into protein structure. The pre-mRNA sequence is composed of introns and exons. Conversion of the pre-mRNA to mature-mRNA or mRNA transcript sequence requires removal of introns and ligation of exons which codes for a gene. A gene can be coded by one or more transcript sequences which ultimately depend on the selection of exons. The process in which the preferred exons are selected, and certain exons are skipped is called alternative splicing (Figure 1.3).

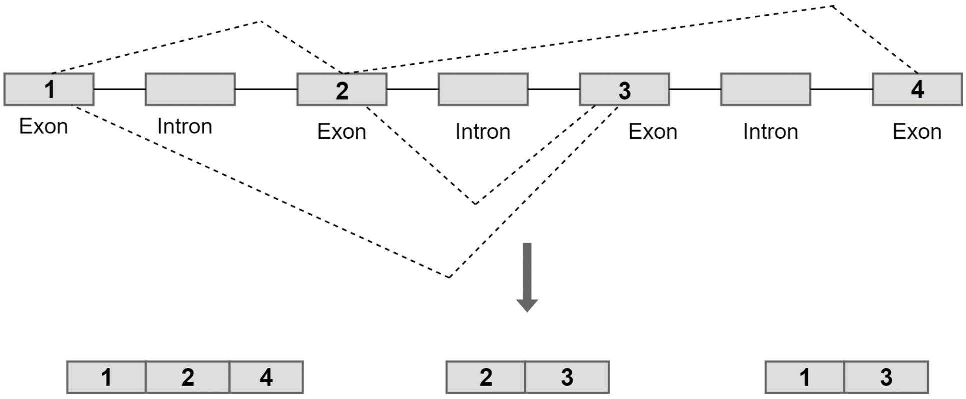


Figure 1.3: Illustration of alternative splicing and generation of transcripts.

As shown in Figure 1.8, mRNA consists of multiple exons and introns. Splicing of the mRNA sequence can generate multiple transcripts of the same gene. For example, if the gene name is ABC1, then the selection of exon 1, 2 and 4 will generate transcript ID ABC1.1, exon selection 2 and 3 will generate ABC1.2 and exon selection of 1 and 3 will generate ABC1.3. This means that all transcripts code for the same function and are represented by the gene name ABC. The entire process of transcription and translation involves three steps: In the first step, RNA polymerase binds to the promoter sequence, also known as the transcriptional start site (TSS) of the DNA strand (Figure 1.4). Transcription generates a primary transcript of the gene called pre-mRNA, which consists of exons and introns. This primary transcript contains multiple start codons (AUG) and stop codons (UAG/UGA/UAA).

The sequence which lies in between the start and stop codon is called the open reading frame (ORF). A pre-mRNA transcript may contain single, multiple or no ORFs, depending upon the sequence it contains. The exons and introns are contained within the ORF sequences which are separated by GT-AG motifs called exon-intron boundaries. The sequence starting from the start codon until the sequence before GT motif, consists of an exonic sequence (exon 1). The sequence starting from GT and ending with first AG motif consists of an intronic sequence (intron 1). The sequence beginning after the first AG motif until sequence before the GT motif is exon 2 and so on, until the end of the sequence is reached.

The second step involves production of mature mRNA transcript by alternative splicing mechanism. The mature mRNA consists of 5' untranslated region (UTR) which is located upstream to the start codon, along with 5'CAP which caps the mature mRNA sequence to provide stability during the translation process. Capping is a process in which nucleotides on the 5' end of the DNA undergoes modification to provide stability. 5' CAP and 5'UTR itself is not translated but are required for the stability. Similarly, on the 3' end of the sequence, the mRNA sequence is sometimes polyadenylated (poly-A) during the RNA sequencing. The poly-A tails also ensures stability and nuclear export of the mRNA to tRNA molecule for protein synthesis.

Figure 1.4: Illustration of transcription, alternative splicing and translation events. After (Ben-Hur *et al.*, 2008)

The third step involves export of the mature mRNA transcript from the nucleus into the cytoplasm where it binds to the rRNA molecule and the protein sequence is generated with the help of tRNA. The tRNA bearing the codon sequence complementary to the mRNA codon binds and releases the amino acid which forms the protein sequence. The protein sequence is then folded to form a protein structure which is exported to the different parts of the eukaryotic cell.

1.10 Next-generation sequencing (NGS)

NGS or deep sequencing is a high-throughput technology for sequencing of base-pairs in DNA or RNA samples. It is a revolutionary genomic tool by which valuable insights into the whole genome can be obtained. The whole genome sequencing application of NGS helps in identifying genetic variants such as Single Nucleotide Polymorphisms (SNPs), insertions, deletions and structural variants which contribute to many human diseases including cancer.

NGS technology evolved by fundamental discovery of the DNA structure and developments of sequencing methods such as Sanger sequencing (Sanger and Coulson, 1975). To achieve routine sequencing on genomic scale, advances in multiple areas were brought together which led to development of polymerase chain reaction (Saiki *et al.*, 1985, 1988). This led to development of fluorescent-based automated DNA sequencing and enabled sequencing of human genome which was accomplished in 2001 by Human Genome Project (Consortium, 2001). Since then, many technologies have emerged which promoted many advances and the growth of bioinformatics.

Current sequencing platforms require shearing of DNA into molecular weights of several different sizes. DNA fragments with higher molecular weight are extracted and prepared as libraries for sequencing. Adapter sequences are ligated to 5' and 3' ends. Different sequencing technologies use different sets of adapters which depend on compatibility of adapter sequences with downstream processes in the protocol. Pre-processing also requires choosing suitable template for preparing sequence libraries which ultimately leads to detection of signal and bases from the genome.

Several different platforms have been built for sequencing of genomes. The Illumina technology (Illumina, 2010) uses bridge amplification technology and sequencing-by-synthesis approach during library preparation steps. Fluorescently labeled dNTPs are incorporated into a growing DNA chain during sequencing such that each base is identified and acts as a reversible terminator. With this platform, high quality paired-end reads with length up to 2×150 bp can be generated in less than 30 hours.

The sequencing templates in the Ion Torrent platform (Rothberg *et al.*, 2011) developed by Life Technologies are generated on Sphere or Bead via emulsion Polymerase Chain Reaction (PCR) (Nakano *et al.*, 2003). The Ion torrent chips consist of solid-state pH sensors which detect bases incorporated during sequencing by the release of H+ ions during the extension of each nucleotide, which changes the pH within the sensor wells. Since the sequencing is based on ion detection, it fails to differentiate between different bases which leads to the generation of homo-polymer errors (Merriman, Torrent and Rothberg, 2012). With ion torrent technology, average read lengths up to 400 bp can be produced with 60-80 million reads per run with ~10X (10 times) coverage in 4 hours.

Another sequencing platform called Single Molecule Real-Time (SMRT) (Eid *et al.*, 2009) developed by Pacific Biosciences is based on single molecule detection using optics for detecting fluorescently labelled nucleotides and the ligated adapters have hairpin loop structure which becomes circular after ligation to double stranded DNA fragments during library preparation. With SMRT sequencing, read length upto 15 kbp can be produced in 4 hours. Based on the above-mentioned sequencing technologies, NGS has several applications such as expression analysis using RNA-seq (Transcriptome sequencing), methylation analysis using Methylated DNA ImmunoPrecipitation Sequencing (MeDIPSeq), identification of protein binding sites using Chromatin Immunoprecipitation Sequencing (ChIP-Seq), *de novo* Whole Genome Sequencing (WGS), disease gene identification using Whole Exome Sequencing (WES) and identification of rare variants by Targeted Sequencing (TS).

WGS is commonly used for identification of disease association in whole genome which interrogates 3.2 billion base pairs of human genome. WES on the other hand is a cost-effective method and uses targeted sequencing technology which represents sequences using less than 2% of the whole genome (van Dijk *et al.*, 2014). *De novo* sequencing sequences DNA in the absence of a reference genome where sequence reads are assembled into short reads or "contigs" and quality of the coverage depends on the continuity and size of the contigs. In targeted sequencing, only subset of genomic region is isolated and sequenced with high coverage with 500-1000x coverage which allows researchers to focus data analysis on specific area of interests and allow identification of rare variants. Transcriptome sequencing allows the study of gene expression, which provides a comprehensive snapshot of the transcriptional profile of the cell rather than a fixed subset of genes. Additionally, it allows for the detection of splice junctions, isoforms, novel transcripts and gene fusions. Methylation sequencing application is primarily based on the detection of 5-methyl cytosine (5mC) methylation states in the DNA which significantly regulates gene expression (Phillips, 2008). With ChIP-Seq analysis, protein-DNA or protein-RNA interactions can be determined which significantly affect many biological processes.

1.11 RNA Sequencing

RNA Sequencing (RNA-seq) is a method for precisely measuring transcript levels and their isoforms. This includes messenger RNAs (mRNAs), non-coding RNAs (ncRNAs) and small RNAs (sRNAs). RNA-seq determines transcriptional profiles of RNAs by quantifying gene expression levels, splicing patterns, start sites and post-transcriptional modifications. Several technologies have developed methods for quantifying the transcriptome, which includes sequence-based and hybridisation approaches, incubating fluorescently labelled cDNAs or high-density oligo microarrays for the detection and quantification of spliced isoforms. These however possess several limitations, such as restriction on the range of signal detection which often requires intricate normalisation methods. On the other hand, the microarray-based approach determines cDNA sequences but is relatively expensive, of low throughput and most importantly not quantitative.

To overcome the limitations over existing approaches, RNA-seq presents several advantages. First, RNA-seq is not limited to transcript detection of existing genomic sequences which makes it a particularly attractive tool for sequencing of non-model organisms whose sequences have not yet been determined. Secondly, the generation of short reads sequences provides information about how two exons are connected. Thirdly, RNA-seq reveals critical sequence variations in the transcribed regions (Marioni *et al.*, 2008). Fourthly, RNA-seq generates very low background noise, compared to microarrays. In contrast to microarrays, RNA-seq is highly accurate in terms of the quantification of expression levels. Therefore, RNA-seq is the first sequencing-based method that provides very high-throughput and quantitative results than other methods.

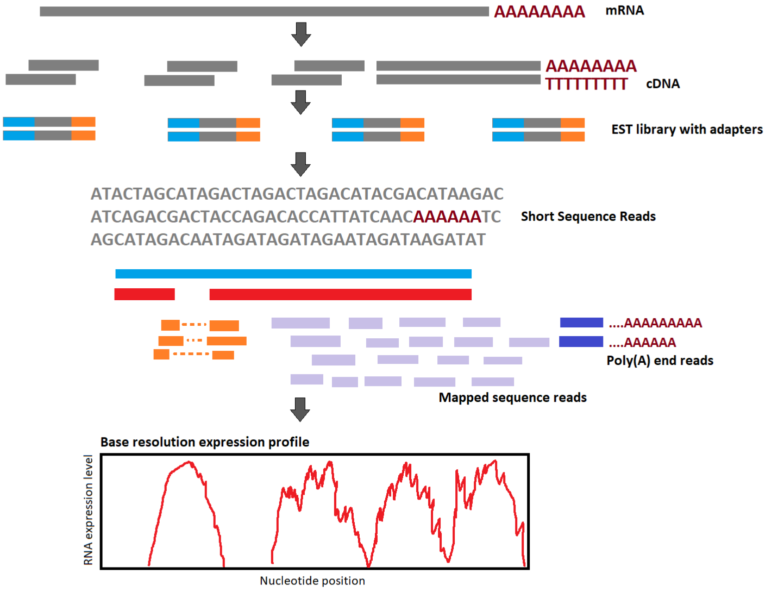

AAAAAAAA
mRNA
AAAAAAAA
TTTTTTTTT
cDNA
EST library with adapters
ATACTAGCATAGACTAGACTAGACATACGACATAAGAC
ATCAGACGACTACCAGACACCATTATCAACAAAAAATC
AGCATAGACAATAGATAGATAGAATAGATAAGATAT
Short Sequence Reads
....AAAAAAAAA
....AAAAAA
Poly(A) end reads
Mapped sequence reads
Base resolution expression profile
RNA expression level
Nucleotide position

Figure 1.5: Workflow of RNA-seq experiment. Long RNAs are converted into shorter sequence fragments by DNA fragmentation. In the next step, sequencing adapters are ligated to each cDNA fragment. Resulting sequence reads are then aligned against reference genome which are then classified into junction reads, exonic reads and poly(A) end-reads. These three types are then used to generate expression profile for each gene.

For the identification and quantification of RNA sequences, several steps are involved in transcript profiling (Figure 1.5). Unlike smaller RNA sequences such as micro RNAs (miRNAs), piwi-interacting RNAs (piRNAs) and short interfering RNAs (siRNAs) which can be easily sequenced after ligation of adapters; larger RNA sequences require fragmentation into 200-500 bp short read sequences for compatibility with major high-throughput sequencing approaches. Apart from sequencing, RNA-seq also faces certain informatics challenges such as development of efficient approaches for storing, retrieving and processing large amount of data to reduce errors in base calling and removal of low quality reads. Once high quality reads are obtained, short reads are mapped to reference genome using bioinformatics tools such as

Bowtie (Langmead *et al.*, 2009), STAR (Dobin *et al.*, 2013) and Tophat (Trapnell, Pachter and Salzberg, 2009).

Read mapping reveals the transcriptome landscape of a sequenced sample. Where poly(A) sequences are identified by the presence of multiple As and Ts at the end of sequence reads, exon-exon junctions are identified by the presence of specific sequences (GT-AG dinucleotides) which can be confirmed by the detection of lower expression of intronic sequences that are removed during splicing. For larger transcriptomes, performing alignment leads to alignment of sequence reads at multiple locations on the genome. A solution to this problem is to assign these sequences based on mapping of reads to neighbouring unique sequences which can be applied to low copy number repeat sequences (Mortazavi *et al.*, 2008). However, for reads having higher copy numbers and larger repetitive regions, paired-end sequencing can be applied in forward and reverse directions of the DNA strands, which extends the fragment length to 200-500 bp.

1.11.1 RNA-seq data analysis

For performing sequencing analysis, a workflow is required which entails steps for processing of raw sequence data and preparing it for further downstream analysis. Since RNA-seq has a variety of applications and analyses scenarios, an optimal pipeline cannot be suggested. The application of a workflow is adopted depending on the organisms being studied and their research objectives. For organisms having sequenced genomes, short reads from samples are aligned to the reference genome whereas for those without any sequenced genome, *de novo* assembly of reads is performed, which is followed by the mapping of contigs onto the transcriptome.

Every experimental scenario in RNA-seq data analysis consists of five primary steps:

(1) Performing quality checks,
(2) Sequence alignment,
(3) Transcript quantification,
(4) Normalisation, and
(5) Identifying DEGs.

A key step in RNA-seq data analysis is the identification of DEG. Quality control of raw reads involves analysis of GC content, sequence quality, presence of adapter sequences, duplicate reads, overrepresented *k*-mers and possible contamination due to PCR artifacts. The checks mentioned above can be performed by FastQC (http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/) or NGSQC (Dai *et al.*, 2010) tools. For trimming low-quality reads and adapter sequences, Cutadapt (Martin, 2011) or Trimmomatic (Bolger, Lohse and Usadel, 2014) are generally used which eliminates these sequences to retain high-quality reads.

The second step is the alignment of sequence reads where reads are mapped against the transcriptome reference sequence. This involves fine-tuning of multiple parameters, which depends specifically on the organism under study. Since the majority of reads are mapped at multiple locations, the fraction of multi-mapping reads is comparatively higher than those coming from unannotated transcripts. In the absence of reference genomes, RNA-seq reads are assembled *de novo*, using the tools Trinity (Haas *et al.*, 2013), Trans-ABySS (Grabherr *et al.*, 2011) and SOAPdenovo-Trans (Xie *et al.*, 2014b). Due to the presence of low expressed transcripts, it becomes impossible sometimes to assemble these reads as they lack sufficient coverage and therefore leads to misassembly of reads. Hence, it is often recommended to perform computational reduction of reads (Haas *et al.*, 2013).

Once the reads are aligned, the estimation of gene and transcript expression is required. For transcript quantification, raw counts of mapped reads are aggregated using the HTSeq-count tool (Anders, Pyl and Huber, 2015) which uses a Gene Transfer Format (GTF) file containing genomic coordinates of genes and exons for producing raw read counts. Since raw read counts are often affected by multiple factors such as sequencing bias, total number reads and transcript length, normalisation of raw sequence counts is performed to convert it to a RPKM (Reads Per Lilobase of exon model per Million) mapped reads value which removes library-size and feature length effects. Some tools convert raw sequence counts to FPKM (Fragments of Per Kilobase of exon model per Million) mapped reads or TPM (Transcripts Per Million) values. Correction of gene length is required for comparing gene expression changes within gene and across samples. Tools such as Cufflinks (Trapnell *et al.*, 2012) estimate transcript expression with the aid of the expectation-maximisation approach, using mapped reads aligned from Tophat. The key step in RNA-seq analysis is the DEG step that requires the comparison of gene expression values among samples.

Normalisation methods such as RPKM, FPKM and TPM normalize the sequencing depth. However, these methods perform poorly when samples have diverse transcript distributions, which skews the distribution of counts. To resolve such issues, normalisation methods are preferred, such as DESeq (Anders and Huber, 2010) and PoissonSeq (Li *et al.*, 2012) which ignore highly variable features. Other normalisation packages such as NOISeq (Tarazona *et al.*, 2012) identify sources of biases in the data and correct the variation in transcript length across samples, positional bias in coverage and GC contents.

Despite sample-specific normalisation, batch effects are sometimes present in the data which can be removed by batch correction methods such as COMBAT (Johnson, Li and Rabinovic, 2007). Some popular methods, such as edgeR (Robinson, McCarthy and Smyth, 2010) conduct an integrated normalisation and differential expression analysis by using a negative binomial distribution for normalisation. Certain Bayesian approaches such as EBSeq (Leng *et al.*, 2013) and baySeq (Hardcastle and Kelly, 2010) utilizes negative binomial distribution by computing the posterior probability of each experimental group and for each gene.

1.12 Non-coding RNAs

Non-coding RNA (ncRNA) commonly refers to the class of RNA that does not encode proteins. This means they do not contain any information and hence do not perform any function. However, recent evidence suggests that the majority of the genomes are in fact transcribed, which includes miRNAs and snoRNAs (Dieci, Preti and Montanini, 2009; Schanen and Li, 2011). Most of the functions of ncRNAs are still unknown and might have important regulatory functions such as RNA splicing, DNA binding, transcription, translation and turnover.

From past years, it has been clear that the mammalian transcriptome largely consists of two major types of ncRNAs, namely (1) small non-coding RNAs (sncRNAs) and (2) long non-coding RNAs (lncRNAs). sncRNAs is a family made up of three sub-classes of ncRNAs: (a) short interfering RNAs (siRNAs), (b) miRNAs and (c) piRNAs. These have been associated with multiple biological pathways which leads to specific gene silencing and protection against viruses, retro-elements, mobile repetitive DNA sequences and transposons (Moazed, 2009).

miRNAs and siRNAs are 20-30 nucleotides (nt) long RNA sequences which originate from the double stranded RNA (dsRNA) precursors and are endogenously produced during gene expression on sense and antisense DNA strands. siRNAs are small RNA duplex molecules that are produced by the ribonuclease III enzyme (Meister and Tuschi, 2004). miRNAs are transcribed by RNA polymerase II and possess a stem loop structure (Jinek and Doudna, 2009).

piRNAs are 24-31 nt long RNA sequences and are one of the least characterised class of sncRNAs which are extensively expressed in different cells and tissue types. Although not much is known about this class of sncRNA, certain loss-of-function mutations studies identified the role of piRNAs in transposon silencing (Chen, Pane and Schüpbach, 2007). Certain studies also demonstrate their role in developmental gene regulation and heterochromatin formation (Rangan *et al.*, 2011; Simonelig, 2011).

1.12.1 Long non-coding RNAs

Although at least 2% of the genome has been reported to contain protein-coding genes, while the remaining 98% of the human genome consists of non-protein coding sequences, most of which contains lncRNAs. Surprisingly, transcription is not limited to protein-coding genes but is in fact ubiquitous in mammalian genome (Carninci *et al.*, 2005). Actually more than 90% of the genome is probably transcribed (ENCODE Consortium, 2007). A hypothesis stated by Ulitsky and Bartel says that most of the annotated lncRNAs are non-functional (Ulitsky and Bartel, 2013). Due to the non-perfection of the transcription machinery, spurious RNAs are produced with no significant biological purpose (Struhl, 2007) and many lncRNAs are polyadenylated, capped and spliced.

However, many lncRNAs have been reported to play important biological roles in regulation and transcription processes. For example, *Xist* lncRNA has been reported to control X-chromosome inactivation (Penny *et al.*, 1996). lncRNAs have also been reported to play an important cell cycle regulatory roles as well as in the establishment of the cell identity (Pauli, Rinn and Schier, 2011; Rinn and Chang, 2012). Importantly, lncRNA dysregulation has been found to be associated with several human disorders such as in cancer and neurological disorders (Mitra, Mitra and Triche, 2012; Bhan and Mandal, 2014). lncRNAs exhibit distinct expression patterns in tumors and metastases, which can be primarily used for diagnosis and prognosis of cancer and could potentially serve as an aim for therapeutics (Tsai, Spitale and Chang, 2011).

lncRNA sequences are comparatively smaller in size than mRNA sequences. They have also been found to possess fewer exons on an average which is attributed to both incomplete assembly and lower abundance (Pauli *et al.*, 2012). Regarding features and characteristics of lncRNAs, the basic features are equivalent to mRNAs. Firstly, lncRNAs have also been known to exhibit alternative splicing (Derrien *et al.*, 2012). From an experimental dataset obtained by Cabili et al. (2011), 98% of spliced lncRNAs possess only two exons. Secondly, lncRNAs are characterized by ‘K4-K36’ domains consisting of histone lysine trimethylation along with the transcribed regions (Cabili *et al.*, 2011; Derrien *et al.*, 2012). Thirdly, through experimental studies, lncRNAs have been found to be transcribed by RNA polymerase II and contain canonical polyadenylation signals, similar to mRNAs (Pagano *et al.*, 2007). In contrast to mRNAs which are highly conserved in multiple species, lncRNAs exhibit poor sequence conservation. However, lncRNA sequences originating from the promoter region of the DNA have been found to be more conserved exhibiting similar characteristics to mRNAs (Cabili *et al.*, 2011; Derrien *et al.*, 2012). Similar to mRNAs, lncRNAs also contain Open Reading Frame (ORF), however the length of the ORF is much shorter than those found in mRNAs (Dinger *et al.*, 2008).

1.13 Machine learning

Machine learning methods are approaches for learning functional relationships from the data without any need to define them a priori. It is a process which causes the system to improve with experience by learning from data provided to the machine. There are three main categories of machine learning: (1) Supervised learning, (2) Semi-supervised learning and (3) Unsupervised learning. In supervised learning, both input and output variables are observed and the results of subsequent classification processes depend on the output results from previous steps. Semi-supervised learning refers to the class of supervised learning where the classifier classifies a large amount of unlabeled data, using small amounts of labelled data. Unsupervised learning is the method where inferences are drawn from datasets having unlabeled response variables. The general workflow of machine learning consists of the following seven steps (Figure 1.6):

1) Data acquisition: The first step in machine learning is to acquire the data needed for training and testing. Therefore, reliable data should be obtained to solve the problem and to perform initial descriptive analysis.
2) Feature extraction: The second step is to extract relevant features from the dataset under study. For example, if flower petals and sepals are studied from the iris flower dataset (Bache and Lichman, 2013), then values are extracted for each characteristic feature such as petal length, petal width, sepal length and sepal width. This step also involves construction of response class values. For example, in the case of binary classification, certain features are extracted from Sentosa flower, while other features are extracted from versicolor plant.
3) Data preparation: The third step is to prepare the data by removing outlier values, removing missing values, transformation of non-numeric to numeric values and normalising the values by scaling them between a given range.

4) Classifier selection: Once the features are extracted and data is normalised, a classifier is required to make decisions on the data and provides good accuracy on the testing data. Several classifiers have been developed for performing classification, such as Random Forests, Support Vector Machines, Neural Networks, Logistic Regression, Linear Regression, KNN and K-Means. The right classifier to apply depends on the dataset used in the study.
5) Model fitting: Once the appropriate classifier is chosen, training data is selected from the normalised data matrix. The training data is then fed into the classifier for learning and development of a predictive model. This predictive model is specifically dependent on the training dataset as the classifier only knows the values which have been provided in the training set.
6) Prediction of testing data: After the classifier is trained and the predictive model is generated, testing data is used for prediction into either one of the classes.
7) Model validation: In this last optional step, the predicted testing dataset is used for cross-validation, such that the complete dataset is divided into equally sized groups called folds, consisting of training and testing data, and each fold consists of separate training and test sets. This process is repeated so that each fold receives an opportunity of being left out and therefore act as testing data. Model validation process helps in evaluating the classifier capability and feature strength.

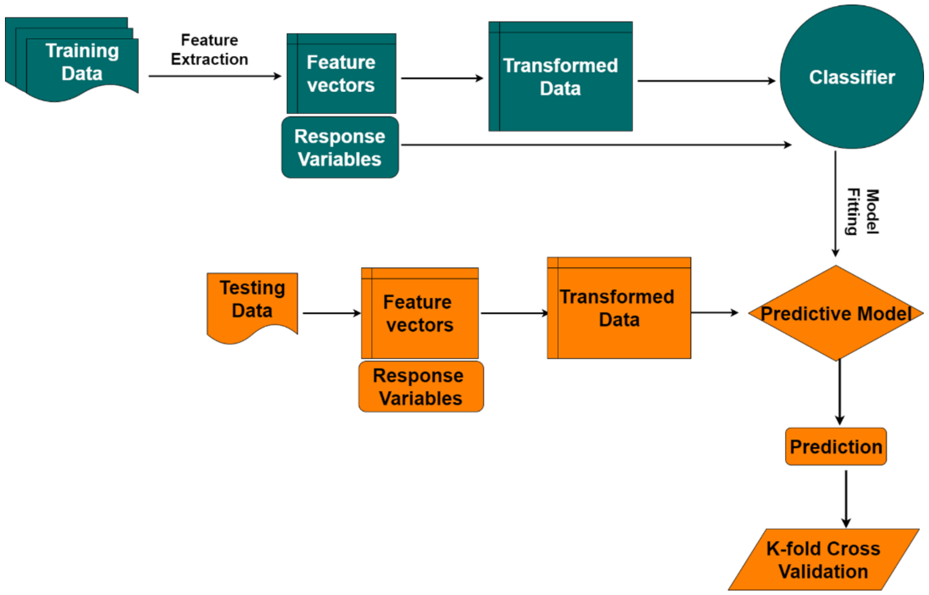


Figure 1.6: Schematic workflow of machine learning process.

With advancements in high-throughput genome sequencing, which has resulted in generation of thousands of samples, new completed sequences are getting deposited in the repositories everyday which has led to an enormous increase in the volume of the data and a need to computationally analyze massive amounts of data with smarter algorithms in computer science. There are approximately around a thousand databases of interest to biologists (Galperin, 2008) which contain crucial information, ranging from sequences, structures, annotation, networks, etc.

Machine learning has many applications in engineering and computing, such as pattern recognition, process optimisation and image analysis. Apart from its applications in engineering and computing, it has also several known applications in computational biology which includes identification of protein-coding genes from genomic sequences, prediction of protein function, identification of binding sites which includes protein-DNA and protein-RNA, prediction of protein secondary and tertiary structure based on amino acid sequence (Cheng, Tegge and Baldi, 2008; Suresh, Gromiha and Suwa, 2015a; Liu, 2017). These applications may require supervised or unsupervised learning, for example, given a dataset of protein sequences with experimentally labeled associated functions, a classifier can be trained such that it can predict the function for a novel sequence, which can be performed using supervised learning, whereas identification of functional modules in gene expression data can be performed using the unsupervised learning strategy. Since the function of a novel sequence is unknown, supervised learning strategy helps in predicting its function based on the sequence characteristics of similar sequences associated with functions. Whereas the functional modules of gene expression data can be inferred from the public databases such as Kyoto Encylcopedia of Gene and Genome (KEGG) (Ogata *et al.*, 1999) and Gene Onotology (GO) (Ashburner *et al.*, 2000).

In a traditional problem, features extracted from genomic data are constructed into a feature set and are generally associated with a single class label. This is called the *single-label classification problem*. When two classes are present, features in the set are associated with either of the two elements, which is called the *binary classification problem*. If the set has more than two elements, then it defines a *multi-class classification problem* which has been applied in several biological applications. An example of such type of class is where many proteins and genes have multi-functional association. Most existing algorithms fail to classify them due to the complexity of the data, therefore they assign it to a subset of the node in the hierarchy when performing classification, which produces *hierarchical multi-label classification problem*

(Boutell *et al.*, 2004). The solution to this problem is to transform it into *k*-binary class classification by constructing *k* datasets, such that the dataset is labeled as *1* if has certain label or *0* otherwise. The classifier returns prediction of the test set by classifying into one of the class labels.

1.13.1 Random Forests

Random Forest (RF) is a method for making predictions by averaging over several predictions of independent base models. The RF was introduced by Breiman (2001) who originally devised it as a method for combining several classification and regression trees, using the bagging approach. Since its first introduction, RF has been applied in several applications to solve numerous problems. RFs are built by combining independently trained predictions from several trees. Prediction using RF is based on creation of decision trees. Decision trees are created based on a rule-based system. Given a dataset of features and targets, the decision tree algorithm performs the prediction on the test dataset based on the set of rules. The decision trees can be understood by considering a binary classification scenario. For example, playing with a ball is dependent on weather conditions. To decide whether to "play" or "not play" is guided by the creation of a tree. The decision is made based upon the traversal down the leaf nodes where the data is bucketed into smaller parts. This can be clearly illustrated by Figure 1.7. The data shown in the figure demonstrates weather characteristics of 14 days which are labelled as "Play" and "Not Play". The data consists of five features, namely, sunny, overcast, rain, humidity and wind conditions. Based on the feature values, the prediction on test data is made whether a day is suitable for playing or not playing outside.

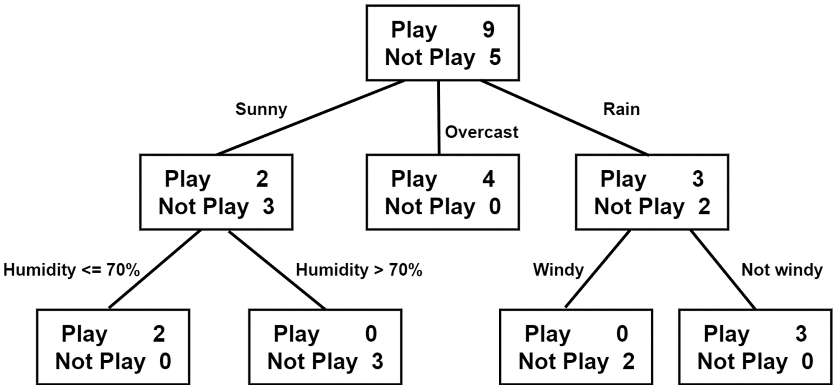


Figure 1.7: Example illustration of decision tree estimation on weather data.

The feature set also consists of two classes: "Play" and "Not Play". Based on the principal of bootstrapping, random trees are generated (Figure 1.8). An example of a random tree is shown in Figure 1.7. The prediction of the test dataset is dependent on both the feature values and class labels. The correlation between these is important for accurate prediction of test data into one of the following classes. In this example, the dataset consists of 9 and 5 samples classified into “Play” and “Not Play”, respectively. The feature matrix is split based on the threshold and binary values of the features. The tree is first split into three leaf nodes based on three primary features containing discrete values: “Sunny”, “Overcast” and “Rain”.

Based on the split, two samples have been recognized as “Sunny” and fit for playing football outside, whereas three samples have been recognized as “Not Play” and considered as inappropriate for playing football outside. The first leaf node is further splitted into sub-leaves based on the “Humidity” feature with samples having threshold values <= or > 70. If the values are <= 70, the day is labeled as “Play”, otherwise “Not Play”. The second leaf node consists of four samples categorised based on “Overcast” feature. The third leaf node classified five samples based on “Rain” featue. These samples are further splitted into sub-leaves based on “Windy” and “Not windy” features. If the feature values are positive for “Windy” within “Rain”, the samples are classified into “Not Play”, otherwise the samples classified into “Not windy” are classified as “Play”.

Each decision tree in the random forest starts the traversal with a root node from which splitting takes place. The attribute value in the root node is compared with the internal nodes until decision node is reached. The root node for each tree is selected based on two criterions: Information Gain and Gini Index. These criterions calculate the values for each attribute or feature. The calculated values are sorted, and higher value is assigned the root node in the tree. Each tree creates several leaf nodes based on the features selected in that decision tree.

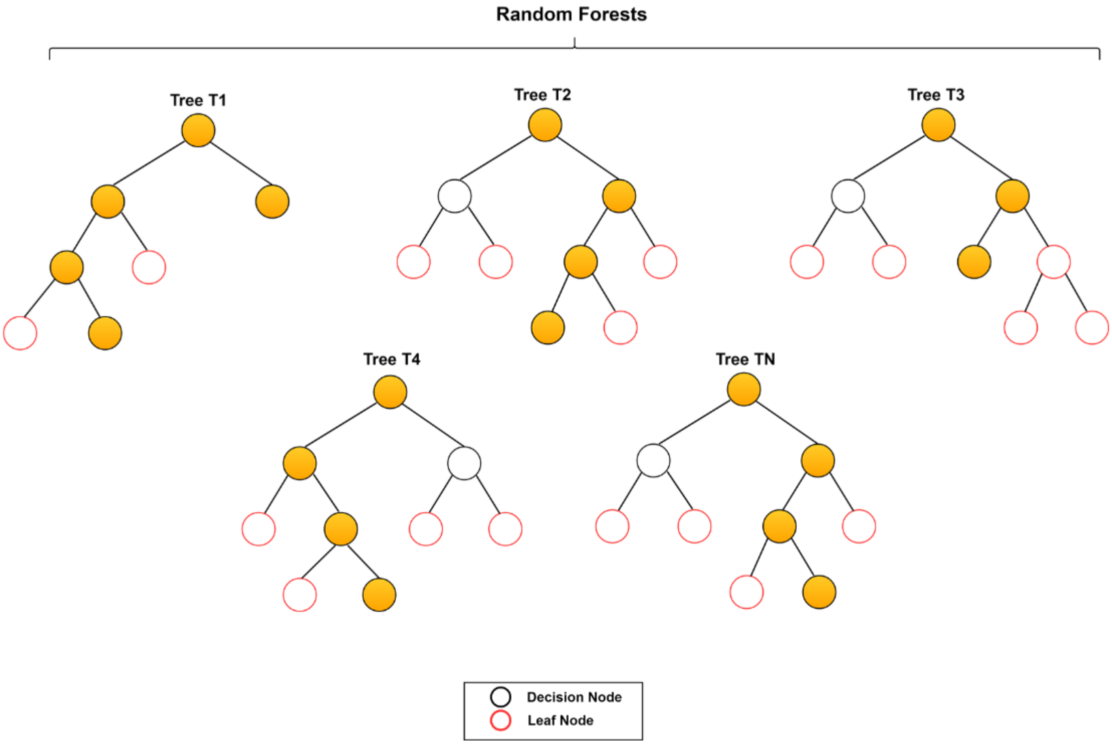


Figure 1.8: Illustration of RF model with each tree consisting of decision nodes and leaf nodes.

There are two steps in RF algorithm: (1) creating RF, and (2) performing predictions based on RF created.

To construct a tree, there are three major choices which should be made and considered: (1) the methods for leaf splitting, (2) the type of the predictor that will be used in each leaf, and (3) the method for injection of randomness into the trees.

To specify a method for leaf splitting, the selection of the shapes of candidate splits and the method for evaluation of quality of each candidate are required. For leaf splitting, axis aligned splits can be used, where the data is routed to sub-trees which depend on whether it exceeds the threshold value. The threshold value can be randomly chosen by optimising a function. To split a leaf, a collection of splits is generated and a candidate split is chosen which optimizes the purity function over the created leaves and which maximises the information gain (Hastie, Tibshirani and Friedman, 2009).

The second choice is to choose the type of predictor and the most common choice is to use the average response over the training points falling over a leaf. Third choice is to inject randomness in the tree construction, where the dimensions need to be chosen for splitting candidates at each split and coefficients should be chosen for generating random combinations of features. Another method for introducing randomness is to use a sub-sampled or a bootstrapped dataset for building each tree, which introduces differences between the trees.

To find the patterns in the data and achieve the decision node in the tree, the randomness is injected by selecting random records and random features. Each tree is built from random samples of data using bootstrap sampling. This generates random samples of data for each tree which sometimes leads to overfitting. Random feature selection involves examination of each feature and selecting the best split from the features. Therefore, RF selects random subset of features for each split.

Therefore, the complete algorithm works by growing M randomised trees. Before construction of each tree, n observations are randomly drawn from the dataset. Then a split is performed on each cell of the tree by maximising the Classification and Regression Trees (CART) criteria (Breiman *et al.*, 1984). The CART criteria measure the difference between the variance before and after the split is performed. This process is repeated M times. As M grows the variance decreases. This also reduces overfitting and hence, more accurate predictions can be obtained with large values of M (Breiman, 2001).

1.13.2 Iterative Random Forests

With the development of tree-based methods in machine learning, several methods have been developed for the detection of interactions among the features, which include RF (Breiman, 2001), CART (Breiman *et al.*, 1984), Forest Garrote (Meinshausen, 2009), Node Harvest (Meinshausen, 2010) and RuleFit3 (Friedman and Popescu, 2008). These methods have been applied in the field of genomics as machine-learning classifiers such as identification of DNA tetranucleotide frequencies in bacterial genomes (Dyer, Kahn and Leblanc, 2008), regression of peptide data against RNA expression data (Bánfai *et al.*, 2012), application of RF in large genome-wide association studies (Goldstein *et al.*, 2010), classification of microRNA precursers (Jiang *et al.*, 2007) and prediction of non-synonymous polymorphisms (Bao and Cui, 2005). However, these methods produce shallow trees to prevent overfitting, with the exception of the RF and therefore exclude the possibility of the detection of higher-order interactions without affecting the accuracy in a computationally feasible manner. The RF creates deep decision trees that produce higher-order interactions without affecting the prediction accuracy. Due to the instability in decision paths, interpretation of results from RF remain a challenge and therefore cannot be considered as an alternative.

Iterative Random Forests (iRF) (Basu *et al.*, 2018) overcome these challenges by searching important local higher-order interactions. iRF algorithm is based on the Principle of Stability (Yu, 2013) which grows feature-weighted RF sequentially to perform a soft dimensional reduction of feature space and to stabilise decision paths. The fitted RF are decoded and higher-order feature interactions are determined by the Random Intersection Trees (RIT) algorithm (Shah and Meinshausen, 2014) which extracts stable higher-order feature combinations in the RF decision tree ensemble. The iRF uses the supervised learning approach for identifying class-specific index sets which are needed for RIT algorithm. This framework allows for detection of higher-order combinations in feature-weighted RFs. iRF classifier has been applied for the prediction of Drosophila embryo and alternative splicing transcripts in human-derived cells where it derived novel third-order transcriptional factor interactions (Basu *et al.*, 2018).

For classification and determination of interactions, iRF consists of three steps:

1) Iteratively re-weighted Random Forest
2) Generalized RIT
3) Bagged Stability Scores

By performing these steps, iRF can recover higher-order combinations of features. The details of the algorithm implementation will be discussed in Chapter 4 (Methodology).

1.13.3 Support Vector Machines

A Support Vector Machine (SVM) is a supervised ML algorithm which is used for classification problems. A SVM plots each data point into n-dimensional space (where n=number of features) with each feature value being a coordinate. The classification is performed by finding a hyper-plane that separates the two classes effectively (Figure 1.9). A hyperplane is a line that linearly separates and classifies the data points. The further the data points from the hyperplane, the stronger the classification accuracy. The distance between the data points and the hyperplane is known as the margin. For achieving higher accuracy within the training dataset, a hyperplane separating the data points with higher margin values is required.

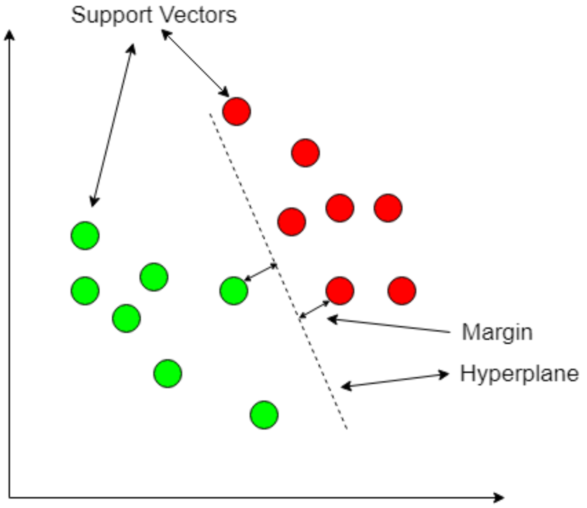


Figure 1.9: Illustration of an SVM containing Support Vectors separated by a hyperplane.

The classification process primarily depends upon the identification of an optimal hyperplane. To accurately separate the red and green circles, the SVM starts by constructing three different hyperplanes: A, B and C (Figure 1.10). Scenario-1 illustrates the hyperplanes segregating the data points in an n-dimensional space whereas scenario-2 demonstrates margin distances between the data points and three hyperplanes. The scenario shows that the margins between the data points and hyperplanes B and C are smaller than the hyperplane A. Therefore, hyperplane A maximises the margin of the training data.

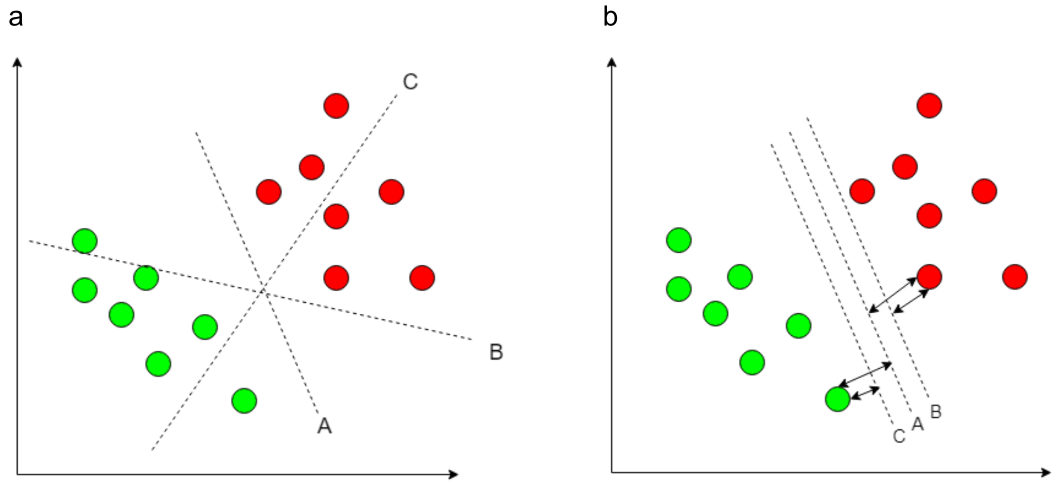


Figure 1.10: Illustrations of different hyperplanes in SVM classification. (a) Scenario-1, and (b) Scenario-2.

1.14 Feature selection in machine learning

With an increase in dimensionality, computational cost increases as well. To counter this problem, there are two approaches. First is the subset selection of features and second is to extract meaningful features. A typical example of multi-dimensional complex data is the microarray cancer data, where the data can be originated from many cancer types n and each type of data can have multiple features m which makes $n\times m$ features. When machine learning methods are applied on the data, the general outcome of the study depends on whether the data can be classified as cancerous or non-cancerous. The feature subset selection criteria work by removing redundant or non-relevant features from the dataset provided that selected features give best performance according to an objective function. However, when compared to feature extraction methods, feature selection does not alter the natural representation of the data (Saeys, Inza and Larranaga, 2007). Algorithms for feature selection are categorised into three types:

1) Filters: Those which extract features from the dataset without requiring any learning.
2) Wrappers: Those which use the learning approach for evaluating whether the features are useful.
3) Embedded techniques: Those which combine feature selection and classifier construction steps.

Filter methods works without any classifier which makes them computationally efficient. They are divided into univariate and multivariate methods. Univariate methods, such as unconditional mixture modeling (Law, Jain and Figueiredo, 2000) assumes binary states of the genes which affect the classification process using mixture-overlap probability. Whereas multivariate methods, such as markov blanket filtering (Zeng, Luo and Lin, 2009) finds features which are independent of class labels, such that the removal of these features does not affect accuracy. Another popular multivariate feature selection method used in Machine Learning (ML) is the minimum Redundancy Maximum Relevance (mRMR) (Peng et al., 2005a) which maximises the relevance of genes with class label while minimising redundancy in each class. mRMR uses Mutual Information (MI) for measuring the information a random variable gives about another, such as class label or gene activity (Peng et al., 2005b). Another multivariate approach is Correlation-based Feature Selection (CFS) (Hall, 1999) which is based on the principle that a feature subset is good when it highly correlates with a class and does not correlate with one another. Therefore, CFS evaluates the features based on this criterion. Another method ReleifF selects features which distinguishes the data among different classes (Hall and Smith, 1998).

Wrapper methods are based on a supervised classification approach due to which these methods can be computationally inefficient. These are generally based in two categories: Deterministic and Randomised. Deterministic wrappers use a combination of wrapper and Sequential Forward Selection (SFS) for the feature selection, by adding several possible single-attribute expansion to existing attributes and evaluating the accuracy on each step (Pudil, Novovičová and Kittler, 1994). The Feature Selection (FS) starts with an empty set of features. Features are added to the empty set one-by-one and accuracy is evaluated using a Support Vector Machines (SVM), a neural network, or k-nearest neighbors. Randomized wrappers use Genetic Algorithms with SVM (GA-SVM) and simulated annealing. GA-SVM creates a population of chromosomes as binary strings representing feature subsets which are evaluated using an SVM (Perez and Marwala, 2012). Whereas simulated annealing works by exploring neighbours for seeking solutions which minimises the objective function and avoid local minima.

Embedded methods work better, in contrast to wrapper methods, by performing classifier dependent selection which might not work with other classifiers. A popular implementation of the embedded technique is the RF method, where RF are created iteratively and the forest with the smallest number of features producing lowest error is selected. This method of feature selection is called Block Diagonal Linear Discriminant Analysis (BDLDA) (Lingyan *et al.*, 2009). BDLDA works by limiting the number of features by imposing a block diagonal structure on the covariance matrix. Accuracy is evaluated by SVM classification and features are selected based on the accuracy. The SVM-Recursive Feature Elimination (SVM-RFE) method begins by including all features and excludes those features that cannot identify separating samples in different classes (Huang *et al.*, 2014).

Feature selection can also be achieved with statistical regression techniques such as *t*-statistics, in which the significance of individual predictors can be judged under the assumption that the set of predictors is fixed in advance. Fixing the predictor set can lead to bias and overfitting during the classification. An example of overfitting is commonly observed in the Partial Least Squares (PLS) method (Abdi, 2003). Therefore, improvement in feature selection provides interpretable and unbiased generalised models with accurate predictions.

There are several kinds of regression methods which have been developed for model fitting and feature selection. The first method is the Forward Stepwise Regression (FSR) which begins by selecting a single predictor variable which produces the best fit or smallest residual sum of errors (Mundry and Nunn, 2009). Subsequently, another predictor is added, which produces the best fit in combination with the first one, which is followed by the third and so on. This continues until the stopping criteria is reached, which is based on no improvement in fit.

Similar to FSR, another approach is the Backward Stepwise Regression (BSR), in which we start with a larger subset of features and iteratively remove one-by-one similar to the implementation by SVM-RFE except it uses Efroymson's procedure (Efroymson, 1960) that combines backward and forward steps. The linear regression method simplifies the BSR approach by computing the stepwise and subset procedures. Computing the procedures in a single pass through the dataset significantly improves speed. However, all these methods utilise linear regression by computing the residual sum of squares which has a high tendency of overfitting. Shrinkage methods on the other hand estimate coefficients by significantly reducing the variance, thereby improving the problem of overfitting.

Shrinkage methods such as Ridge regression (RR) (Marquardt, 1970; Tibshirani, 1996) reduces the variance by adding a degree of bias or penalty to the regression estimates, thereby shrinking the estimates towards zero. Therefore, RR has two major advantages over Ordinary Least Squares (OLS) method: (1) It penalises the estimates such that less influential features are more penalised than higher influential ones and (2) the addition of penalty term converts correlation of variables (multicollinearity) to independent variables. Such methods are called "penalised regression" methods. Least Absolute Shrinkage and Selection Operator (LASSO)

(Tibshirani, 1996) is a similar technique to RR: it also penalises regression coefficients and shrinks the values to zero. LASSO differs from RR such that instead of squares it uses absolute values in the penalty function which leads to penalisation of values to exact zero. Penalty value is directly proportional to shrinkage. The larger the penalty term, the more estimates get shrunk to zero.

1.15 Bayesian networks and Markov models

The Bayesian network represents a probabilistic relationship between a set of random variables. The relationship between the random variables is represented by a joint probability distribution. A Bayesian network consists of two major parts: Directed Acyclic Graph (DAG) and a set of conditional probability distributions. The DAG consists of set of random variables which are represented by nodes in the network. A directed edge between two nodes in the network represents an existence of causal probabilistic dependence between two random variables. A conditional probability distribution for each node in the network is defined by a possible outcome of the preceding causal nodes. Any node in the Bayesian network is conditionally independent of all the nodes in the network given their relationship with parent nodes. Therefore, the joint probability distribution of all the random variables in the network factorizes into a series of conditional probability distributions of random variables given their relationship with parent nodes.

A Markov Random Field (MRF) or a Markov network is a class of Bayesian network represented by undirected graphs. The Bayesian Networks (BN) represents the probability distribution of variables by directed graph. MRFs possess several advantages over BNs: (1) Due to non-dependency relationships in MRF, they can be applied to wider range of applications; (2) MRFs can express certain relationships or dependencies which BNs cannot easily describe; (3) MRFs provide more abilities than BN.

If A, B, C and D are variables which are connected to each other such as: (A,B), (B,C), (C,D) and (D,A), the relationship between these variables is represented by an undirected graph (Figure 1.11).

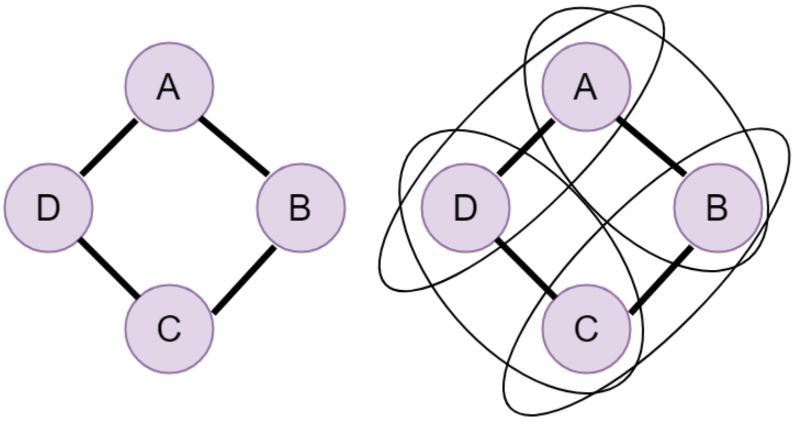


Figure 1.11: Representation of an undirected graph having joint probability over four variables. Graph on the right represents pairwise factors present in the model.

The joint-probability of the four variables is expressed by the following form:

$$\tilde{p}(\ \ ,B,C,D)\mathrm{e}\quad \phi(\mathrm{e},B)\phi(B,C)\phi(C,D)\phi(D,\ \ )\mathrm{e} \qquad (1.1)$$

where $\phi(X,Y)$ is a factor that assigns more weights to the relationship between X, Y. Therefore, the factors in the unnormalised distribution becomes:

$$p(A,B,C,D) = \frac{1}{z}\tilde{p}(A,B,C,D)\mathrm{e} \qquad (1.2)$$

where z is a normalising constant which ensures that the distribution sums to one.

A MRF is a probability distribution over the variables $x_1, x_2, x_3, \ldots, x_n$ defined by an undirected graph where nodes correspond to variables x_i. The probability distribution is given by:

$$p(x_1, x_2, \ldots, x_n) = \frac{1}{Z}\prod_{c \in C}\phi_c(x_c) \qquad (1.3)$$

where C denotes set of cliques (cliques are fully connected subgraphs). The value of Z is calculated as follows:

$$Z = \sum_{x_1, x_2, \ldots, x_n}\prod_{c \in C\mathrm{e}}\phi_c(x_c). \qquad (1.4)$$

Thus, the probability distribution in a graph G may contain factors which are determined by clique in G which can be node, edge, triangle, node, etc.

MRF is mainly applied for “guilt-by-association” approaches, particularly in protein function prediction problems where a network is constructed. The edges represent pairwise interactions between the proteins in the network. The network is generally represented by three classes of interactions. The first class of interaction is $G_1(1,1) = \beta^{(1,1)}$ where both interacting proteins performs functions; the second class of interaction is $G_1(1,0) = \beta^{(1,0)}$ where only one interacting protein performs function; and the third class of interaction is $G_1(0,0) = \beta^{(0,0)}$ where none of the interacting proteins have known functions. The corresponding number of protein pairs in the three classes are defined as N_{11}, N_{10} and N_{00}. Therefore, the energy function of the MRF is defined using the classes such that:

$$\alpha\mathrm{e}\sum_{i=1\mathrm{e}}^{N\mathrm{e}} x_{i\mathrm{e}} + \beta^{11}N_{11} + \beta^{10}N_{10} + \beta^{00}N_{00}. \tag{1.5}$$

For determining functions of unannotated proteins, repeated sampling of the neighbouring proteins having unknown function is performed which is defined by Gibbs sampling (Geman and Geman, 1984).

A similar method called Bayesian Markov Random Fields (BMRF) implements the Bayesian approach and draws inference from the joint probability density of $x, \alpha, \beta^0, \beta^1$ using Markov Chain Monte Carlo (MCMC) (Geyer, 1991). Using Gibbs sampling method, the elements of $x^{(t)}$ which corresponds to unannotated proteins are updated based on the values of $x, \alpha, \beta^0, \beta^1$. The parameter update of $x, \alpha, \beta^0, \beta^1$ values is performed using the Differential Evolution Markov Chain (DEMC) (Ter Braak and Vrugt, 2008) method. The use of an adaptive DEMC approach in MRF leads to the accurate estimation of parameters and protein prediction when compared to the standard implementation of MRF for protein function prediction application.

1.16 Tools for RNA-seq data analysis

1.16.1 Development of RNA-seq workflows in plants

In the past few years, several research studies have been conducted for the development of computational workflows and bioinformatics tools employed for processing and analysis of RNA-seq data. A study conducted by Liu et al. (2014) provided a comparison and benchmarking of several methods for the detection of differential splicing of transcriptomes in plant species. The study compared eight software tools using simulated and real *A. thaliana*

RNA-seq data. From the analysis they found that the annotation accuracy provides a major impact on the detection of alternative splicing events, therefore they suggested the consideration of annotation in Differential Expression (DE) analysis. Altogether, Cufflinks (Trapnell *et al.*, 2012) showed a better tradeoff between recall and precision metrics in the presence of incomplete annotation. Whereas DEXSeq (Anders, Reyes and Huber, 2012) performed relatively well for simulated data with an accurate and strong annotation of alternative splicing. In the case of complex alternative splicing events, Multivariate Analysis of Transcript Splicing (MATS) (Shen *et al.*, 2012) showed better performance for real RNA-seq datasets.

Another research study conducted by Zhang et al. (2014) suggested the use of Cufflinks-Cuffdiff2 (Trapnell *et al.*, 2012), DESeq (Anders and Huber, 2010) and edgeR (Robinson, McCarthy and Smyth, 2010) tools for DE analysis. Using MicroArray Quality Control Project data, K_N RNA-seq and lymphoblastoid cell lines data, benchmarking of edgeR with DESeq and Cuffdiff2 was conducted in which they found better performance of edgeR in terms of its ability to uncover true positives. However, they recommended to involve the intersection of edgeR with two or more tools to obtain true positives and less false positives. They also recommended to include the RNA-seq dataset in research studies that have biological replicates, which further increases the chance of obtaining true positives.

A research study conducted by Klepikova et al. (2015) on *A. thaliana* species from the apical-shoot meristem revealed expression dynamics of major flowering genes in cell-cycle related events. The authors conducted an RNA-seq data analysis, using CLC Genomics Workbench (Sequencing, 2011) as tool for read trimming and genome mapping to TAIR10 genome and DE analysis was conducted using DESeq. For data processing, authors employed the use of default parameter. Through DE analysis of the Shoot Apical Meristems (SAMs), the authors found a number of DE genes during transition from vegetative to inflorescence stage. They also obtained and identified DE genes expressed during the cell division phase in transition to flowering stage, by conducting hierarchical clustering analysis using the R package "fastcluster" (Müllner, 2013). A similar study conducted by Chen et al. (2010) on Arabidopsis male meiocytes, involved use of bioinformatics and statistical analysis pipelines through which reads were aligned to TAIR10 genome using the GSNAP tool (Wu *et al.*, 2016) and *de novo* assembly using ABySS-P (Birol *et al.*, 2009) and SSAKE (Warren *et al.*, 2007). PCAP (Huang and Yang, 2005) was used for assembly merging. Using these methods, several transposable element genes were found to be DE in anthers during meiosis with potential functions identified using Revigo GO analysis toolkit (Du *et al.*, 2010).

Another study conducted by Zhang et al. (2015) involved the identification of genes expressed during meiosis in rice genome. For obtaining the DE genes, the authors removed the adapter sequences and low quality reads prior to aligning raw reads to the rice genome using the SOAPaligner/soap2 tool (Xie *et al.*, 2014b) with two base mismatches. Furthermore, ERANGE software was used for computing the gene expression levels which generated reads per kilo-base per million reads (RPKM) values. Transcriptional gene activity was determined by applying a cutoff of RPKM>0. For gene enrichment analysis of DE genes, Blast2GO (Conesa *et al.*, 2005) was used. Moreover, pathway analysis was also undertaken using reference KEGG database from which well-conserved meiotic genes were identified from RNA-seq data. Another study involving emergence of plant diseases caused due to microbes and parasites studied gene expression of plant *Ocimum basilicum* and its obligate parasite *Peronospora belbahrii* using de novo sequencing assembly tools for identification of virulence and host defense genes during parasitic infection. Due to absence of reference genome, authors proposed a computational pipeline which utilized Trimmomatic (Bolger, Lohse and Usadel, 2014) for adapter and quality trimming and RSEM (Li and Dewey, 2011) for *de-novo* assembly and transcript abundance estimation with default parameters. Using PANTHER tool for GO enrichment, distinct genes were identified suggesting biological functions enriched in transport, localisation, photosynthesis, precursor metabolites generation, energy production, etc.

Many of these RNA-seq studies focus on the identification of DEGs using bioinformatics tools in plant genomes. However, very often these comparative studies fail to consider optimal parameters that are required for upstream processing of the data prior to DEG analysis. Due to this, RNA-seq studies involving plants, sometimes do not generate optimal results. Furthermore, commercial RNA-seq software, such as CLC Genomics fails to consider this aspect, which can lead to unreliable results. Therefore, considering the impact of the parameters on read mapping to identification of DEGs, a standardised computational pipeline is required. Also, previous studies did not consider the impact of DEG intersection approach using multiple methods. Hence, the subsequent study attempts to develop a bioinformatics pipeline for processing and analysis of RNA-seq data in plants which employs species-specific parameters, obtained through experimental results through the intersection of different normalisation methods. The use of optimal parameters not only generates valid results but also ensures and demonstrates a reliable scientific approach that helps in reducing the outcome of false positives during the differential expression analysis.

1.17 Tools for lncRNA identification and annotation

1.17.1 Tools developed for lncRNA identification

With the emergence of NGS technologies, a number of tools have confirmed the presence of lncRNAs in the human genome. Due to their non-conservation of sequence, lncRNAs have become one of the most poorly studied area. A number of studies have demonstrated the critical role of lncRNAs in biological processes and their involvement in diseases. Several databases and tools have been developed for identification and annotation of lncRNAs in the past few years. Most of the computational lncRNA prediction methods are based on machine learning approaches, which include PhyloCSF (Lin, Jungreis and Kellis, 2011), Coding Potential Calculator (CPC) (Kong *et al.*, 2007), Coding Potential Calculator 2 (CPC2) (Kang *et al.*, 2017), Coding-Non-Coding Index (CNCI) (Sun *et al.*, 2013), Coding Potential Assessment Tool (CPAT) (Wang *et al.*, 2013), Predictor of long noncoding RNAs and messenger RNAs based on an improved *k*-mer scheme (PLEK) (Li, Zhang and Zhou, 2014a), lncScore (Zhao *et al.*, 2016a), PLncPRO (Singh *et al.*, 2017), Coding potential calculation tool based on multiple features (COME) (Hu *et al.*, 2016), LncRNA-ID (Achawanantakun *et al.*, 2015), lncRScan-SVM (Sun *et al.*, 2015), lncRNA-MFDL (Fan and Zhang, 2015), LncRNApred (Pian *et al.*, 2016) and DeepLNC (Tripathi *et al.*, 2016).

Based on comparative genomics method, Lin et al. (2011) proposed PhyloCSF which analyzes sequence alignments of nucleotides from multiple species. The authors reformulated the Codon Substitution Frequencies (CSF) metric by implementing the use of multiple alignments in a phylogenetic framework that produces likelihood ratios as output. PhyloCSF assesses coding potential of the individual exons from transcripts and aligns to one or more genomes at certain phylogenetic distances. For the parameter estimation, it also requires the genome of interest to possess good quality gene annotations. For distinguishing coding from non-coding regions, two models are assumed. One representing the evolution of codons in protein-coding genes and another one representing the evolution in nucleotide triplet sites in non-coding regions. Using the alignment, the Maximum Likelihood Estimate (MLE) of coding and non-coding models are determined. Protein-coding or non-coding decision is taken based on the value of log-likelihood ratio.

CPC is a web-server application that is used for assessing the coding potential of a protein using six biological sequence features. The first three features are ORF based features, in which it uses log-odds score and coverage as the first two features. The third feature is the integrity of ORF which indicates the ORF start, end and in-frame stop codon. The next three features are alignment-based features, namely the number of hits of sequence to protein database, the hit score of a sequence with measurements of High-scoring Segment Pairs (HSPs), and the frame score for measuring the distribution of HSPs among 3 open-reading frames. These six features are incorporated in the Support Vector Machine (SVM) classifier implemented in the LIBSVM package (Chang and Lin, 2011) for measuring the classification performance. CPC2 on the other hand, computes the coding probability of the sequence by computing its peptide length, isoelectric point, Fickett score (Fickett, 1982) and ORF integrity. CPC2 employed SVM using RBF kernel for training 17984 protein-coding and 10452 non-coding transcripts from Refseq, Ensembl (v87), and EnsemblPlants (v32) databases. Similar to CPC, PLncPRO is an alignment-based lncRNA prediction tool which derives features from BLASTX tool (O'Donovan *et al.*, 2002) using alignment of the query sequence with the protein-coding sequences deposited in Non-Redundant (NR) database. The tool uses RF for classification of FASTA sequences derived from plants into lncRNA or proteins.

CNCI distinguishes lncRNAs from protein-coding sequences by profiling Adjoining Nucleotide Triplets (ANTs). CNCI constructs an ANT matrix by identifying the Most-Like Coding Sequences (MLCDS) in each transcript sequence, which is calculated in all six reading frames. Using MLCDS, CNCI extracts five features: score-distance, length percentage, S-score, length and codon-bias which are incorporated in the SVM with a standard radial basis kernel function like CPC for classification. CPAT on the other hand, classifies lncRNAs using logistic regression as a classifier by extracting four sequence based features, namely, maximum length of ORF, ORF coverage, Fickett score (Fickett, 1982) and hexamer score (Fickett and Tung, 1992). The Fickett score is used for evaluating the unequal distribution of codons in the sequence, whereas hexamer score is used for measuring the bias in codon usage of adjacent amino acids. PLEK is another alignment-free tool which uses calibrated *k*-mer frequencies of a sequence and sliding window approach as features for classification. However, when compared to CNCI using multiple species, PLEK does not perform well as the algorithm fails to consider insertions and deletions in the sequence when performing classification. Similar to CPC and CNCI, PLEK also uses SVM with radial basis kernel function.

lncScore is another alignment-free tool which also uses logistic regression on 11 sequence-based features namely, hexamer score, hexamer score distance, sequence length, coding score, coding score percentage, Fickett score, hexamer score, ORF length, ORF coverage, and hexamer score distance. These features can also be calculated from the partial length mRNA transcript sequences. The features are calculated from all three frames which are independent of start or stop codons since some of the partial length transcript sequences lack start/stop codons. This affects the protein coding potential computation of ORF based features. Like CPAT, this tool also uses logistic regression for assessing the coding potential of a transcript.

COME is another tool which uses a combination of sequence-based and experiment-based features by employing the decompose-compose method for the construction of features. Unlike other tools, COME constructs features on genome level by indexing genomes and using indexed bins of 100-nt size which overlaps with the exons. These overlapping bins were converted to feature vectors using the mean, maximum and variance for constructing the feature matrix. COME also used expression and histone modification profiles as the experimental features which evaluated the performance using different datasets. For classification of lncRNA sequences, COME uses Balanced Random Forest (BRF). In contrast to COME, LncRNA-ID uses three sets of Feature Groups (FG): ORF-based, ribosome interaction based, and protein conservation based. ORF-based FG included ORF length and ORF coverage whereas ribosome interaction FG included two initiation interaction features: nucleotides at the positions {-3, +4} and {-2, -1}, and two features based on Translation and Termination process: ribosome coverage and Ribosome Release Score (RRS). Using protein conservation, it extracted alignment score, alignment length in the query sequence and alignment length in the HMM profile. Identical to COME, LncRNA-ID also uses BRF for the classification.

lncRScan-SVM classifies transcripts by extracting six features, namely, transcript length, standard deviation of counts of stop-codons between three frames, CDS score, exon length, exon count and sequence conservation using PhastCons scores from UCSC genome browser (Kent *et al.*, 2002). Unlike COME, lncRScan-SVM tested the model performance using GENCODE (Harrow *et al.*, 2012) humans and mice datasets and used SVM for classification. lncRNA-MFDL is using the deep learning approach for classification, which is based on four sequence features: *k*-mer, ORF, MLCDS and secondary structure which are integrated to construct a classification model based on the deep stacking network. LncRNApred on the other hand uses a self-organising map clustering method for selecting samples as a training set. Using the ORF length, ORF coverage, GC content and *k*-mer, it transforms the query sequence to a binary vector in order to construct a signal-to-noise ratio feature which is Fourier transformed using Discrete Fourier Transform (DFT) to obtain a power spectrum curve. Protein-coding sequences are differentiated from lncRNAs using the peak observed at N/3 position in the sequence where N=length of transcript, using 3-periodic property. With 86 features, they classified lncRNAs with 92.9 % accuracy using RF on NONCODE humans and mice datasets. DeepLNC uses deep neural networks and classifies lncRNAs using *k*-mer based 1104 features by calculating the possible combinations of *k*-mers with k=2,3,4 and 5. To achieve a reasonable accuracy, it chooses the best possible combination of features from four sets (i) 2, 3; (ii) 2, 3, 4; (iii) 2, 3, 5; and (iv) 2, 3, 4, 5 by using the Forward selection backward elimination (FBSE) method.

1.17.2 Tools developed for identification and genomic annotation of lncRNA sub-classes

lncRNAs are generally classified into different types which depends on their position in the genome. These can be classified into: (1) Sense-overlapping: lncRNAs overlapping the exons and located on the sense DNA strand, (2) Antisense-overlapping: lncRNAs overlapping the exons but located on opposite DNA strand, (3) Bidirectional: lncRNAs which are oriented head-to-head within 1 kilo basepair distance, (4) Intergenic: lncRNAs transcribed and expressed between two- protein-coding genes. The fifth class is called circular RNA. It is not considered as lncRNA but identified as non-coding RNA which are single-stranded circular molecules that regulate gene expression and have been identified as potential biomarkers of cervical cancer (Qu *et al.*, 2015). For the annotation of lncRNAs, fewer tools and computational methods have been developed. Wucher et al. (2017) developed a computational tool called FEELnc for identification and annotation of lncRNAs using multi *k*-mer frequencies. Based on the predicted lncRNAs, authors classified the transcript sequences into long-intergenic ncRNA (LincRNA), genic-sense and genic-antisense. The classifier module of the FEELnc framework uses a sliding window approach that reports all reference transcripts within the sliding window around the lncRNAs. It further uses a set of rules for sub-classification which depends on the direction (antisense or sense) and interaction type (intergenic or genic). The authors employed the classifier module on the reference human Ensembl v83 dataset and on the dog RNA-seq dataset where it identified and annotated lincRNAs and antisense exonic lncRNAs.

Another research study undertaken by Zhao et al. (2016a) proposed genome-wide identification of lncRNAs in RNA-seq samples obtained from patients with Intervertebral Disc

Degeneration (IDD) and spinal cord injury. The authors employed CPC (Kong *et al.*, 2007), PhyloCSF (Lin, Jungreis and Kellis, 2011) and CPAT (Wang *et al.*, 2013) tools for identification of lncRNAs and further classified the differential expressed lncRNAs.

A third study conducted by Pan et al. (2015) developed a computational framework called PredcircRNA for the identification and classification of circular RNAs from lncRNAs using hybrid features. Using combination of conservation, sequence and graph features from transcript sequences, the authors classified circular RNAs from other lncRNA types (lincRNA, antisense, sense intronic, sense overlapping and processed transcripts) with 77.8% overall accuracy using the Multiple Kernel Learning (MKL) approach. Results of the multi-class classification analysis shows that the classifier can differentiate classes of lncRNAs (antisense, lincRNA, circularRNA and processed transcripts) with 60.4% accuracy.

Current tools and methods enlist alignment-free and alignment-based features but do not provide the significance of the features in the classification process. As it is widely known that lncRNAs exhibit poor sequence conservation and are relatively expressed at lower levels, alignment-based methods such as CPC (Kong *et al.*, 2007), PLncPRO (Singh *et al.*, 2017) and PhyloCSF (Lin, Jungreis and Kellis, 2011) rely on the alignment of the transcript sequence with the reference sequence database and assigns scores for each target sequence; the latter can sometimes become inaccessible and increase the computation times. Additionally, alignment-free methods such as PLEK (Li, Zhang and Zhou, 2014a), LncRNA-MFDL (Fan and Zhang, 2015) and FEELnc (Wucher *et al.*, 2017) heavily rely on computation of *k*-mer frequencies from transcript sequences. Due to this, they demand higher computational resources as well as increased computation times. Also, most current computational prediction methods target at mammalian genomes, and do not work well on plant species. Moreover, current alignment-free methods do not consider the importance of codon-bias features which can potentially impact and improve the classification performance. Currently developed computational methods for classification of different lncRNA classes do not provide reasonable accuracies and often misclassifies the lncRNA sub-class. Currently developed methods for lncRNA sub-classification are based on machine learning based approaches, which heavily rely on the availability of the training set. Due to the unavailability of experimental lncRNA sub-class data in plant genomes, a computational approach is required which can accurately classify the predicted sequences in the absence of training datasets. Also, to address the issue of inaccurate identification of lncRNAs in plant species, a light-weight computational approach is required. Hence, a need for development of an appropriate method arises.

1.18 Selection of optimal features in lncRNA identification

Identification of lncRNAs is primarily conducted using computational approaches which employ the extraction of features for classification and characterisation of lncRNAs and protein-coding genes. However, accurate identification depends on the choice of features selected for classification analysis. In this domain, fewer studies have been performed. With regards to the selection of features for classification of lncRNAs, two major research studies have been conducted. The first relevant approach attempted by Hu et al. (2015) proposed a strategy called RNAfeature for the determination of the essential features which can accurately identify ncRNAs in multiple species. Using this approach, 622 datasets from five species were curated. They calculated expression, TRF (transcription and regulation factors) binding signals, histone modifications from 100 nucleotide genomic bins. These genomic bins were then annotated using gold standard datasets which helped in determining training and testing sets. For the feature selection process, a supervised machine learning framework with cross-validation was employed which implemented Recursive Feature Elimination (RFE) (Granitto *et al.*, 2006) for filtering inessential features. Furthermore, to rigorously eliminate features, Greedy Backward Algorithm (GBA) (Harikumar and Bresler, 1996) was implemented. The final set of the selected features were obtained by intersecting feature sets from multiple species. Using the initial feature set, 15 features were extracted from three species for feature selection. These included protein conservation, DNA sequence conservation, GC content, RNA secondary structure homology, stability, conservation and ORF property. Finally, based on the accuracy, the authors obtained 10 features selected for four species: DNA sequence conservation, GC content, protein sequence conservation, small RNA-seq, ORF property, histone modification signals, poly(A)+ RNA-seq signal and poly(A)- RNA-seq signals.

A study conducted by Ventola et al. (2017) designed a web-based tool for feature selection that included some of the novel feature sets such as nucleotide repeat occurrence in transposable elements. Using different feature selection algorithms, the prediction ability was evaluated by studying humans, zebrafish and mouse genomes. Authors collected ~130 genomic features which were grouped into 5 categories: (1) Basic, (2) ORF metrics, (3) conservation scores, (4) nucleotide arrangements and composition, and (5) novel features based on repeat elements. Authors implemented 11 different feature selection approaches which were classified into (1) Filter-based methods: Wilcoxon-test, Gain Ratio (GR), Information Gain (IG) and RFE (Guyon and Elisseeff, 2003). (2) Wrapper-based methods: Greedy Forward Selection (GFS) (Zhang, 2008) and RFE with SVM (Guyon and Elisseeff, 2003), (3) Embedded methods: Elastic net (Zou and Hastie, 2005), Lasso regression (Tibshirani, 1996) and RF, (4) Ensemble methods: which merges the outcome of different algorithms by computing the score for each feature. By employing this methodology for feature selection, the authors evaluated the stability of feature selection and identified a signature set of features. These features were selected based on the intersection of the results from the feature selection approaches. Due to the instability in consistently obtaining features, some of the algorithms were discarded. In comparison with other tools, the authors obtained ~21-24% increase in accuracy.

Currently developed computational approaches for selection of optimal features however pose many drawbacks. Firstly, wrapper-based FS methods such as SVM-RFE are computationally inefficient and fail to identify optimal feature subsets. Whereas filter-based FS methods, such as IG and GR assign relevance score or rank to each feature by considering each feature separately and ignoring any dependencies between features which lead to a worse classification performance. Regression based approaches utilized by Ventola et al. (2017) employed the elastic net method for feature selection. Elastic net uses a combination of $\ell 1$ and $\ell 2$ regularisations. Usage of ℓq norm (with q < 1 or q > 1) approaches for optimisation are generally non-convex and make the minimisation computationally challenging. Additionally, elastic net regularisation is meant to be used for solving problems with higher number of features (p) and less number of variables (n). All research studies conducted employ a greater number of variables with several thousand sequences and fewer features. Under such circumstances, the method may fail to generate a reliable set of features. Moreover, previous attempts for development of FS methods focused mainly on mammalian datasets which can potentially bias the analysis. Therefore, considering the potential drawbacks from the literature reviewed, the development of a potential feature selection approach has been undertaken in this research work. Unlike previous work which measures the stability of different FS methods, the research approach employed in this project implements regression-based approach and compares the performance of the developed approach with other methods which fails to provide reasonable set of features.

1.19 Tools for predicting lncRNA-protein interactions

lncRNA-protein interactions are essential for understanding important biological processes. These interactions play a major role in splicing, post-translational gene regulation, signaling, translation and in the progression of many complex diseases. Thus, identifying these interactions is critical for gaining insights into diverse functions and molecular mechanisms of lncRNAs. Since experimental methods for the detection of lncRNA-protein interaction is time consuming, several computational approaches have been proposed. Bellucci et al. (2011) proposed CatRAPID in which pairs of lncRNA and proteins are encoded into feature vectors and are scored using matrix computation. Similarly, the RPIseq method was proposed, which implemented RF and SVM classifiers for the prediction of lncRNA-protein interaction that exploited sequence information of lncRNAs (Muppirala, Honavar and Dobbs, 2011).

Surest et al. (2015) proposed RPI-pred in 2015 by developing a computational approach for identifying binding partners of RNA-protein interaction pairs. Using 16 structural fragments which they called Protein Blocks (PBs), an accurate representation of protein structures was made. Using experimentally verified PDB structures of RNA and protein from Protein Data Bank (PDB), a training set was created. Using higher-order structures of RNAs and PBs, a SVM classifier could be applied on a query set for predicting RNA-protein interactions.

Li et al. (2015) developed a network-based approach called lncRNA-protein interaction prediction based on Heterogenous Network Model (LPIHN) in which a heterogenous network was constructed using Protein-Protein Interactions (PPI), known lncRNA-protein interactions and expression similarity of lncRNAs. Random Walk with Restart (RWR) approach was then applied on the heterogenous network for elucidating novel lncRNA-protein interactions. Based on a similar, approach Ge et al. (2016) proposed the lncRNA-protein bipartite network inference (LPBNI) method, which is using lncRNA-protein bipartite network. The propagation process in LPBNI is derived from recommendation algorithms (Zhou *et al.*, 2007) which use known interactions of lncRNA and proteins. On the other hand, Hu et al. (2017) proposed an eigenvalue transformation-based semi-supervised link prediction called LPI-ETSLP, for identifying relationships between proteins and lncRNAs. The advantage of this approach is that it does not need any negative samples for the prediction during the classification process. Using this approach, they achieved an AUC score of 0.8876.

Using the heterogenous network model, Xiao et al. (2017) proposed the PLPIHS method which uses the HeteSim measure for computing the relatedness of lncRNA-protein pairs in the heterogenous network. Identical to LPIHN, the heterogenous network is made up of lncRNA-protein association network, PPI network, lncRNA-lncRNA similarity network. Using HeteSim scores, SVM is used for predicting lncRNA-protein interactions. The HeteSim is a path-constrained measure which computes the relatedness of objects of different or similar types in a uniform framework. Using a transition probability matrix, the similarity of lncRNA and proteins is calculated and HeteSim score is assigned between to the lncRNA and protein pair.

Identical to LPI-ETSLP method, Liu et al. (2017) proposed a matrix factorisation computational method for determining lncRNA-protein interactions; this is a semi-supervised approach and does not need negative samples for prediction, as it deduces the interactions mainly based on similarities and their known interactions. The method uses the neighborhood regularized logistic matrix factorisation approach thereafter called LPI-NRLMF method. The method combines the similarity of the modified matrix with the Gaussian interaction profile for achieving accuracy in prediction. The method focusses on the prediction of the probability of association of lncRNA with protein by mapping protein and lncRNA to low dimensional space. Moreover, the local structure of data association was also studied for achieving a higher accuracy, which exploited the influence of neighbors of the most similar proteins and lncRNAs. Using leave-one-out-cross validation, the LPI-NRLMF method achieved an AUC score of 0.9025 with significant improvement in the prediction performance over previous prediction models. Development of LPI-NRLMF method was based on original implementation of NRLMF method proposed by Liu et al. (2016). The method was developed for prediction of drug-target interactions using logistic matrix factorisation.

Currently developed approaches for prediction of lncRNA-protein interactions utilises both network-based and structure-based approaches, which partially depends on the availability of known lncRNA-protein interactions. Known interaction data of lncRNAs and mRNAs is currently available for human and mouse genome, however no interactions have been reported in plant species. Unavailability of lncRNA-protein interactions significantly limits the prediction of potential functions. Therefore, since current tools and techniques have primarily focused on the identification of lncRNA-mRNA interactions in mammalian genomes, less attention has been given on function prediction based on computational prediction of interactions in plant genomes. Moreover, from the literature reviewed, many studies have confirmed that lncRNAs tend to co-express with mRNAs (Guo *et al.*, 2015; Sun *et al.*, 2016; Wang *et al.*, 2017). Therefore, in this study, a combinatorial approach for function prediction using NRLMF and co-expression of genes has been devised for predicting novel interactions.

1.20 Tools for lncRNA function prediction

1.20.1 lncRNA function prediction

Over the past few years, sequencing approaches have revealed the transcriptional complexity of genomes. Through RNA sequencing methods and expression microarrays, there has been an increase in the number of lncRNAs which now exceeds protein-coding genes. Despite of such an enormous catalogue of lncRNA sequences, only a small number of lncRNAs have known functions. Currently developed experimental investigations have provided insights into functions of lncRNAs, however, the majority of lncRNAs still remains functionally uncharacterised. Some of the known functionally characterised lncRNAs include HOTAIR (Hajjari and Salavaty, 2015), XIST (McHugh *et al.*, 2015), COLDAIR (Kim, Xi and Sung, 2017), and H19 (Zhang *et al.*, 2017) which illustrate their potential involvement in protein and gene expression. With the growing need for identification of the lncRNA function, several computational techniques have been developed for imputation of lncRNA function. These include: (1) Differential expression, (2) Guilt-by-Association, (3) Condition-specific expression, (4) Disease association, (5) Conservation, (6) lncRNA-protein interactions. One of the easiest ways of inferring functions is through differential expression analysis; however, DE does not alone provide functional insights. Alternative methods such Guilt-by-Association are needed for exploiting the biological network of genes and their regulation.

Langfelder and Horvath (2008) developed an R package for the imputation of the lncRNA function through Weighted Gene Co-expression Network Analysis (WGCNA), which relies on correlation networks of genes across microarray experimental samples by finding clusters of highly correlated genes. WGCNA has been successfully applied on cancer, yeast genetics, mouse genetics and in the analysis of brain imaging data. Another method developed by Xiao et al. (2015) relies on the prediction of the lncRNA function, based on Bayesian networks. Using Bayesian networks, dependency relationships of lncRNA and proteins was built. Using lncRNA-protein interaction network, lncRNAs connected to protein-coding genes in the network were eventually used to infer functions of corresponding lncRNAs. Through this approach, 762 lncRNAs were allocated to functions and were found to be involved in embryo development and tissue development in 58 prostate cancer samples. Identical to WGCNA, Yao et al. (2015) implemented co-expression networks for identifying enhancer RNAs (eRNAs) in the human brain, by constructing an eRNA-protein gene interaction network across fetal brain and multiple adult brain regions. Through this, they found eRNA association in autism.

Zhou et al. (2015) proposed a novel rank-based approach for disease association analysis, called RWRHLD which implements Random-Walk with Restart (RWR) on Heterogenous lncRNA and Disease networks. They constructed lncRNA-lncRNA networks by examining the co-occurrence of shared miRNA response elements on the transcripts of lncRNAs, disease-disease similarity network and known lncRNA disease association networks. They integrated all these networks to construct a heterogenous network and implemented the RWR on this heterogenous network to impute the association of lncRNAs in diseases. Disease association analysis has also been performed by constructing a functional similarity network using information from miRNA (Chen, 2015). lncRNA-disease association was predicted by integrating lncRNA-miRNA interaction and miRNA-disease association information to construct hyper geometric distribution of lncRNA-disease association inference (HGLDA). Colorectal, breast and lung cancer samples were used for lncRNA-disease association prediction. By integrating disease semantic similarity using direct acyclic graphs and MeSH descriptors, lncRNA functional similarity based on lncRNA-miRNA interactions and miRNA functional similarity, functions were associated to lncRNAs. The miRNA functional similarity was computed based on the miRNA-disease association and disease semantic similarity.

Chen et al. (2013) proposed LRLSLDA, a semi-supervised learning approach for lncRNA-disease association by integrating phenome-lncRNAome network which was acquired from LncRNADisease database, lncRNA similarity network and disease similarity network. The method assumed that similar diseases interact with similar lncRNAs. Wang et al. (2016a) proposed LncDisease, an improvement over LncRNADisease database, by predicting the lncRNA association with hypertension and breast cancer. For the prediction of lncRNA-miRNA interactions, miRanda (Betel *et al.*, 2008) and TargetScan (Friedman *et al.*, 2009) were used, whereas for the prediction of lncRNA-disease association, the TAM method (Lu *et al.*, 2010) was used; the latter uses disease associated miRNAs as its input from the HMDD database (Li *et al.*, 2014b) and enrichment analysis is performed which outputs the significance of the miRNAs predicted in each of the disease-associated miRNA set. Identical to the RWRHLD method, Sun et al (2014) proposed a network-based method called RWRlncD, which integrated the known lncRNA-disease association, disease similarity networks and functional networks of lncRNAs.

Based on the correlation of lncRNAs and protein-coding genes, Jiang et al. (2015) developed a comprehensive web-based resource, called LncRNA2Function, consisting of functional association of 9625 human lncRNAs with biological pathways and GO terms. Using RNA-seq data from 19 human normal tissues and annotation information of lncRNA and protein from the GENCODE database, expression values were computed using Cufflinks. Through these expression values, Pearson Correlation Coefficient and significantly expressed lncRNAs and mRNAs were calculated with PCC > 0.9 and adjusted P-value < 0.05. Using correlation, GO annotation and pathway annotation data of protein-coding genes, lncRNAs were annotated. In their work, Perron et al. (2017a) suggested a similar approach based on correlation analysis. by calculating the co-expression for 9 vertebrates and 30 human tissues using a rank product algorithm. They calculated a functional prediction score from a set of RNA-seq samples and quantified the gene expression of each sample. Tissue-specific and phylogenetic conserved gene expression was evaluated in 10 mammalian species and 8 organs which were published by Necsulea et al. (2014). From this dataset, they profiled 5400 lncRNAs and 22000 mRNAs. They also collected tissue-specific expression of genes from 2923 samples distributed across 30 tissues from which 7000 lncRNAs and 19500 mRNAs were profiles. lncRNAs were functionally annotated by assigning Gene Ontology (GO) terms assigned to protein-coding genes. Through this analysis, they found several lncRNAs *PTENP1*, *BRAFP1*, *TUSC7* and *MYCNUT* predicted to be involved in cancer.

Guo et al. (2013) proposed a novel network based approach called bi-coloured network, which integrates protein-interaction and gene expression data. The lncRNA global function predictor (lnc-GFP) method is a bi-coloured network which uses coding-noncoding expression data and protein interaction data. Using this method, functions for 1625 lncRNAs were assigned from a total of 1713 lncRNAs. By constructing the network, 87874 edges were determined having 29393 mRNA-mRNA interactions, 59173 co-expression and 692 both mRNA-mRNA and co-expression. Through this analysis, 1625 lncRNAs were found to be associated with 5284 GO terms.

Previous work on lncRNA function prediction included mapping of long-intervening ncRNAs to chromatin states, through which the prediction function was assigned (Guttman *et al.*, 2009). Khalil et al. (2009) used the same strategy and identified ~3300 long-intervening ncRNAs in six human cell types and examined association between long-intervening ncRNAs and PRC2 complex. Identical LncRNA2Function, Cabili et al. (2011) defined a catalogue of more than 8000 long-intervening ncRNAs and characterizing them functionally through co-expression between non-coding and protein-coding genes.

Apart from the functional identification of lncRNAs in mammals, a few studies have been performed which included functional characterisation of lncRNAs in stress drought plant (Li *et al.*, 2017). Li et al. (2017) performed a co-expression study on cold and drought stress affected *Manihot esculenta* (cassava) plant for screening and identifying functions of lncRNAs under stress and drought conditions. They used a strand-specific RNA-seq approach for investigating genome-wide transcriptome reconfiguration of *Manihot esculenta*. Using 9 samples through whole-transcriptome ssRNA-seq, 453 lincRNAs and 229 lncNATs were identified using CPC, CPAT and CNCI as lncRNA predictors. For identifying functions of stress-responsive lncRNAs, co-expression analysis was performed to identify trans-regulatory networks. 45 GO terms were associated to stress-responsive lncRNAs.

Functional prediction of lncRNAs was also performed lncRNAs using gene expression microarrays. Zhang et al. (2016) obtained 481 DE lncRNAs from tumorous and normal tissue samples of nonkeratinizing carcinoma (NKC). Through co-expression network, transcription factor binding motif, interactive miRNAs and gene ontology analysis, functional prediction was performed for inferring lncRNA functions in NKC.

1.20.2 Protein function prediction

Although most of the known protein-coding genes have associated functions, several proteins remain functionally uncharacterised. Certain *in-silico* approaches have been conducted for associating functions to these proteins. Using neural networks, Rifaioglu et al. (2017) developed a multi-task deep neural network architecture using GO terms called DEEPred, for protein function prediction. The DEEPred implements post-processing of prediction based on GO direct acyclic graphs. Using the subsequent profile map (SPMap), feature vectors of protein sequences are generated, which are then clustered together based on the BLOSSUM-62 matrix. The clusters are transformed into probabilistic profiles, where each GO term is assigned to an individual profile which is classified by the classifier in order to assign a function. Another approach using Multimodal Deep Autoencoders (MDA) was also developed using a network fusion method called deepNF (Gligorijević, Barot and Bonneau, 2018). The method implemented the RWR approach for vector representation and Positive Pointwise Mutual Information (PPMI) approach for constructing the matrix by capturing structural information of network. PPMI matrices were fused using MDA and then predicted lncRNA functions using SVM classifier.

Studies have also been conducted for predicting functions by identifying DNA and RNA-binding proteins using machine learning RF models (Peled *et al.*, 2016). Nucleic Acid (NA) binding proteins were predicted based on the assumption that the distribution of the predicted binding site differentiates protein which binds NA more accurately than proteins which do not bind NA. Certain studies have also been performed for protein function prediction based on sequence, structure and protein-protein interaction information. The COFACTOR web server uses hybrid models which combine information from sequence and structure homologies, and protein-protein interaction networks for the protein function prediction. COFACTOR implements sequence-based, structure-based and PPI-based pipelines for inferring GO function prediction which is given by the confidence score (Zhang, Freddolino and Zhang, 2017). Delattre et al. (2016) implemented a distance homology search approach for constructing a tool called Phagonaute based on Hidden Markov Models (HMM) for > 80,000 proteins derived from phages and archaeal viruses and performed pairwise comparisons. Using this approach, the function of unknown phage protein can be inferred.

Like lncRNA function prediction, network-based approaches have also been used for function prediction of proteins. Sharan et al. used direct and indirect methods for function prediction (Sharan, Ulitsky and Shamir, 2007). Direct methods involve function assignment of an unknown protein, when the unknown protein interacts with a known protein having a function, whereas indirect methods involve the identification of functional modules in the network. The overrepresented or enriched functions in these modules are used for annotating the unannotated proteins in the network. Deng et al. (2002) developed a direct method based on a probabilistic approach called Markov Random Fields (MRF). MRF method says that the function of protein ideally depends on two conditions: (1) Its direct interaction with neighbouring proteins having associated function and (2) interaction with those that do not perform function. Using logistic regression (Nelder and Wedderburn, 1972), the parameters of the relationships can be known and learned from the training set. Gibb's sampling is then implemented for determining the functions of proteins with unknown functions. Lee et al. (2006b) combined the properties of MRF and SVM to generate a Kernel Logistic Regression (KLR) approach in which parameter estimation and predictions could be performed much faster. The use of the diffusion kernel for parameter estimation outperformed MRF and SVM when several experiments were carried out using *Mus musculus* datasets for functional inference.

An improvement of MRF method was proposed by Gehrmann et al. (2013) called Conditional Random Fields (CRF), which removed the requirement of modeling relationships between various data sources, thus providing substantial improvement over the data derived from the genetic interaction networks. Using a network-based approach, Mostafavi et al. (2008) developed a fast heuristic algorithm, using ridge regression, by integrating multiple functional association networks for predicting protein functions. Kourmpetis et al. (2010) discovered a potential problem in the parameter estimation step in the MRF approach which could be troublesome when annotated proteins are connected with unannotated proteins as neighbors.

Therefore, the authors revised the MRF method by implementing a Joint Parameter Estimation (JPE) and prediction step with moderate computational cost. JPE on missing datasets performs iterative estimation of parameters using logistic regression in the first step and then unknown function is estimated by optimizing the objective function till convergence. They named this method Bayesian Markov Random Fields (BMRF) (Kourmpetis *et al.*, 2010). Using the BMRF approach, it outperformed MRF and KLR methods on function prediction when tested on 1170 *Saccharomyces cerevisiae* unannotated proteins.

Recent advances in lncRNA function prediction primarily focus on mammalian datasets where genome annotation and co-expression data are easily available (Jiang *et al.*, 2015; Xiao *et al.*, 2015; Perron, Provero and Molineris, 2017). Therefore, less attention has been paid on functional prediction on non-model plant RNA-seq datasets. Development and application of current methods for function prediction has primarily focused on the annotation of lncRNAs involved in diseases such as breast cancer. Development of such methods significantly biases the analysis as they are mostly used for identifying disease-related functions of lncRNA sequences. Therefore, functional roles unrelated to diseases becomes difficult to impute. Other methods, such as RPI-pred (Suresh *et al.*, 2015b) rely on the experimental structure of lncRNAs and proteins for the prediction of novel binding partners. Due to limited availability of the experimental structures and their fixed binding interactions with protein-coding genes, the identification process becomes restricted and computationally resource intensive. Therefore, the work in this thesis attempts to overcome the above-mentioned drawbacks, by employing the Bayesian approach for identification of lncRNA functions in plant RNA-seq data.

1.21 lncRNA visualisation tools

With the increase in the size of data generated by high-throughput sequencing experiments visualisation tools are required to visualise, analyse and interpret these datasets. A genome consists of a vast amount of Information about various genes, transcripts, mutations, substitutions, inversions, translations, structural variations, length of sequence, gaps in sequence, open reading frames in the sequence, etc. A genome browser visually conveys this information as well as the spatial relationship between different bits of sequence data in the genome.

A genome browser helps to visually compare and correlate information from different sources and provide an informative and comprehensive graphical representation of the data. With the absence of web-based bioinformatic applications the enormous amount of data generated by HTS machines could not be shared or processed and visualised. With development and publication of the first web-based genome browser (UCSC human genome browser) 15 years ago, the visualisation of human genome was achieved, which provided significant details about characteristics and limitless information of the genome in detail (Kent *et al.*, 2002). This encouraged the development of web-based visualisation tools, not only for visualisation, but also for data processing and analysis as well. Many web-based genome browsers have been developed for the visualisation of genome which includes the UCSC genome browser, JBrowse (Skinner *et al.*, 2009), Ensembl genome browser (Fernández-Suárez and Schuster, 2010), Integrated Genome Browser (IGB) (Freese, Norris and Loraine, 2016), pileup.js browser, which enable the visualisation of large genomic sequences (Vanderkam *et al.*, 2016). The recent development of web-based data processing, analysis and visualisation of genomic sequences has filled the gap of sequence generation and data interpretation on the web.

Like lncRNA identification and function prediction, several studies have been reported for visualisation of lncRNAs in genomic datasets. Gong et al. (2017) developed a comprehensive workflow called lncRNA-screen for computational evaluation of lncRNA transcripts from large multi-modal datasets. The pipeline provides RNA-seq alignment, transcript assembly, assessment of quality, filtering transcripts, lncRNA identification, estimation and quantification of transcript levels, histone enrichment profile integration, DE analysis, annotation and visualisation. The visualisation component consists of interactive report showing genomic snapshots of mRNA-lncRNA interactions based on Hi-C data. Since visualisation based on sequence conservation of lncRNAs generates false positive results, publicly available ChIP-Seq, CAGE-Seq and DNase-Seq databases can be used for providing improved precision in the visualisation. Avila Cobos et al. (2017) developed Zipper plot which uses the genomic coordinates of transcriptional start sites (TSS) of lncRNAs and produces a summary table with statistics which was implemented using jQuery, HTML5 and PHP. Volders et al. (2013) developed a web-based database called LNCipedia consisting of annotated lncRNAs derived from *Homo sapiens*, which allows the user to query and download sequences and structures of lncRNAs. Using Perl, it also allows visualisation and querying of data.

Although limited visualisation tools are available for lncRNAs, many tools and applications have been developed for the visualisation of RNA-seq data. Thorvaldsdóttir (2013) developed a desktop application called Integrated Genomic Viewer (IGV) for NGS data visualisation. Using the data tiling approach, originally developed by Google Inc. (Google.com, 2014), interactive exploration of large scale genomic data can be achieved on a standard desktop computer. By dividing the genome into tiles which corresponds to viewable region on the screen, an increase in zoom proportionately increases the tiles of the chromosome, which corresponds to the screen pixel displayed at that resolution. The application further optimises the computational usage by removing tiles which are no longer needed to support current view, thereby providing browsing on all resolution with minimal memory. Certain tools used for DE analysis of RNA-seq data also provide visualisation of transcripts.

Anders and Huber (2014) developed DESeq2 which provides DE analysis as well as visualisation and gene ranking which is based on the stable estimation of Logarithmic Fold Change sizes (LFCs). For visualisation of RNA-seq data, the count data should be transformed using either log transformation of variance stabilising transformation. Log-transformed or variance stabilised count data can then be used for visualisation as heat map of raw and transformed data, sample to sample distances using Euclidean distance and principal component analysis plot of the samples. Another tool called ngs.plot and developed by Shen et al. (2014) utilises and integrates the information from genomic databases to provide genomic visualisation of enrichment patterns of DNA-interacting proteins. This is achieved by collecting and retrieving functional elements from publicly available datasets and plotting them using the R tool (R Development Core Team, 2016). Ngs.plot selects the region of interest and uses the genome crawler which grabs genomic annotation from databases and packs the information into an archive. The information is used by the script that calculates and visually inspects the correlation among the samples. This is then plotted using R graphical functions. It produces two plots which provide an average profile of the mean of all regions and a heatmap showing the enrichment of the region across genome.

A similar R package has also been developed, called Scater (McCarthy *et al.*, 2017), which provides data pre-processing, quality control, normalisation and visualisation of single cell RNA-seq data. For visualisation of scRNA-seq data, it provides functions such as plotPCA for performing and visualising principal component analysis, plotTNSE function for performing t-distributed stochastic neighbour embedding, plotMDS for generating multi-dimensional scaling plots and plotDiffusionMap for generating diffusion map of differential processes.

For NGS data visualisation based on web technology, the BrowserGenome tool was developed by Schmid-Burgk and Hornung (2015) for data analysis and visualisation of RNA-seq data. BrowserGenome is mainly focused on the analysis of mRNA-seq data and provides a circular representation of mapped density using FASTQ data. User interface is developed based on principles of Google Maps (Google.com, 2014) such that exons and gene names can be displayed with higher zoom levels. Some studies were also performed for visualisation of RNA-seq data in three dimensions, which was accomplished by Shifman et al. (2016) by developing the Cascade tool, which provides a 3D visualisation of cancer RNA-seq data. Using analysed data from RNA-seq, it can be mapped onto the biological pathways defined by the users. For interactive 3D representation, it implements three.js library which generates a "hair-ball" network style diagram, where gene names are represented as nodes and are connected to each other by edges/lines with concentric rings as the representation of depth in the pathway. Cascade uses MySQL database for storing pathway information, gene expression, copy number variants information, mutations and alternative splicing information, and gene lists. This information is retrieved using PHP scripts and the gene pathway network is represented using three.js.

Currently developed visualisation tools have primarily focused on providing improved analysis and visualisation of RNA-seq data. However, less effort has been put on the visualisation of lncRNAs, its sub-classes and its function in the genome. Therefore, the present study attempts to develop a visualisation application using a combination of statistical-based and Javascript-based approach, which is expected to provide a comprehensive view of the genome that can display annotated lncRNAs and its sub-classes and also annotated with function predicted using the Bayesian approach.

1.22 Summary

This chapter introduced the background to various concepts and a review of relevant work has been carried out and reported in this chapter. The general background of RNA-seq data analysis consisted of two sub-sections. In the first sub-section, various tools developed for analysis of raw RNA-seq data have been listed. In the second sub-section, several research studies employing these tools for identification and analysis DE genes in plant genomes were reviewed. Several tools developed for prediction and identification of lncRNAs from FASTA sequences and RNA-seq datasets as well as advancements in lncRNA sub-classification were discussed which provided a comprehensive understanding of the technical developments in the field of computational biology. Current developments in the prediction of lncRNA-protein interactions using *in-silico* approaches in which tools developed from 2011 to 2018 have been reviewed and discussed as well. Furthermore, the chapter discussed several computational tools developed for function prediction of lncRNAs. Since this thesis adopts a novel probabilistic computational approach for the prediction of the lncRNA function, this section also attempted to present the tools that have been developed for protein function prediction using evolutionary sequence-based and statistical methods. Several computational technological developments in terms of the visualisation of lncRNAs were reviewed. Due to the limited availability of tools for lncRNA visualisation, current and previous research on genomic visualisation of RNA-seq data has also been discussed, as this thesis has adopted a methodology for the development of web-based visualisation of lncRNAs from RNA-seq data. This section discussed and enlisted various shortcomings/demerits in previous work and demonstrated potential gaps in lncRNA sequence analysis. This has motivated the author to undertake the research work. The development of a novel computational approach, which to the author's knowledge has not been carried out before, is presented in this thesis.

Chapter 2 is intended to provide detailed description of the methods used and algorithms developed to fulfill the research objectives. The methods discussed in the forthcoming chapter addresses the limitations discussed and provides a computational framework for identification, classification and function prediction of lncRNA sequences in the plant species.

CHAPTER 2: RESEARCH METHODOLOGY

2.1 Introduction

The previous chapter surveyed tools and research studies for RNA-seq data analysis approaches from a wide range of algorithm developments to implementation and application of developed methods for construction of analysis pipelines. We comprehensively reviewed tools developed for identification, classification and prediction of long non-coding RNAs (lncRNAs) from web-based genomic databases using machine learning approaches. Various features used in lncRNA identification and their performance were also reviewed. Later, several developed computational approaches for identification of lncRNA-protein interactions as well as previous and current developments in the function prediction of lncRNAs and proteins were discussed. Several computational developments in visualisation of lncRNAs and genomic visualisation of RNA-seq datasets using desktop-based and web-based methods were discussed.

This chapter provides a review of the key ideas and detailed implementation of topics discussed in the literature review chapter. In this chapter, the methodology and its characteristics for enhancing the identification and function prediction of lncRNAs in RNA-seq datasets has been described.

This chapter is organized in 12 sections. Section 2.2 explores processing and analysis of RNA-seq data using computational and statistical methods. Section 2.3 provides the datasets used in this research study. In Section 2.4, the computational pipeline and workflow for performing lncRNA identification, classification and prediction is presented. The features used for classification and identification of lncRNAs is discussed in Section 2.5. In Section 2.6, the methodology of the steps required for feature extraction from RNA-seq datasets is presented. Section 2.7 reviews the iterative random forest classifier method used for classification. The detailed implementation of the classifier for identification and differentiation of lncRNAs from coding sequences is provided in Section 2.8. Section 2.9 discusses detailed implementation of the optimisation method coupled with the classifier on RNA-seq datasets. Sections 2.10 and 2.11 presents the details of performance evaluation methods using cross validation approach. The methodology for classification of lncRNA sequences based on genomic position is presented in Section 2.12. Section 2.13 outlines the implementation of function prediction approach for lncRNA sequences. The methodology for web-based visualisation of lncRNAs is provided in Section 2.14. A summary of this chapter is discussed in Section 2.15.

2.2 RNA-seq data analysis

2.2.1 RNA-seq datasets collection

Two RNA-seq datasets were used for the identification of DE genes. The first dataset consists of 10 samples derived from the apical shoot meristem time-series dataset from *A. thaliana* genome obtained from the NCBI SRA database (Project ID: PRJNA268115) (A. V Klepikova *et al.*, 2015). This consists of 10 samples from Day-7 to 16 with two replicates from 9-14 days. The samples have been denoted as S1, S2, …, S16. 9 sample pairs were constructed by comparing samples from Day 8-16 against Day-7. The second dataset consisted of 11 time-series samples (from 0 to 20 days) from whole seed of *Z. mays* inbred line B73 which was obtained from SRA database (Project ID: SRP037559) (Chen *et al.*, 2014).

2.2.2 Data processing workflow

For the identification of DE genes from RNA-seq datasets, a computational pipeline was constructed (Figure 2.1) with customised parameters for reference-based RNA-seq datasets. The pipeline starts with the conversion of raw sequence reads from SRA format to FASTQ format using SRA toolkit (Ostell and McEntyre, 2007) as FASTQ files were needed for sequence alignment. In the next step, a quality metric report was generated using the FastQC tool (http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/) which briefly outlines the metrics of sequence quality, quality scores, sequence content, sequence length distribution, sequence duplication levels, overrepresented sequences, adapter content and Kmer content. Based on the metrics, reads were trimmed to generate trimmed read files for each sample using Cutadapt (Martin, 2011). Following read trimming, samples were again checked for contaminated sequences, adapters, and poor-quality reads using the FastQC tool so that they could be removed before alignment in the next step.

Individual sample reads were aligned to the genome using TopHat2 (Kim *et al.*, 2013) which is a fast splice junction mapper based on Bowtie2 (Langmead and Salzberg, 2012). Cufflinks and Cuffmerge (Trapnell *et al.*, 2012) were then used for transcript assembly and transcript merging. Differential Gene Expression (DGE) was performed using Cuffdiff (Trapnell *et al.*, 2012). For reducing the chances of obtaining false positives and increasing true positives from data, DESeq (Anders and Huber, 2010) and edgeR (Robinson, McCarthy and Smyth, 2010) tools were also employed for DGE analysis. Using DESeq and edgeR, BAM files obtained from Tophat2 are converted to raw read counts using the HTSeq (Anders, Pyl and Huber, 2015) tool. There are many other transcript quantification tools available, such as RSEM (Li and

Dewey, 2011) and StringTie (Pertea et al., 2015), which utilize the BAM file generated from Tophat2 and reference GTF file and produce Reads Per Kilobase of transcript per Million (RPKM) reads. HTSeq utilizes a simpler approach and produces raw read count from the SAM file and reference GTF file.

Raw reads were then used for DGE using DESeq (Anders and Huber, 2010) and edgeR (Robinson, McCarthy and Smyth, 2010). There are many other tools available for RNA-seq DGE analysis however Cuffdiff was chosen as it is specifically designed for DGE analysis from transcripts, spliced regions and promoters, and is best suited to use in conjunction with TopHat2. Another advantage of employing DESeq and edgeR in DE analysis is that both tools are designed to work with and without replicates, which provides additional validity in the approach. Coupling HTSeq with DESeq and edgeR helps in direct integration of raw read counts from htseq-count as input into DESeq and edgeR programs.

Post-analysis was performed using the SpliceR (Vitting-Seerup, B. T. Porse, *et al.*, 2014) tool for annotation of transcript features obtained from Cuffdiff. Results from Cuffdiff, DESeq and edgeR were merged to obtain collective DEGs in the sample pairs. The final step of the pipeline consisted of gene enrichment, pathway analysis and protein-protein interaction (PPI) network analysis using the Araport portal (Krishnakumar *et al.*, 2015), ClueGO (Bindea *et al.*, 2009) and GeneMania (Montojo *et al.*, 2010) for identifying novel gene clusters associated with flower development.

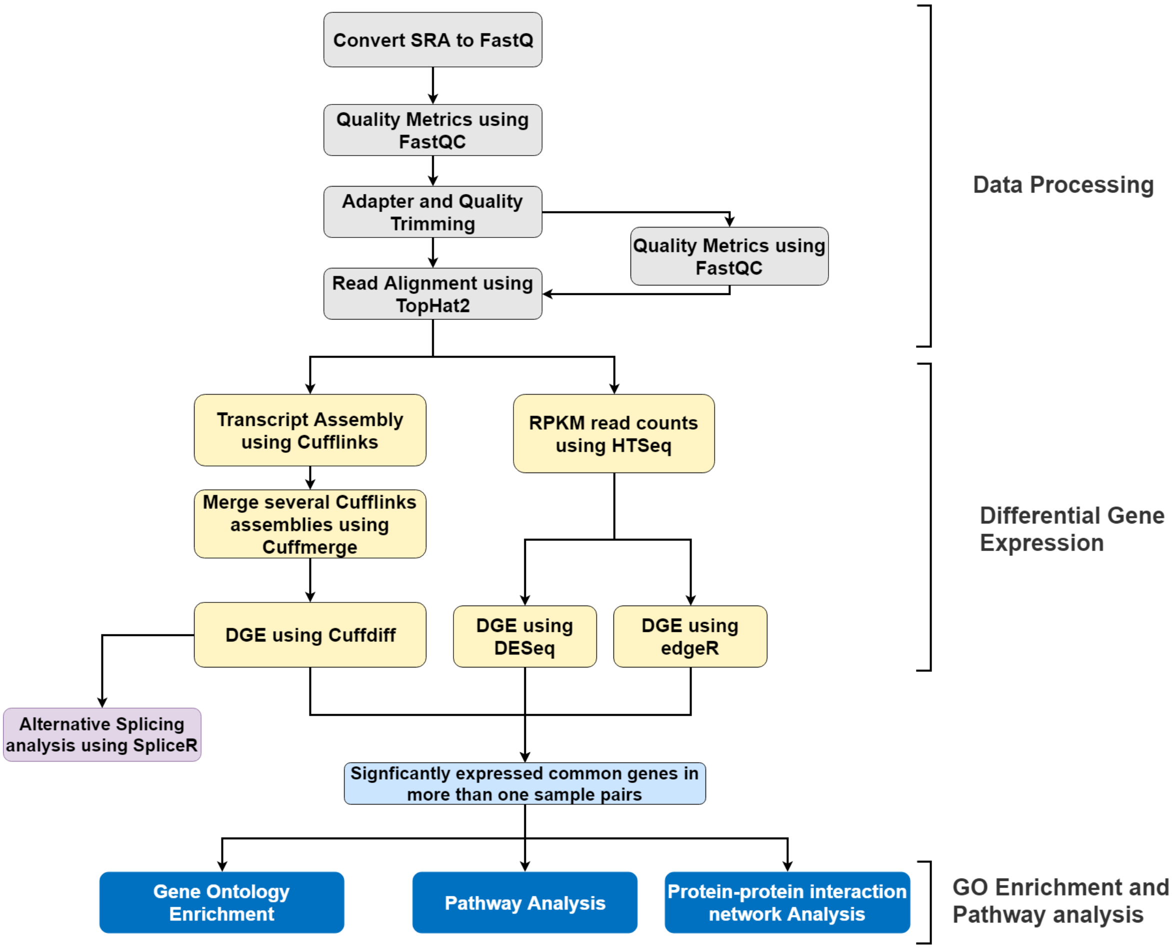

Convert SRA to FastQ
Quality Metrics using FastQC
Adapter and Quality Trimming
Quality Metrics using FastQC
Read Alignment using TopHat2
Data Processing
Transcript Assembly using Cufflinks
RPKM read counts using HTSeq
Merge several Cufflinks assemblies using Cuffmerge
Differential Gene Expression
DGE using Cuffdiff
DGE using DESeq
DGE using edgeR
Alternative Splicing analysis using SpliceR
Signficantly expressed common genes in more than one sample pairs
Gene Ontology Enrichment
Pathway Analysis
Protein-protein interaction network Analysis
GO Enrichment and Pathway analysis

Figure 2.1: Flowchart of the proposed RNA-seq data analysis pipeline. The workflow is divided into three stages namely, data processing, DGE and GO enrichment & network interaction analysis.

2.2.3 Read trimming, reference genome mapping and transcript assembly

Adapter trimming and genome mapping are represented in the pre-processing step, as seen in Figure 2.1. The first 15 base pairs of the reads were trimmed using Cutadapt to remove the adapter sequences. Adapter trimming retains only high quality reads with a quality score (Q-score) greater than or equal to 30 (Martin, 2011). Each sample consists of two reads: therefore, each read was trimmed and a FastQC report was regenerated on the trimmed data to examine the quality and verify that the resulting reads satisfied the criterion.

The *A. thaliana* trimmed reads were mapped to the *A. thaliana* genome (TAIR10) and the *Z. mays* trimmed reads were mapped to the *Z. mays* genome (AGPv4 Ensembl Genomes 39) using the TopHat2 aligner (Kim *et al.*, 2013). With customized parameters for different datasets, minimum intron length and maximum intron length were adjusted based on values obtained through previous experimental results. Therefore, TopHat2 was run on both the reads with values of default parameters changed to suit *A. thaliana* and *Z. mays* genome intron lengths. For *A. thaliana* dataset, minimum intron length (-i) was set to 40, maximum intron length (-I) was set to 5000, segment length was set to 20, segment mismatches was set to 2, number of max-multihits (-g=1), minimum normalised depth (F) was set to 0 and minimum anchor length was set to 10 (-a=10).

For *Z. mays* dataset, minimum and maximum intron lengths were set to 5 and 60,000 respectively, segment mismatches was set to 1, max-multihits was also set to 1 and segment length to 25. The rest of the parameters were kept to the default. The parameter values are summarized in Table 2.1.

Trimmed reads were also aligned using Bowtie2 with minimum (i) and maximum (I) intron lengths as mentioned above for three datasets. Similarly, the maximum intron length for plant genomes, which is otherwise set to 500bp, is much larger than vertebrates. By setting the max-multihits option to 1, we are forcing unique mapping of the reads to the genome which will allow for the best mapping of the read to the genome. In the *A. thaliana* dataset, by setting the value of minimum anchor length to 10 instead of 8, TopHat2 will report junctions spanned by reads with at least this many bases on each side of the junction. Finally, to eliminate the heuristic filter associated with vertebrate genomes, the minimum normalised depth was set to 0 instead of 300.

Table 2.1: List of some parameters used for reference alignment of reads using Tophat2. Each parameter contains their description, default value and the changed value for the analysis.

Flag	**Meaning**	**Default Value**	**Customized Value for *A. thaliana***	**Customized Value for *Z. mays***
-i	The minimum intron length.	70 nt	40 nt	5 nt
-I	The maximum intron length.	500000 nt	5000 nt	60000 nt
--segment-length	Each read is cut up into segments, each at least this long. These segments are mapped independently.	25 segments	20 segments	25 segments
-g	Instructs TopHat to allow up to this many alignments to the reference for a given read, and choose the alignments based on their alignment scores if there are more than this number.	20 alignments	1 alignment	1 alignment
-a	The "anchor length".	8 bp	10 bp	8 bp
-F	Minimum normalised depth	300 bp	0 bp	300 bp
--segment-mismatches	Read segments which are mapped independently allows this many number of mismatches in each segment alignment	2 mismatches	2 mismatches	1 mismatches

Reads aligned using TopHat2 were then used by Cufflinks (Figure 2.1) for assembling individual transcripts with the above-mentioned minimum and maximum intron lengths parameters. In plant genomes, the difficulty of estimation of transcript abundance arises due to multi-reads and the genome becomes highly-repetitive. Therefore, to address the uncertainty, an Expectation-Maximisation algorithm (EM) has been applied using Cufflinks for estimating transcript abundance. It computes the fractional distribution of each multi-read after read alignment in the E-step and then estimates relative abundance of transcripts in the M-step until it converges. After obtaining the transcripts for each read, transcripts from two comparable samples were merged using Cuffmerge (Figure 2.1). For example, for comparing S7 with S10, the transcripts of each read of the two samples will have 3 read transcripts (i.e. one for S7 and two for S10 as S10 contains one biological replicate). These were merged to form an assembled transcript GTF file for further analysis.

2.2.4 Differential gene expression analysis

Differential expression analysis of the reads was carried out by testing the samples against the first sample to obtain DEGs at each consecutive stage. For the *A. thaliana* dataset, comparisons of the two samples from consecutive days were also performed (Table 2.2). The reason why the first day in time-series data is chosen for benchmarking was that when the samples are compared against the first sample, significant changes can be observed in plants when time advances, which leads to differential expression of number of genes with significant fold-changes. These analyses were carried out using Cuffdiff. The multi-read-correct option was enabled to carry out an initial estimation procedure that weights and maps the reads to multiple locations on the genome. Quartile normalisation was used to obtain Fragments of Per Kilobase of transcript per Million (FPKM) and fragment counts via the ratio of 75th quartile fragment counts to 75th quartile value across all samples. The significantly expressed genes were obtained by filtering the genes having q-values ≤ 0.05.

Table 2.2: Comparison chart for differential expression analyses. Two analyses for *A. thaliana* were carried out: first, all samples were compared to day 7 (S7) when plants had two leaves visible; second, a step-wise analysis was done between two successive days. For *Z. mays*, ten samples were compared against Day-0 in a consecutive manner.

Arabidopsis thaliana		***Zea mays***
Against S7	**Step analysis**	**Against Day-0**
S7 vs S10	S9 vs S10	Day-0 vs Day-2
S7 vs S11	S10 vs S11	Day-0 vs Day-4
S7 vs S12	S11 vs S12	Day-0 vs Day-6
S7 vs S13	S12 vs S13	Day-0 vs Day-8
S7 vs S14	S13 vs S14	Day-0 vs Day-10
		Day-0 vs Day-12
		Day-0 vs Day-14
		Day-0 vs Day-16
		Day-0 vs Day-18
		Day-0 vs Day-20

Sequence read counts were obtained from the reads aligned by Tophat2 using the HTSeq tool to generate raw read counts. The read counts were then used to produce a list of DEGs using the DESeq and edgeR tools. For *A. thaliana*, comparative analysis of S7 against S8 to S16 and step-wise analysis were conducted. Since the dataset contains partial replicates for 5 samples (S9N to S14N), we used blind dispersion estimation for samples with no replicates along with the sharing mode set to ‘fit-only’ and we used pooled empirical dispersion for samples with one or more replicates. The negative binomial method was applied for obtaining DEGs. Results were filtered based on FDR (q-value) <= 0.05 and log2 fold-change less than -2 and greater than 2. To compare samples involving replicates, the Generalised Linear Model (GLM) was applied for estimating common and tagwise dispersion. To compare samples for which no replicates were found, Fisher’s exact test was applied with the biological coefficient of variation set to 0.2 (Benjamini and Hochberg, 1995). For performing DGE analysis using edgeR for samples having no biological replicates, we used common Biological Coefficient of Variation (BCV) with square-root dispersion value which was set to 0.4 for humans and 0.1 for genetically identical organisms (Robinson, McCarthy and Smyth, 2010).

2.2.5 Alternative splicing classification analysis

To obtain statistics of transcript level information, we utilised SpliceR (Vitting-Seerup, B. Porse, *et al.*, 2014) to classify isoform transcripts obtained from Cuffdiff. Output files containing FPKM tracking, count tracking and read group tracking files enabled us to detect Exon Skipping/Inclusion (ESI) events, Alternative Transcription Start Site (ATSS), Alternative

Transcription Termination Site (ATTS), Alternative 3-prime splice site (A3), Alternative 5-prime splice site (A5) and Mutually Exclusive Exon (MEE) events. Additionally, the average number of transcripts per gene and the average number of ESI events per transcript were computed using the spliceR function for each of the sample pairs in three datasets.

2.2.6 Gene Ontology (GO) enrichment, pathway and protein-protein interaction analysis

Results obtained from the overlap of Cuffdiff, DESeq and edgeR were used for the functional enrichment to categorise genes and their associated functions. Overlapping DEGs that were expressed more than once were retained for further analysis. GO enrichment functional annotation and clustering of the genes were performed using the Araport portal (Krishnakumar *et al.*, 2015) to identify genes associated with enriched categories. Gene identifiers were used as inputs into the Araport Thalemine tool. These identifiers were then used for enrichment in gene ontologies (biological process, cellular component and molecular function). Pathway analysis was performed using the ClueGo plugin (Bindea *et al.*, 2009) of the Cytoscape software (Shannon *et al.*, 2003).

Gene identifiers were used to identify the association and clustering of genes in pathways using KEGG (Ogata *et al.*, 1999), REACTOME (Croft *et al.*, 2014) and WikiPathways (Kutmon *et al.*, 2015) databases. Enrichment or depletion of GO categories in ClueGO was performed using the two-sided hypergeometric test and FDR was calculated for the enriched GO categories using the Benjamin and Hochberg (1995) approach. Gene enrichment and clustering results obtained from Araport and Cytoscape were further filtered with FDR ≤ 0.05 to identify highly significant enriched clusters. A PPI network was constructed using the GeneMania plugin (Mostafavi *et al.*, 2008) of the Cytoscape software to obtain prevalent interactors and their degree of interactions from the network.

2.2.7 Calculation of relative expression values

To calculate relative expression values, FPKM counts were used in each sample pair. Counts were normalised by dividing the sample pair read count by the maximum read count value from all other sample pairs to obtain values between 0 and 1. Expression profiles of each gene were constructed by comparing expression values from Cuffdiff and DESeq-edgeR.

2.2.8 Calculating correlation of expression values

For calculating the correlation between the expression profiles, Pearson's Correlation Coefficient (PCC) (Williams, 1996) was used. Expression profiles of DEGs involved in flower development for *A. thaliana* dataset were compared against the expression profiles of FLC and LFY genes to obtain the PCC between them. Also, the difference in expression using PCC was also evaluated by comparing the expression profiles of genes obtained from Cuffdiff, DESeq and edgeR with those obtained from Klepikova et al. (2015).

2.3 Datasets

2.3.1 Reference sequence datasets

Since a reliable dataset is important for model training and prediction, an unbiased random selection of protein-coding and lncRNA transcripts were obtained from the Refseq (Pruitt, Tatusova and Maglott, 2007) and GENCODE (Harrow *et al.*, 2012) databases for constructing reference gold-standard datasets which are composed of FASTA files for different species. These reference datasets contain two categories: mammalian and plants.

For mammalian, protein-coding (mRNA) and long non-coding RNA (lncRNA) sequences of *Homo sapiens* (HS) and *Mus musculus* (MM) were downloaded from the GENCODE database. For HS, a total of 95146 mRNA and 27720 lncRNA sequences were extracted. Whereas for MM a total of 62112 mRNA and 16113 lncRNA sequences were obtained out of which 5000 were randomly selected from HS and MM datasets.

For plants, as there is no dedicated IncRNA database, lncRNA and mRNA of *Arabidopsis thaliana* (ATH), *Brassica rapa* (BRA), *Brassica napus* (BNA), *Brassica oleracea* (BOL), *Zea mays* (ZM), *Oryza sativa* (OS), *Solanum tuberosum* (ST) and *Solanum lycopersicum* (SL) were downloaded from the RefSeq database. For ATH, 66066 mRNA sequences and 4950 ncRNA sequences were obtained from Refseq out of which 4219 mRNA and lncRNA sequences were randomly selected. lncRNA sequences for all plant species were obtained by applying a threshold cutoff of 200bp on ncRNA FASTA files. Details of the number of transcript sequences extracted from RefSeq and GENCODE have been provided in Table 2.3.

Table 2.3: Number of transcript sequences obtained from RefSeq and GENCODE.

Species	**Source**	**Number of mRNA transcripts**	**Number of lncRNA transcripts**	**Number of ncRNA transcripts**
Arabidopsis Thaliana	RefSeq	66066	4219	4950
Brassica Rapa	RefSeq	68631	8670	8983
Brassica Napus	RefSeq	114795	16391	16835
Brassica Oleracea	RefSeq	57387	7774	7942
Homo Sapiens	GENCODE	95146	27720	-
Mus Musculus	GENCODE	62112	16113	-
Oryza Sativa	RefSeq	105139	5516	6406
Solanum Lycopersicum	RefSeq	53678	4182	4351
Solanum Tuberosum	RefSeq	38004	3194	3559
Zea Mays	RefSeq	230720	7917	9274

2.3.2 RNA-seq datasets

As described in Section 2.2.1, two RNA-seq datasets were used for the identification of DE genes. Details of the datasets have been presented and discussed in the Section 2.2.1.

2.4 Workflow of computational framework

The complete workflow (Figure 2.2) of the analysis is divided into four components:

1) Sequence mapping
2) Feature extraction, optimisation and prediction
3) lncRNA sub-classification
4) lncRNA function prediction

In the first component, raw sequence reads are mapped based on reference genome or mapped *de-novo* in the absence of reference genome. The second component extracts features from the transcript sequences, performs feature selection and optimisation, and predicts lncRNAs from sets of transcript sequences. The third component sub-classifies the lncRNAs sequences. The fourth component of the framework performs function prediction of lncRNAs by first computing lncRNA and protein interactions using NRLMF approach (Liu *et al.*, 2016). Using lncRNA-protein interactions, protein-protein interaction and protein function association data, molecular functions of lncRNAs are predicted using the BMRF approach (Kourmpetis *et al.*, 2010).

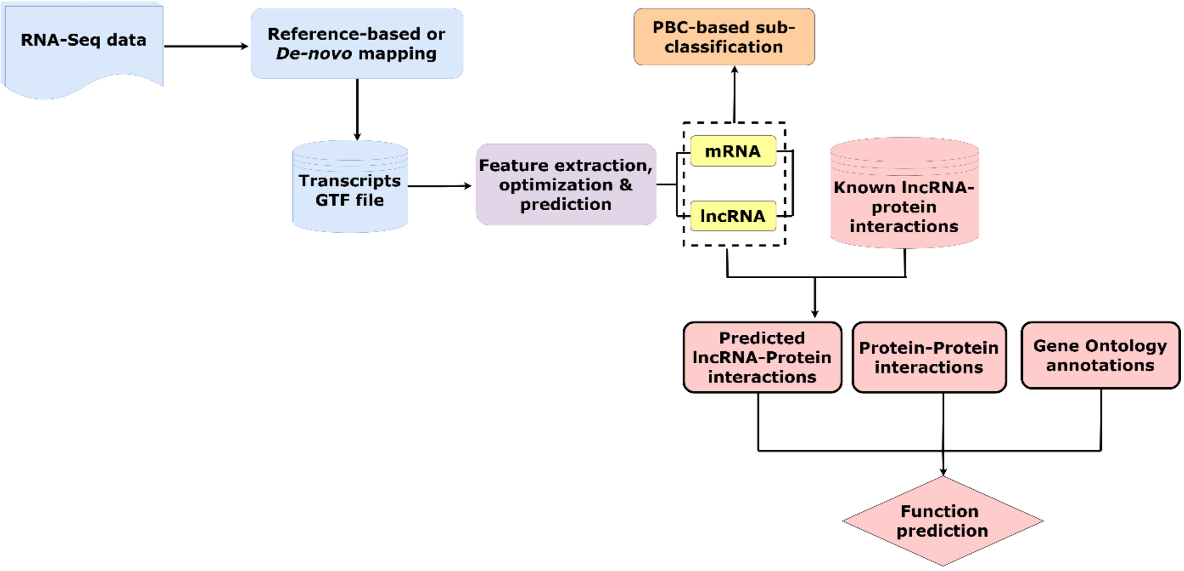

RNA-Seq data
Reference-based or *De-novo* mapping
PBC-based sub-classification
Transcripts GTF file
Feature extraction, optimization & prediction
mRNA
lncRNA
Known lncRNA-protein interactions
Predicted lncRNA-Protein interactions
Protein-Protein interactions
Gene Ontology annotations
Function prediction

Figure 2.2: Workflow of the framework for identification and functional prediction of lncRNAs.

2.5 Features for lncRNA identification

To identify and classify lncRNA and mRNA sequences, many features were extracted from FASTA sequences which are categorised into either ORF-based features or codon bias features (Table 2.4). These features constitute a feature set $\mathrm{F} = \{\mathrm{f1}, \mathrm{f2}, \mathrm{f3} \ldots \mathrm{fn}\}$, where fn denotes the n^{th} feature. The features used are derived from two separate groups: (1) Open Reading Frame (ORF) based and sequence-based features, and (2) codon-bias based features, which are extracted for producing the feature matrix for the identification of lncRNA sequences. Since the framework employs alignment-free approach for lncRNA prediction, the features were selected based on previous knowledge of sequence measures and codon bias measures (Fickett and Tung, 1992; Roth, Anisimova and Cannarozzi, 2012).

Table 2.4: Features used for classification of lncRNAs

ORF and Sequence based features			***Codon Bias features***		
Feature name	**Feature number**	**No. of features**	**Feature name**	**Feature number**	**No. of features**
Max ORF length	$f_{1\mathrm{e}}$	1	Frequency of Optimal codons (Fop)	$f_{8\mathrm{e}}$	1
ORF coverage	$f_{2\mathrm{e}}$	1	Codon Usage Bias (CUB)	$f_{9\mathrm{e}}$	1
Mean ORF coverage	$f_{3\mathrm{e}}$	1	Relative codon bias (RCB)	$f_{10\mathrm{e}}$	1
Transcript length	$f_{4\mathrm{e}}$	1	Weighted sum of relative entropy (Ew)	$f_{11\mathrm{e}}$	1
GC content	$f_{5\mathrm{e}}$	1	Synonymous codon usage order (SCUO)	$f_{12\mathrm{e}}$	1
Fickett score	$f_{6\mathrm{e}}$	1	Relative synonymous codon usage (RSCU)	$f_{13\mathrm{e}}$	61
Hexamer score	$f_{7\mathrm{e}}$	1			

2.5.1 ORF and Sequence based features

We extracted three ORF-based features: maximum ORF length (f_1), ORF coverage (f_2) and mean ORF coverage (f_3) and 4 sequence-based features: transcript length (f_4), GC content (f_5), Fickett score (f_6) and Hexamer score (f_7). f_1 is the maximum length of the ORF. f_1 is one of the most fundamental feature used to distinguish lncRNA from mRNA as majority of protein-coding genes have ORFs greater than 100 amino acids (Frith *et al.*, 2006). f_2 is the ORF coverage defined as the length of the longest ORF divided by the transcript length. This feature has also been shown to produce good classification performance when compared to ORF length (Wang *et al.*, 2013; Zhao, Song and Wang, 2016). f_3 is the mean ORF Coverage defined as average of the total ORF lengths divided by transcript length for sequence. f_4 is the total length of each transcript sequence. f_5 is the GC content, which is also a common measure to differentiate lncRNA from protein-coding transcripts as coding sequences have been reported to have higher GC content in exons over introns (Amit *et al.*, 2012). GC content is simply calculated as absolute total number of GC motifs in a sequence. f_6 is the Fickett score (Fickett, 1982) obtained by calculating four base pair position values in transcript sequence. f_6 is calculated as follows: Let

A_1 = Number of A's in positions 1, 4, 7, 10, ….,
A_2 = Number of A's in positions 2, 5, 8, 11, …, and

A_3 = Number of A’s in positions 3, 6, 9, 12, ….,

Then $A_{position}$ is defined as:

$$A_{position} = \frac{\mathrm{MAX}(\mathrm{A}_1, \mathrm{A}_2, \mathrm{A}_3)\mathrm{e}}{\mathrm{MIN}(\mathrm{A}_1, \mathrm{A}_2, \mathrm{A}_3) + 1\mathrm{e}} \quad (1.1)$$

and $T_{position}$, $G_{position}$ and $C_{position}$ are calculated similarly. In a similar manner, $A_{content}$, $T_{content}$, $G_{content}$ and $C_{content}$ of the sequence are determined by calculating percentage composition of each base in the sequence. These eight values are then converted to a probability value (*p*) using a lookup table (Fickett, 1982) and multiplied by a weight (*w*) for each base. The Fickett score f_6 is then determined as:

$$f_6 = \mathrm{e}\sum_{i=1\mathrm{e}}^{8\mathrm{e}} p_i w_{i\mathrm{e}} \quad (1.2)$$

$f_{7\mathrm{e}}$is the hexamer score which is computed by making a hexamer table of 4096 (64×64 hexamers) *k*-mers using a reference set of coding and non-coding sequences. Hexamer score is calculated by first measuring frequencies of hexamers in the test set sequences. The logarithmic ratio of coding and non-coding sequences is then computed for each hexamer having non-zero frequency in the test set. Positive f_7 indicates higher probability of protein-coding sequence whereas negative score indicates higher probability of non-coding RNA sequence. The in-frame hexamer frequency of protein-coding sequences is given by $\mathrm{F(h_{i\mathrm{e}})}$ where $\mathrm{i} = 0, 1, \dots, 4095\mathrm{e}$and in-frame hexamer frequency of lncRNA sequences is given by $\mathrm{F'(h_i)}$ where $\mathrm{i} = 0, 1, \dots, 4095$. Therefore, for each hexamer sequence, $S = H_1, H_2, H_3, \dots, H_m$, f_7 is given by:

$$f_7 = \frac{1}{m\mathrm{e}}\sum_{i=1\mathrm{e}}^{m\mathrm{e}} \log\left(\frac{F(h_i)}{F'(h_i)}\right) \quad (1.3)$$

2.5.2 Codon-biased features

In protein-coding genes, the translational mapping process of codons (or nucleotide triplets) to amino acids involves the use of synonymous codons which codes the same amino acids which are non-distinguishable at protein level. However, it has been reported that there exists a non-uniform codon usage in most genes which causes codon bias (Clarke, 1970; Ikemura, 1982). Many indices have been proposed for measuring codon bias; usage of all the indices is beyond the scope of this study. Therefore, we carefully selected six codon-bias measures which could be important in distinguishing lncRNAs from mRNAs. These are frequency of optimal codons (f_8) (Fickett, 1982; Amit *et al.*, 2012), codon usage bias (f_9) (Karlin and Mrázek, 1996), relative codon bias (f_{10}) (Roymondal, Das and Sahoo, 2009), weighted sum of relative entropy (f_{11}) (Suzuki, Saito and Tomita, 2004), synonymous codon usage order (f_{12}) and relative synonymous codon usage (f_{13}) (Wan *et al.*, 2004).

f_8 is the frequency of optimal codons (*Fop*) which is calculated as the ratio of the total number of optimal codons to the total number of synonymous codons. Fop was also one of the measures proposed by Ikemura (Ikemura, 1982, 1985). Optimal codons are defined based on nucleotide chemistry and must fulfill two criterions: (a) pyrimidine two codons AA prefer A-ending over G-ending (Bulmer, 1988), (b) purine two codons AA prefer C-ending over U-ending (Bulmer, 1988). The number of optimal codons is calculated as:

$$O_{opt} = \mathrm{e} \sum_{c \in C_{opt\mathrm{e}}} O_c \tag{1.4}$$

Where C_{opt} is defined as subset of optimal codons from all codons C and O_{tot} is the total number of codons in the sequence. Therefore, f_8 is calculated as:

$$f_8 = \mathrm{e} \frac{opt}{O_{tot\mathrm{e}}} \tag{1.5}$$

Amino acids with one codon Methionine (M) and Tryptophan (W) were excluded from the analysis as they do not contribute any information.

f_9 is the Codon Usage Bias (CUB) which assesses the codon bias in the test set relative to the reference set. It is based on the weighted sum of distances of relative codon usage frequencies between the reference set and test set sequences (Karlin and Mrázek, 1996). The reference set is used as standard to which other sequences can be compared against f_9 is defined as:

$$f_9 = \sum_{a\in \mathrm{Ae}} F_a d(f_a, f_a^{ref}) \tag{1.6}$$

where F_a is frequency of amino acid a in the test set sequence whereas f_a and $f_a^{refe}$are codon frequencies for amino acid a in test and reference sets, respectively and d is the L1 norm or manhattan distance for the codon frequency f_a and f_a^{ref} vectors which is calculated as:

$$d(f_a, f_a^{refe}) = \mathrm{e}\sum_{c\in C_{a\mathrm{e}}} |f_{ac}, f_a^{ref}|\mathrm{e} \tag{1.7}$$

where f_{ac} is the frequency of codon c encoding amino acid a in the test set sequences and $f_a^{refe}$is the frequency of amino acid a in the reference set sequences.

f_{10} is the Relative Codon Bias (RCB) (Roymondal, Das and Sahoo, 2009) which is a measure that defines the contribution of a codon as:

$$w_c^{RCB\mathrm{e}} = \frac{{}_c - E[\mathrm{e}_c]\mathrm{e}}{E[O_c]}, \tag{1.8}$$

where $E[O_c]$ is the expected number of codon occurrences in three codon positions. Once w_c^{RCB} is determined the RCB score is calculated by the following method for each sequence:

$$f_{10} = \exp(\mathrm{e}\frac{1}{O_{tot}}\sum_{c\in \mathrm{Ce}} log\ w_c^{RCB}) - 1\mathrm{e} \tag{1.9}$$

f_{11} feature used is the weighted sum of the relative entropy (Ew) which measures the degree of deviation from equal codon usage (Suzuki, Saito and Tomita, 2004). Therefore, Ew is defined as the sum of relative entropy of each aa weighted by its relative frequency in the test sequence which is given by:

$$f_{11\mathrm{e}} = \mathrm{e}\sum_{a\in \mathrm{A}} F_a E_{a\mathrm{e}} \tag{1.10}$$

Here F_a is the relative frequency of amino acid a in the test sequence and E_a is computed as:

$$E_a = \frac{H_a}{log_2 k_{a\mathrm{e}}} \tag{1.11}$$

where k_a is number of synonymous codons observed in the test sequence and H_a is the entropy which measures the uncertainty of codon usage in the test sequence for amino acid a and is computed as:

$$H_a = -\mathrm{e}\sum_{c \in C_{a\mathrm{e}}} f_{ac} log_2 f_{ac}. \tag{1.12}$$

f_{12} is the Synonymous Codon Usage Order (SCUO) and is also an entropy-based codon bias measure, similar to Ew which differs only by the way entropy is calculated for each amino acid (Wan *et al.*, 2004). Instead of calculating the relative entropy, normalised difference between maximum and observed entropy is computed as:

$$E_a = \mathrm{e}\frac{log_2 k_a - H_{a\mathrm{e}}}{log_2 k_{a\mathrm{e}}}. \tag{1.13}$$

Then the SCUO is computed as:

$$f_{12\mathrm{e}} = \mathrm{e}\sum_{a \in \mathrm{A}} F_a E_{a\mathrm{e}} \tag{1.14}$$

f_{13} is the Relative Synonymous Codon Usage (RSCU) score which defines the relationship between observed codon frequencies and the number of times codon is observed when synonymous codon usage is random with no codon bias (Sharp, Tuohy and Mosurski, 1986). This is calculated as:

$$RSCU_{ac} = \frac{ac\mathrm{e}}{\frac{1\mathrm{e}}{k_{a\mathrm{e}}} \sum_{c \in C_a} O_{ac}} \tag{1.15}$$

where O_{ac} is the frequency of codon c for amino acid a. $RSCU_{ac}$ is the RSCU score for each codon c encoding amino acid a and is computed for 61 codons individually by the above equation. Methionine (M), Tryptophan (W) and stop codons were excluded from the analysis as M and W do not have any synonymous codons and stop codons do not contribute any information. Therefore, in the total RSCU score provided 61 features for the classification.

The codon-biased or codon-usage features were computed by computing codon-bias on the whole transcript sequence. However, an alternative to this approach is to compute the codon-bias characteristics based on the longest ORF in the transcript. However, the codon-bias features computed from the longest ORF in the transcript did not generated increase in the prediction accuracies when compared with those compared against features from the whole transcript sequence. Therefore, codon-biased featured based on the former approach has been implemented in this thesis.

2.5.3 Feature normalisation

Features extracted from the pool of transcript sequences were concatenated to generate a single matrix $\mathrm{X}^{N \times d}$ containing N features and d sequences and Y class label consisting of binary class values of size d. Feature vector matrix was normalised to scale values between 0 and 1 using the following equation:

$$X_{Normalised} = \frac{X - \min(X)}{\max(X) - \min(X)}. \quad (1.16)$$

Normalised feature matrix ($X_{Normalized}$) was then used for the creation of training and testing datasets by randomly selecting 75% and 25% from $X_{Normalized}$ feature matrix.

2.6 Feature extraction from RNA-seq datasets

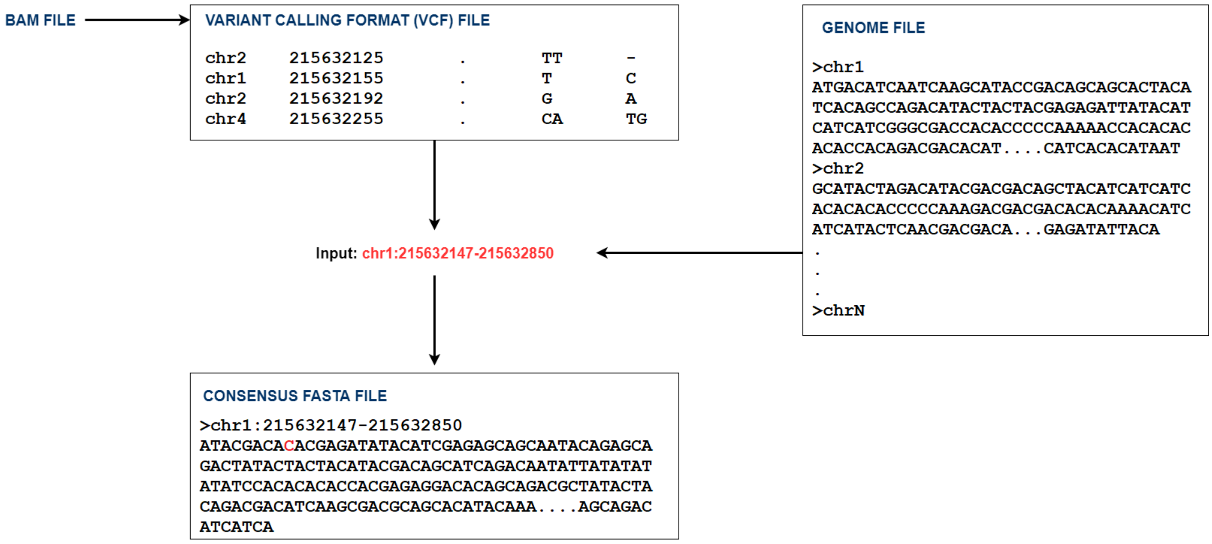
For the identification of lncRNAs from RNA-seq datasets, an aligned sequence file (BAM file) was created from Tophat2. The BAM file was used for extraction of FASTA sequences. For FASTA sequence extraction from the BAM file, a consensus FASTA sequence for each transcript coordinate was constructed by a two-step process (Figure 2.3):

(1) SNP and INDEL calling of BAM file using samtools *mpileup* which generated a VCF file, and

(2) sequence extraction from the genome and consensus sequence generation using variants from VCF by samtools *faidx* tool (Li *et al.*, 2009).

The first step generated a list of variants from the BAM file. These were used in conjunction with the genomic coordinates and genome file to extract consensus FASTA sequences from user defined genomic regions.

Figure 2.3 represents an example workflow of how a desired consensus transcript FASTA sequence is produced. From the BAM file, a Variant Calling Format (VCF) file is generated using samtools. For each query coordinate supplied, “chr1:215632147-215632850” in this case, the program extracts the consensus FASTA sequences using the genome file of the species to substitute “T” with “C” in 215632155 base position in the genome file. Once the FASTA sequences are extracted, features are extracted for transcript sequences to construct feature matrix file. The feature set is normalized as described in Section 2.5.2.


BAM FILE
VARIANT CALLING FORMAT (VCF) FILE
Input: chr1:215632147-215632850
GENOME FILE
CONSENSUS FASTA FILE

```
chr2    215632125    .    TT    -
chr1    215632155    .    T     C
chr2    215632192    .    G     A
chr4    215632255    .    CA    TG
```

```
>chr1
ATGACATCAATCAAGCATACCGACAGCAGCACTACA
TCACAGCCAGACATACTACTACGAGAGATTATACAT
CATCATCGGGCGACCACACCCCCAAAAACCACACAC
ACACCACAGACGACACAT....CATCACACATAAT
>chr2
GCATACTAGACATACGACGACAGCTACATCATCATC
ACACACACCCCCAAAGACGACGACACACAAAACATC
ATCATACTCAACGACGACA...GAGATATTACA
.
.
.
>chrN
```

```
>chr1:215632147-215632850
ATACGACACACGAGATATACATCGAGAGCAGCAATACAGAGCA
GACTATACTACTACATACGACAGCATCAGACAATATTATATAT
ATATCCACACACACCACGAGAGGACACAGCAGACGCTATACTA
CAGACGACATCAAGCGACGCAGCACATACAAA....AGCAGAC
ATCATCA
```

Figure 2.3: Conceptual workflow of consensus FASTA file generation from sequence alignment file.

2.7 Classifier used for lncRNA identification

For the classification of labeled data, eg. the reference dataset obtained from Refseq (Pruitt, Tatusova and Maglott, 2007) and GENCODE (Harrow *et al.*, 2012), an iterative Random Forests (iRF) classifier (Basu *et al.*, 2018) has been used on the extracted features for classification purposes. Based on the Principle of Stability, iRF can detect higher order interactions of DNA and protein structures. Classification results using iRF classifier on

Drosophila species indicate several fifth and sixth order interactions (Basu *et al.*, 2018). The inherent structure of the random forest algorithm implemented in the iRF classifier enables to detect higher-order feature combinations which is particularly suitable for applications in genomic, transcriptomics and epigenomics NGS datasets.

iRF uses the supervised learning approach for identifying class-specific index sets which is needed for the RIT algorithm (Shah and Meinshausen, 2014). This framework allows for the detection of higher-order combinations in feature-weighted RF. Considering a binary classification setting with training dataset D, the data is represented in the following form where, $\{(x_i, y_i)\}_{i=1}^{n}$ with categorical or continuous variables, where, $\mathrm{x} = (x_{1,} x_{2,} \dots, x_p)$ with binary labels y_{ie}∈ $\{0,1\}$, our goal is to find subsets of features or interactions which are highly frequent or common within a class $C \in \{0,1\}$ and provide recognisable differentiation between the two classes. To generalise the results, interactions are searched in decision tree ensembles fitted on bootstrap samples. For the classification and determination of interactions, iRF consists of three components: (1) Iteratively re-weighted Random Forest, (2) Generalised RIT, and (3) Bagged Stability Scores.

(1) Iteratively re-weighted Random Forest: Given K (an iteration number), iRF iteratively grows weighted RFs on data D such that $RF\left(w^{(k)}\right), k = 1, \dots, K$. The first iteration of iRF when $k = 1$ starts with $w^{(1)} := \left(\frac{1}{p}, \dots, \frac{1}{pe}\right)$, and stores the Gini importance, also called as mean decrease in Gini impurity of p features is denoted as $v^{(1)} = (v_1^{(1)}, \dots, v_p^{(1)})$. In the second iteration when $k = 2$, we set the $w^{(k)} = v^{(k-1)}$ and weighted RFs are grown with weights set equal to the importance of RF feature from previous iteration.

(2) Generalised RIT: Generalized RIT is applied on the last feature-weighted RF grown in the Kth iteration. The collection of trees generated in the process of fitting $RF\left(w^{(k)}\right)$ provides mappings from categorical to binary features, which produces a collection of interactions. To determine feature combinations or interactions, each tree $t = 1, \dots, T$ in the output tree ensemble of RF has leaf nodes collected and indexed by $j_t = 1, \dots, J(t)$. Each feature-response pair (x_i, y_i) is represented for each tree by (Z_{i_t}, I_{i_t}) where I_{i_t} is the set consisting of unique feature indices which falls on the path of the leaf node containing (x_i, y_i) in the t^th^ tree. Therefore, each $(x_i, y_{ie}$ produces T index set-label pairs which corresponds to T trees. The pairs are aggregated across trees and observations such as $R = \{\left(Z_{i_t}, I_{i_t}\right): x_i \text{ falling on leaf node } i_t \text{ of tree } t\}$ to obtain a set R of interactions.

(3) Bagged Stability Scores: Once the set R of interactions is obtained, the stability score of an interaction is defined as $sta(S) = \frac{1}{B}\sum_{b=1}^{B} 1\{S \in S_{(b)}\}$ which represents the number of times interaction occurs.

2.8 Implementation of iRF for lncRNA identification

2.8.1 Classification on labeled feature set

For the identification of lncRNAs from protein-coding transcript sequences and to benchmark the classification accuracy of iRF against known coding-potential tools, sequences were extracted from reference databases and tested using the iRF classifier. 73 features were constructed from each FASTA sequence to generate feature matrix $X^{n\times d}$ (where n is the number of transcript sequences and d is the number of features) and Y number of classes ($Y =$e $\{0,1\}$) where mRNA is 0 and lncRNA is represented by 1. iRF classifier was then used for model fitting using training and test sets feature matrices with labeled class values. Classification was performed for sequences extracted from reference sequence datasets (Refseq and GENCODE) using Algorithm-1 (Table 2.5).

Table 2.5: Algorithm for classification and identification of lncRNAs from feature matrix based on labeled test set.

```
Algorithm-1: iRF classification with reference sequence data
Input: Xtrain: n x d matrix with n-1 features and d feature elements in training set
Ytrain: n x d matrix with (n-(n-1)) vector and d class elements in training set
Xtest: n x d matrix with n-1 features and d feature elements in test set
Ytest: n x d matrix with (n-(n-1)) vector and d class elements in test set
ntrees: number of random forest trees
Output: vector accRfPred containing binary prediction values for Xtest
Load Xtrain, Xtest, Ytrain, Ytest
p ← number of columns(Xtrain)
n ← no. of iterationse
selProb ← replicate($\frac{1}{p}$, p)
initialize rf as liste
for iter = 1 to n do
        rf[iter] = Fit RF(Xtrain, Ytrain, Xtest, Ytest, selProb, ntrees)
        selProb ← GiniImportance( rf[iter])
end for
rfMaxIndex ← Get index of maximum accuracy value from rf[iter] list
rfMax ← extract predictions from the test set stored in rf[rfMaxIndex] list
```

The algorithm starts with training and testing set files. A fixed set of selection probabilities are assigned for each feature in the training set. Number of random forest trees are also assigned. In iRF, initially fixed selection probabilities are assigned as the algorithm starts with a fixed probability of selection of features. The selection probability is then updated with the generation of trees and the classification of sequences. For each iteration, the selection probabilities are stored in rf[iter]e object where length of rf object equals the number of iterations performed. The index of the rfeobject generating the highest accuracy is extracted and stored in rfMaxIndex. Predictions are then extracted from rf[rfMaxIndex][test] and stored in rfMax.

2.8.2 Classification on unlabeled feature set

For the classification of lncRNAs from the test set feature matrix having unlabeled or no class labels, classification was performed using the iRF classifier using Algorithm-2 described in Table 2.6. Results of iRF classification are stored in a predictions vector which are finally written to a text file.

Table 2.6: Algorithm for classification and identification of lncRNAs from feature matrix based on unlabeled test set.

```
Algorithm-2: iRF classification with unlabeled test set
Input: Xtrain: n x d matrix with n-1 features and d feature elements in training set
Ytrain: n x d matrix with (n-(n-1)) vector and d class elements in training set
Xtest: n x d matrix with n-1 features and d feature elements in test set
ntrees: number of random forest trees
Output: vector predictions and output file containing binary prediction values for Xtest
Load Xtrain, Xtest, Ytrain
p ← number of columns(Xtrain)
n1 ← number of rows in Xteste
Ytest ← randomise(e1,2) with size of n1e
selProb ← replicate($\frac{1}{p}$, p)
rf =eFit RF(Xtrain, Ytrain, Xtest, Ytest, selProb, ntrees)
predictions ← extract predictions from rf[test] list
```

Training and test set files are assigned to Xtrain, Xtest, Ytrain variables. Number of RF are assigned to ntrees variable. To identify lncRNAs using iRF, a false class label is created using the "*np.random.randint*" function of Python's Numpy package (Community, 2011) using 1 and 2 as class labels. This function creates randomised values of 1 and 2 with size equals to the number of rows of the "Xtest" feature matrix. This is then appended to the "Ytest" array which is used in iRF for generating predictions on "Xtest". Prediction results are extracted from the rf[test] object and saved to predictions variable.

The iRF classifier has been implemented in this thesis as iRF resulted in higher prediction accuracies when compared against ML-based classifiers. The Artifical Neural Networks (ANN)-based ML methods were not adopted in this project as ANN-based methods do not provide elucidation of potential regulatory motifs or distribution of codons in mRNA and lncRNA transcripts, and therefore the performance of such methods are incomparable against currently available CPC tools. Additionally, ANN-based methods require a significant amount of computational time and resources for model training and is less interpretable which makes it computationally infeasible for lncRNA prediction application.

2.8.3 Performance Evaluation Criteria

To assess performance classification of lncRNAs and mRNA transcripts, the following metrics were calculated:

- Accuracy = $\frac{TP+TN}{TP+FP+FN+TN\mathrm{e}}$
- Sensitivity or Recall = $\frac{TP}{TP+FN}$ measures the proportion of true positive values from the dataset by quantifying false negative values along with true positive values.
- Specificity = $\frac{TN}{FP+TN}$ measures the proportion of true negative values from the dataset by quantifying false positive values along with true negative values.
- Precision or Positive Predictive Value (PPV) = $\frac{TP}{TP+FP}$ is a measure of detecting true positive values from the test dataset by quantifying false positive values along with true positive values.
- F1-Score = $\frac{2\times(Precision\times Recall)}{Precision+Recall\mathrm{e}}$ is weighted average of precision and recall.
- Negative Predictive Value (NPV) = $\frac{TN}{TN+FN}$ is a measure of detecting true negative values from the test dataset by quantifying false negative values along with true negative values.
- Matthews Correlation Coefficient (MCC) = $\frac{(TP\times TN)\mathrm{e} - (FP\times FN)}{\sqrt{(TP+FP)\times(FN+TN)\times(FP+TN)\times(TP+FN)}}$ is a measure of assessment of the quality of two-class classification problem. The correlation coefficient value lies between -1 and +1 with +1 being perfect prediction.

where TP denotes True Positive, TN True Negative, FP False Positive and FN False Negative.

2.9 Feature selection and implementation on RNA-seq datasets

2.9.1 Background

Selection of optimal features is an important optimisation approach for classification. Least Absolute Shrinkage and Selection Operator (LASSO) (Tibshirani, 1996) is a feature selection method which combines least-square loss with the $\ell 1$ norm constraint and produces sparse features by shrinking coefficients to zero. Other approaches such as ridge regression (Marquardt, 1970; Tibshirani, 1996), use the $\ell 2$ norm due to which it produces non-zero coefficients and therefore becomes inefficient for feature selection. Usage of the ℓq norm (with q < 1 or q > 1) approaches for optimisation are generally non-convex and makes the minimisation computationally challenging. Given a dataset D with n feature vectors of length p arranged in a design matrix $X \in \mathbb{R}^{nxp}$, we would like to predict n x 1 response vectors as $y \in \mathbb{R}^n$ in a linear model. LASSO (Tibshirani, 1996) solve the $\ell 1$-regularised optimisation problem by the following objective function:

$$\beta_{lasso} = \mathrm{e}\underset{\beta\epsilon\mathbb{R}^p}{argmin\mathrm{e}}||y - X\beta||_2^{2\mathrm{e}} + \lambda \sum_{j=1\mathrm{e}} |\beta_j| \tag{1.17}$$

$$\beta_{lasso} = \underset{\beta\epsilon\mathbb{R}^p}{argm\ n}||y - X\beta||_2^2 + \lambda||\beta||_{1\mathrm{e}} \tag{1.18}$$

, where $\lambda \geq 0$, $||y - X\beta||_2^2$ is the loss function (i.e. sum of squares), $||\beta||_1$ is the penalty term and λeis the tuning parameter which controls the strength of the penalty. The LASSO estimate can also be written as:

$$\beta_{lasso} = \mathrm{e}\underset{\beta}{argmin\mathrm{e}}\sum_{i=1\mathrm{e}}^{N\mathrm{e}}(y_i - \beta_0 - \mathrm{e}\sum_{j=1} x_{ij}\beta_j)^2 \tag{1.19}$$

$$\text{subject to}\mathrm{e}\sum_{j=1\mathrm{e}}^{p\mathrm{e}} |\beta_j| \leq t.$$

The only reason LASSO is chosen over Ridge regression is that LASSO yields sparse solutions. The equivalent Lagrangian form of LASSO problem is written as:

$$\beta_{lasso} = e \underset{\beta e}{argm\,ne}\left\{\frac{1}{2e}\sum_{i=1e}^{Ne}(y_i - \beta_0 - e\sum_{j=1e} x_{ij}\beta_j)^2 + \lambda e\sum_{j=1} |\beta_j|\right\} \tag{1.20}$$

which can also be written as,

$$\beta_{lasso} = e \underset{\beta}{argm\,ne}\left\{\frac{1}{2n}||X\beta - y||_2^2 + \lambda||\beta||_1\right\}. \tag{1.21}$$

Here coefficients (β) for each feature are calculated by the following formula:

$$\beta e \quad (X^T X)^{-1} X^T Y e \tag{1.22}$$

where X and Y are training feature matrix and class vector, respectively.

2.9.2 Implementation of LiRF-FS on reference datasets

For selection of optimal features from labeled reference datasets, a feature selection algorithm has been constructed following the LASSO method. An iRF classifier which produces sparse coefficient values for the features based on λ in each iteration, performs classification using iRF and benchmarks each feature set against others, based on the difference between the accuracy obtained from each feature set. The method for the selection of the optimal features using a labeled feature set is called LASSO-iRF Feature Selection (LiRF-FS) method (Figure 2.4) (Table 2.7).

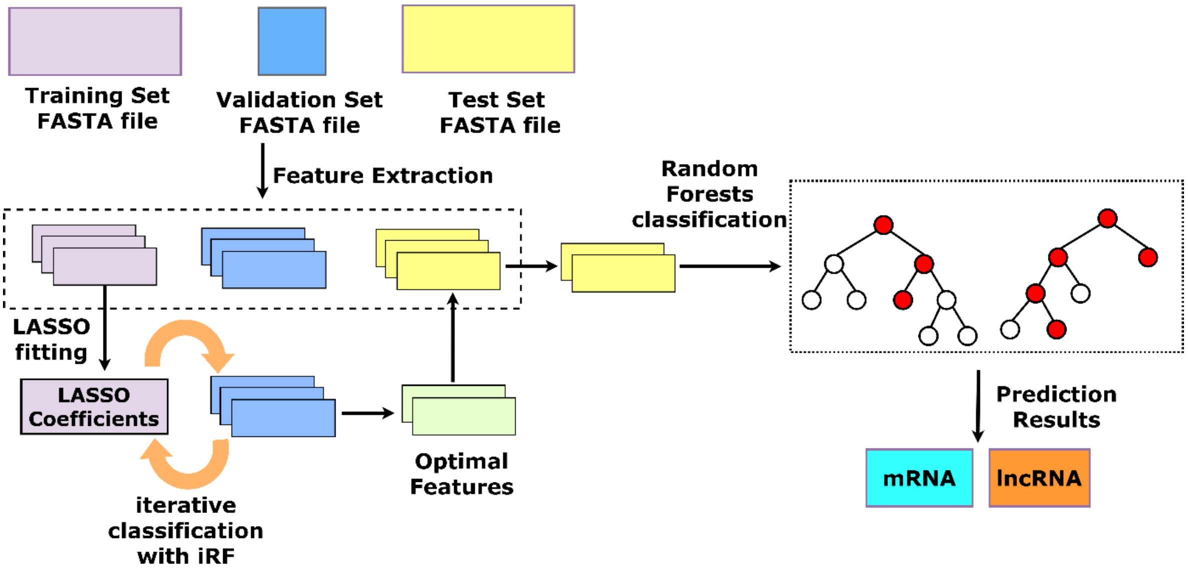


Figure 2.4: LiRF-FS algorithm workflow. Sequence and codon-bias features from the training and validation lncRNA and protein-coding sequences are extracted. LASSO coefficients are generated from the training set and iteratively applied on the validation set using an iRF classifier to test the accuracy at each λ value. Optimal features are selected based on threshold tolerance value which can be applied on the test set sequences using Random Forest classifier. Labelled prediction results are generated for the test set sequences identifying lncRNA or mRNA sequences.

Table 2.7: Algorithm for implementation of LiRF-FS method in labeled dataset.

```
Algorithm 3: LiRF-FS implementation
 Initialize $\lambda_{lower}$, $\lambda_{upp\ r}$, $\lambda_{step-siz}$ , β, n, listArray, thresholdAccDiff, trainingSet, validationSet, ntreese
λ =eList of λ values ranging from $\lambda_{upp}$ tρ $\lambda_{lower}$ value with step-size of $\lambda_{st\ p-size}$
Xtrain =eeature matrix of trainingSete
Xtest =eeature matrix of validationSete
Ytrain =eBinary class vector of trainingSete
Ytest =eBinary class vector of validationSete
tolerance =e1e − 2e
function estBL:
    return $\frac{1}{2n}||X\beta - y||_2^2 + \lambda||\beta||_1$
for i=0 to length(λ) do
beta_estimate =eminimise(estBL, listArray, method =eCD)
  if values in beta_estimate < tolerance then
     set values in beta_estimate =e0e
     beta_estimate_non_zero ← length(values in beta_estimate ≠ 0)
  end if
  beta_estimate_array ← beta_estimatee
```

```
    if beta_estimate_non_zero < length(beta_estimate_array − 1) then
        for j = 1 to beta_estimate_non_zero do
            XtrainF ← Xtrain[j]
            XtestF ← Xtest[j]
            YtrainF ← Ytrain[j]
            YtestF ← Ytest[j]
        end for
        selProb ← replicate($\frac{1}{p}$, p)
        initialize rf as liste
        for iter = 1 to n do
            rf[iter] = Fit RF(XtrainF, YtrainF, XtestF, YtestF, selProb, ntrees)
            selProb ← GiniImportance(rf[iter])
        end for
        maxAccIndex ← Get index of maximum accuracy value from rf[iter] liste
        accRfPred ← maxAccIndexe
    end if
end for
for i=maxAccIndex to 0 do
    diffArrNeg ← accRfPred[i] − accRfPred[i − 1]
end for
for i=maxAccIndex to length(accRfPred) do
    diffArrPos ← accRfPred[i] − accRfPred[i + 1]
end for
for i=0 to length(diffArrNeg) do
    if diffArrNeg[i] ≤ thresholdAccDiff then
        thresArrNeg ← index(diffArrNeg[i])
    else if diffArrNeg[i] ≥ thresholdAccDiff then
        thresArrNeg ← max(accRfPred)
    end if
end for
for i=0 to length(diffArrPos) do
    if diffArrPos[i] ≤ thresholdAccDiff then
        thresArrPos ← index(diffArrPos[i])
    else if diffArrPos[i] ≥ thresholdAccDiff then
        thresArrPos ← max(accRfPred)
    end if
end for
lastElementValueNeg ← Get last element from thresArrNeg liste
lastElementValuePos ← Get last element from thresArrPos liste
optFeaturesNeg ← Extract features from beta_estimate_array using lastElementValueNeg.e
optFeaturesPos ← Extract features from beta_estimate_array using lastElementValuePos.e
```

The main steps of the LiRF-FS method are:

1. Initialization of λ_{lower}, λ_{upper} and $\lambda_{\mathrm{step-size}}$ values; empty listArray vector; n value number of iterations required for iRF classifier; number of trees to generate for ntrees value; empty p integer value; tolerance threshold value (tol) of 10^{-2} and thresholdAccDiff value to user defined value (For ex- thresholdAccDiff = 0.5).

2. Construct the λ list using $\lambda_{\mathrm{lower}}, \lambda_{\mathrm{upp}}$ and $\lambda_{\mathrm{st\ p-siz}}$.
3. Construct Xtrain, Xtest, Ytrain and Ytest using training set and validation set.
4. Calculate β value using X and Y of training set using the following equation, $\beta = (\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}\mathrm{X}^{\mathrm{T}}\mathrm{Y}$.
5. The β_{lasso} in the estBL function calculates the β coefficients using the following equation,

   $$\beta_{\mathrm{lasso}} = \overset{argmin}{\underset{\beta\mathrm{e}}{}} \left\{\frac{1}{2n}||X\beta - y||_2^2 + \lambda||\beta||_1\right\}.$$

   where the coefficients are calculated by coordinate-descent minimisation (Wu and Lange, 2008). The $\mathrm{beta\_estimate\_non\_zero}$ variable stores the non-zero feature coefficient values from the $\mathrm{beta\_estimate\_array}$ list.
6. Non-zero coefficients are selected which are used as indices to construct filtered training set to construct filtered training and filtered test sets for corresponding iteration.
7. Using λ, execute a loop through the upper bound to lower bound λ values.
8. Using if statement, we check whether the length of non-negative beta estimate array is less than beta estimate array. If True, then extract the all the features using index values of the non-zero coefficient values for that λ value in current iteration and construct filtered training and validation sets $\mathrm{XtrainF}$.
9. Construct selection probabilities vector $\mathrm{selProb}$ by defining p as length of columns of training set $\mathrm{XtrainF}$.
10. Using ntrees, $\mathrm{selProb}$ and filtered training and test sets, run iRF classifier to obtain prediction for each iteration.
11. Store prediction information of $\mathrm{rf[iter]}$e by searching for $\mathrm{rf[iter]}$e producing highest accuracy. $\mathrm{accRfPred}$ list stores accuracies for each corresponding λ values.
12. Run a loop through the $\mathrm{accRfPred}$ array to search for maximum accuracy value and store the index of that λ value producing highest accuracy. Store that index value in $\mathrm{maxAccIndex}$.
13. For extracting the least number of features, a loop is executed from $\mathrm{maxAccIndex}$ value to zero$^{\mathrm{th}}$ value in reverse (i.e. $[\mathrm{i} - (\mathrm{i} - 1)],\ [\mathrm{i} - (\mathrm{i} - 2)],\ \ldots, 0)$ with index of $\mathrm{accRfPred}$. The $\mathrm{diffArrNeg}$ array stores the difference between the previous value and the maximum prediction value. If the difference between the maximum prediction accuracy and the previous value is within the $\mathrm{toleranceCutoffValue}$ value $(\mathrm{toleranceCutoffValue}\ =\mathrm{e}\ \mathrm{maxAccValue} - \mathrm{tolerance})$, the value is stored in the $\mathrm{thresArrNeg}$ array. If the

thresArrNeg array does not contain any values, then extract the index of the maxAccValue. The maxAccValue contains the maximum prediction accuracy value. The index contains the value of λ for that particular prediction accuracy.

14. For extracting the highest number of features, a loop is executed from maxAccIndexe value to zeroth value in forward direction (i.e. $[i-(i+1)],\ [i-(i+2)],\ \ldots, n$) with index of accRfPred. The diffArrPos array stores the difference between the next value and the maximum prediction value. If the difference between the maximum prediction accuracy and the next value is within the toleranceCutoffValue value, the value is stored in the thresArrPos array. If the thresArrPos array does not contain any values, then extract the index of the maxAccValue.
15. Using thresArrNege and thresArrPos, extract the last index values to obtain in lastElementValueNeg and lastElementValuePos variables. The variables are the indexes of the lambda value that contains the least and maximum number of optimal features within tolerance value from the maximum prediction accuracy value.
16. Based on the value of λ in optFeaturesNeg and optFeaturesPos variables, the optimal feature set containing the least and the highest optimal features can be obtained based on beta_estimate_array.
17. optFeaturesNeg contains the least number of features and the optFeaturesPos contains the highest number of features producing accuracy within the tolerance value.

2.9.3 LncRNA prediction on plant RNA-seq datasets

Once the LiRF-FS is executed on labeled training and validation sets obtained from the Refseq database, the optimalFeatureSet obtained from LiRF-FS is then used for selection of optimal features from RNA-seq feature matrix set file created using the instructions mentioned in Section 2.6.

Implementation of feature selection and lncRNA prediction on RNA-seq datasets is performed using the following steps:

1. Using GTF annotation file of the species from Ensembl database, extract protein-coding and lncRNA sequences using the instructions in Section 4.6.
2. Construct a feature matrix using 73 ORF and codon-bias features.
3. Using the optimalFeatureSet created from Algorithm 3, extract optimal features from the feature matrix.

4. Execute the Algorithm 2 (Table 4.6) to obtain predictions on test set sequences.
5. Store the results in an output comma-separated values (CSV) file.

2.10 Performance evaluation with k-fold cross validation

To benchmark the performance of the computational framework against the popular coding potential tools, a k-fold Cross Validation (CV) performance validation test was performed. In k-fold CV, the data is randomly partitioned into k equal sized subsets or folds. Of the k subsets, a single subset is retained as validation set for testing the model, and the remaining k-1 subsets are used as training data. The validation set data selected in each fold does not overlap with the data selected in previous folds. The CV is then repeated k times with each of the k subset used exactly once as the validation set. The performance evaluation was performed based on k=10. To summarise the results, an average over the accuracy values in each fold is calculated.

For benchmarking the performance of the framework against the tools, 90% of the transcript sequences were used as training set, whereas, remaining 10% were selected as validation set sequences. A balanced number of lncRNA and protein-coding transcript sequences were selected for training and validation sets. The training and validation set data selected in each fold was used for evaluating the performance of the framework and other coding potential computation tools. The k-fold CV performance benchmarking was performed on *A. thaliana* and *Z. mays* RNA-seq derived sequences to evaluate the prediction accuracy of the framework on test set sequences against known CPC tools. The RNA-seq datasets were chosen primarily based on the availability of annotated lncRNA sequences from web databases.

2.11 Performance evaluation with repeated k-fold cross validation

To evaluate the robustness of the framework, its prediction accuracy was bechmarked against other CPC tools using repeated k-fold CV with data shuffling. As discussed in Section 2.10, k-fold CV was performed for five repetitions (i.e. iterations) with shuffled FASTA sequences in each repetition. Randomisation of sequences in each iteration creates unbiased analysis of the data and evaluates the robustness of the tool under comparison. Five repetitions were performed with 10-fold CV analysis in each repetition. The performance of the framework was compared against CPAT, lncScore, PLEK and CPC2 tools.

2.12 Sub-classification of lncRNAs

lncRNAs are generally classified into seven different types depending on their relationship with protein-coding genes. These can be classified as:

(1) Sense Overlapping Exonic (SOE): exonic regions of lncRNAs transcripts overlapping one or more exons of protein-coding transcript on the same DNA strand,

(2) Sense Overlapping Intronic (SOI): exonic regions of lncRNAs transcripts overlapping one or more introns of protein-coding transcript on the same DNA strand,

(3) Antisense Overlapping Exonic (AOE): exonic regions of lncRNAs transcripts overlapping one or more exons of protein-coding transcript on the opposite DNA strand,

(4) Antisense Overlapping Intronic (AOI): exonic regions of lncRNAs transcripts overlapping one or more introns of protein-coding transcript on the opposite DNA strand,

(5) Antisense lncRNA: Those originating from the antisense strand of DNA that may or may not overlap the protein-coding sequences,

(6) Intergenic lncRNA: Those which are transcribed and expressed between two protein-coding genes, and

(7) Bidirectional Promoter: Those lncRNAs which are located on the antisense strand and are transcribed within 1 kilo basepair (kB) of protein-coding gene located on sense strand.

Figure 2.5 shows several different classes of lncRNA sub-classes based on the overlap of exonic and intronic sequence.

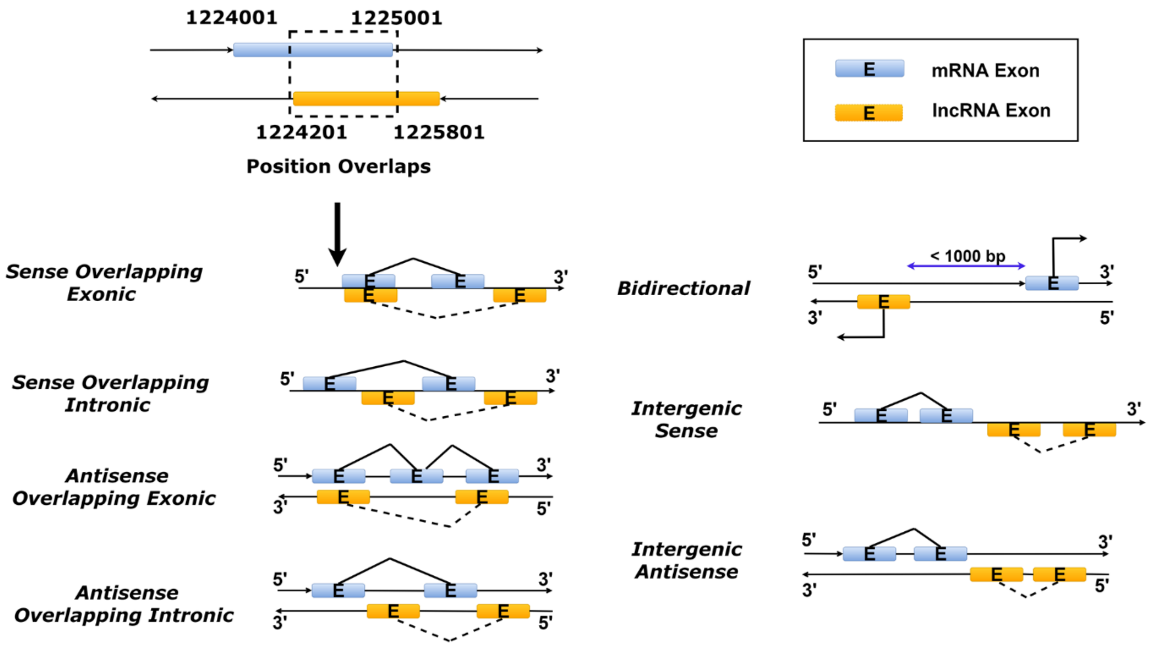


Figure 2.5: Implementation of Position-Based Classification (PBC) strategy lncRNA sub-classification. Sub-classification of lncRNA sequences is performed based on positional coordinates. Sense and antisense-overlapping is performed based on GT-AG exonic and intronic sequences from the ORFs. Intergenic classification is performed by scanning lncRNA sequences between protein-coding genes. Bidirectional lncRNAsequences are classified by finding lncRNAexonic sequences less on 1000 bp from the protein coding exonic sequences.

For the identification of lncRNA sub-classes, a Position-Based Classification (PBC) strategy (Table 2.8) has been developed (Figure 2.5). The method extracts the ORF sequences. The exonic and intronic regions from lncRNAs and protein-coding ORF sequences are extracted based on GT-AG motifs. The sequence overlaps of the exonic (E) and intronic (I) regions of lncRNAs with exonic and intronic regions of mRNAs are obtained by checking for overlaps of the E and I regions based on genomic coordinates. Based on the degree of overlap, sequence alignment is performed that produces an overall score. For identification of seven sub-classes, seven respective rules were developed which helps in identification of the lncRNA sub-class. The main steps of the sequence alignment mapping algorithm in classification are as follows:

1. Extraction of ORFs from lncRNA and mRNA transcripts based on start and stop codons
2. Extraction of E and I regions of ORF sequences of lncRNA and mRNA transcripts
3. For SOE classification, the start and end coordinates of each exon of lncRNA sequence is scanned over genomic coordinates of every single mRNA exon. If one or more E

regions of lncRNA overlaps with E regions of mRNA on ‘+’ strand lying on the same chromosome, it is classified as SOE.

4. For SOI classification, if one or more E regions of lncRNA overlaps with one of more I regions of mRNA on ‘+’ strand lying on the same chromosome, it is classified as SOI.
5. For AOE classification, every single E region in lncRNA sequence located on ‘-’ strand is scanned over every single E region of mRNA on ‘+’ strand. If one or more E regions of lncRNA overlaps with one or more E regions of mRNA lying on the same chromosome, it is classified as AOE.
6. For AOI classification, every single E region in lncRNA sequence located on ‘-’ strand is scanned over every single I region of mRNA on ‘+’ strand. If one or more E regions of lncRNA overlaps with one or more E regions of mRNA lying on the same chromosome, it is classified as AOE.
7. For antisense lncRNA classification, if the sequence lies on the ‘-’ strand, it is broadly classified as ‘antisense_RNA’.
8. For intergenic classification, if the lncRNA transcript sequence lies between the genomic coordinates of two mRNA transcript sequences such that the $\mathrm{start}^{\mathrm{lncRNA}} >$e $\mathrm{end}^{\mathrm{pr\ \ v\_mRNA}}$ and $\mathrm{end}^{\mathrm{lncRNA}} < \mathrm{start}^{\mathrm{next\_mRNA}}$ in a sorted mRNA sequence array, it is classified as intergenic sequence.
9. For bidirectional classification, if the start coordinate of lncRNA first exonic sequence on antisense strand (‘-‘) lies within 1000 bp away from start coordinate of the first exonic sequence of mRNA on sense strand (‘+’) such that $\mathrm{lncRNA}_{\mathrm{firstE}}^{\mathrm{startCoor}} <$ $\mathrm{mRNA}_{\mathrm{firstE}}^{\mathrm{startCoor}}$ and $\mathrm{lncRNA}_{\mathrm{firstE}}^{\mathrm{startCoor}} > \mathrm{mRNA}_{\mathrm{firstE}}^{\mathrm{startCoor}} - 1000$, then the sequence is considered as bidirectional.
10. The overlaps of SOE, SOI, AOE and AOI are based on the following four conditions:
    (1) $\mathrm{lncRNA}_{\mathrm{startPos}} \geq \mathrm{mRNA}_{\mathrm{startPos}}$ and $\mathrm{lncRNA}\ \ _{\mathrm{dPos}} \leq \mathrm{mRNA}\ \ _{\mathrm{dPos}},$
    (2) $\mathrm{lncRNA}_{\mathrm{startPos}} \leq \mathrm{mRNA}_{\mathrm{startPos}}$ and $\mathrm{lncRNA}\ \ _{\mathrm{dPos}} \geq \mathrm{mRNA}\ \ _{\mathrm{dPos}},$
    (3) $\mathrm{lncRNA}_{\mathrm{startPos}} \leq \mathrm{mRNA}_{\mathrm{startPos}}$ and $\mathrm{lncRNA}\ \ _{\mathrm{dPos}} \leq \mathrm{mRNA}\ \ _{\mathrm{dPos}}$ and
    $\mathrm{lncRNA}\ \ _{\mathrm{dPos}} \geq \mathrm{mRNA}_{\mathrm{startPos}},$
    (4) $\mathrm{lncRNA}_{\mathrm{startPos}} \geq \mathrm{mRNA}_{\mathrm{startPos}}$ and $\mathrm{lncRNA}\ \ _{\mathrm{dPos}} \geq \mathrm{mRNA}\ \ _{\mathrm{dPos}}$ and
    $\mathrm{lncRNA}_{\mathrm{startPos}} \leq \mathrm{mRNA}\ \ _{\mathrm{dPos}}.$

Table 2.8: Algorithm for the implementation of PBC approach for lncRNA sub-classification.

Algorithm 4: Implementation of PBC approach for lncRNA sub-classification

Input: noncoding: CSV file containing multiple lncRNA sequences and genomic annotation
coding: CSV file containing multiple protein-coding sequences and genomic annotation

Output: Genomic annotation text file with sub-class annotation for lncRNA sequence

```
codingList ← Load coding file
noncodingList ← Load noncoding file
codingSeqs ← Extract sequences from the codingListe
noncodingSeqs ← Extract sequences from the noncodingListe
codingAnnot ← Extract annotation information from the codingListe
noncodingAnnot ← Extract annotation information from the noncodingListe
orfCoding ← Extract ORF for each sequence from codingSeqse
orfNoncoding ← Extract ORF for each sequence from noncodingSeqs
orfCodingCoord ← Store start and end coordinates of each ORF using information from codingAnnote
orfNoncodingCoord ← Store start and end coordinates of each ORF using information from noncodingAnnote
codingEI ← Extract Exon-Intron boundaries from each ORF sequence in orfCoding
noncodingEI ← Extract Exon-Intron boundaries from each ORF sequence in orfNoncoding
cE ← Using codingEI and orfCodingCoord lists, extract coordinates of exons
cI ← Using codingEI and orfCodingCoord lists, extract coordinates of introns
nE ← Using noncodingEI and orfCodingCoord lists, extract coordinates of exons
nI ← Using noncodingEI and orfNoncodingCoord lists, extract coordinates of introns
annotationList ← Empty annotation list
for i in nE do:
    for j in cE do:
        if chromosome = cchromosome and strand = cstrand then:
            $(\mathrm{nE[i]}^{start} \le \mathrm{cE[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \ge \mathrm{cE[j]}^{end})$ or $(\mathrm{nE[i]}^{start} \ge \mathrm{cE[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \le \mathrm{cE[j]}^{end})$ or $(\mathrm{nE[i]}^{start} \le \mathrm{cE[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \le \mathrm{cE[j]}^{end}$ and $\mathrm{nE[i]}^{end} \ge \mathrm{cE[j]}^{start}$
            annotationList ← "Sense Overlap Exonic"
        end if
    end for
end for
for i in nE do:
    for j in cI do:
        if chromosome = cchromosome and strand = cstrand then:
            $(\mathrm{nE[i]}^{start} \le \mathrm{cI[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \ge \mathrm{cI[j]}^{end})$ or $(\mathrm{nE[i]}^{start} \ge \mathrm{cI[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \le \mathrm{cI[j]}^{end})$ or $(\mathrm{nE[i]}^{start} \le \mathrm{cI[j]}^{start}\ and\ \mathrm{nE[i]}^{end} \le \mathrm{cI[j]}^{end}$ and $\mathrm{nE[i]}^{end} \ge \mathrm{cI[j]}^{start}$
            annotationList ← "Sense Overlap Intronic"
        end if
    end for
end for
for i in nE do:
    for j in cE do:
        if chromosome = cchromosome and strand ! = cstrand then:
            $(nE[i]^{starte} \leq cE[j]^{start}\ and\ nE[i]^{ende} \geq cE[j]^{end})$ or $(nE[i]^{starte} \geq cE[j]^{start}\ and\ nE[i]^{end} \leq cE[j]^{end})$ or $(nE[i]^{start} \leq cE[j]^{start}\ and\ nE[i]^{end} \leq cE[j]^{end}$ and $nE[i]^{end} \geq cE[j]^{starte}$
            annotationList ← “Antisense Overlap Exonic”
        end if
    end for
end for
for i in $nE$ do:
    for j in $cI$ do:
        if chromosome =chromosome and strand ! =strand then:
            $(nE[i]^{start} \leq cI[j]^{start}\ and\ nE[i]^{end} \geq cI[j]^{end})$ or $(nE[i]^{start} \geq cI[j]^{start}\ and\ nE[i]^{end} \leq cI[j]^{end})$ or $(nE[i]^{start} \leq cI[j]^{start}\ and\ nE[i]^{end} \leq cI[j]^{end}$ and $nE[i]^{end} \geq cI[j]^{starte}$
            annotationList ← “Antisense Overlap Intronic”
        end if
    end for
end for
for i in $noncodingAnnot$ do:
    if strand = $'-'$ then:
        annotationList ← “Antisense lncRNA”
    end if
end for
for i in $noncodingAnnot$ do:
    for j in $codingAnnot$ do:
        if $noncodingAnnot[i]^{start} > codingAnnot[j-1]^{end}$ and $noncodingAnnot[i]^{end} < codingAnnot[j+1]^{starte}$
            annotationList ← “Intergenic lncRNA”
        end if
    end for
$nEL$ ← Extract the coordinates of the last exon from each lncRNA sequence
for i in $nEL$ do:
    for j in $cE$ do:
        if chromosome =chromosome and strand ! =strand then:
            if $nEL[i]^{start} < cE[j]^{start}$ and $nEL[i]^{start} > cE[j]^{start} - 1000$ then:
                annotationList ← “Bidirectional Promoter”
            end if
        end if
    end for
end for
```

2.13 lncRNA function prediction analysis

To predict the functions of lncRNAs obtained from the lncRNA identification process using iRF classifier in the RNA-seq datasets, three input parameters are required:

- LncRNA-protein interaction data
- Protein-protein interaction data

- Protein gene ontology enrichment data

The methodology for obtaining individual dataset is presented in following sections.

2.13.1 Identification of lncRNA-protein interactions

Unlike the mammalian species such as *H. sapiens* and *M. musculus* for which known interactions are available in the genomic databases (NPInter (Wu *et al.*, 2006)), known regulatory interactions between the lncRNAs and proteins are currently unknown in plant species. Therefore, to identify potential interactions of the lncRNAs, known interations of lncRNA and protein-coding genes were obtained from the NPInter database (Wu *et al.*, 2006). Inspired by the work of Liu et al. (2017), lncRNA sequence similarity and protein sequence similarity was performed using Smith-Waterman pairwise-sequence alignment (Pearson, 1991) with match of 2, mismatch of -1, gap opening of 5 and gap extension of 2. For construction of lncRNA sequence similarity matrix, lncRNA sequences were extracted from the lncRNA-protein interactions in *H. sapiens*. Each lncRNA sequence of *A. thaliana* and *Z. mays* were matched with the lncRNAs of *H. sapiens* to construct a lncRNA sequence similarity matrix. The protein sequence similarity matrix was also constructed in a similar way. Normalisation of the Sequence Similarity Matrix (SSM) was performed by the following function:

$$SSM(S_i, S_j) = \frac{sw(S_i, S_j) - \min\,[sw(S_i, S_j)]}{\max[sw(S_i, S_j)] - \min\,[sw(S_i, S_j)]e} \qquad (1.23)$$

where $sw(S_i, S_j)$ represents pairwise alignment score of sequence i with sequence j. An adjacency matrix was constructed between the lncRNA-protein interaction partners. lncRNA and protein-coding sequences from plants and humans used to construct the similarity matrix were used for constructing the adjacency matrix. If an association is confirmed, it is represented by 1 in the matrix, and otherwise 0.

Based on the proposed work on identification of drug-target interactions by Liu et al. (2016), the Neighbourhood Regularised Logistic Matrix Factorisation (NRLMF) method was applied for the identification of lncRNA-protein interactions. The probability of the lncRNA-protein interaction is modeled by the following logistic function:

$$p_{ij} = \frac{\exp\,(u_i \;\; {}_j^T)e}{1 + \exp\,(u_i v_j^T)} \qquad (1.24)$$

where $u_{ie}$and v_j are latent vectors of lncRNAs and proteins represented by $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{n \times r}$ respectively, where $u_{ie}$is the ith row in $\mathbf{U}$ and v_j is the jth row in $\mathbf{V}$.

The lncRNA-protein interaction pairs were further filtered based on correlation of FPKM expression values Pearson Correlation Matrix (PCC) ≥ -0.5 computed using Pearson Correlation method (Williams, 1996). The generated similarity matrix is represented by LncRNA-Protein Interaction (LPI) matrix. For computing LPI pairs in Arabidopsis and Maize datasets, a representative random subset of 50 lncRNAs and 402 protein-coding genes were selected for analysis against random subset of 50 ncRNA and 400 protein-coding genes in *H. sapiens* derived from NPInter database. Representative subset was selected in order to remove biasness against selection of data and speed up the computation time for the calculation of SSMs.

2.13.2 Identification of protein-protein regulatory interactions

For identification of interactions between protein-coding transcripts, proteins predicted to be interacting with lncRNAs were used for identifying the protein interaction pairs. The Protein-Protein Interactions (PPI) deposited in the STRING database have been used for inferring interactions of protein-coding transcripts with other proteins. For identifying the PPIs, the following steps have been undertaken:

1. Using the LPI matrix constructed in Section 2.12.1, protein-coding transcript IDs were extracted.
2. Known protein interactions were obtained from the STRING database consists of three columns: Column 1: Interacting protein 1, Column 2: Interacting protein 2 and Column 3: strength of interaction.
3. Protein-coding transcript IDs extracted form the LPI matrix were used to match transcript IDs in column 1 so that interactions of these proteins with other proteins could be determined. Finally, the resulting PPI matrix was generated.

2.13.3 Identification of GO enrichment data for proteins

For function prediction of lncRNAs, an important component required is the gene annotation process which associates gene ontology ID during the prediction process. Therefore, to obtain this component, GO annotation of transcripts in plants has been obtained from Ensembl Plants (Zerbino *et al.*, 2017). To construct the GO annotation file, LPI and PPI matrices were concatenated to generate a LPI-PPI matrix. Protein-coding transcript IDs were extracted from the generated matrix and duplicate IDs were removed. Resulting IDs were used as primary column for extracting GO annotation.

2.13.4 Function prediction of lncRNAs using Bayesian Markov Random Fields method

To predict the functional association of lncRNAs, we used the BMRF method, which has been previously used for the prediction of protein functions of unannotated proteins (Kourmpetis *et al.*, 2010). BMRF is originally based on MRF approach (Deng *et al.*, 2002) where the nodes are coloured and encoded in the binary vector with $\mathrm{X}_{ie}= 1$ if the i^{th} protein performs that particular function and $\mathrm{X}_{ie}= 0$ if the i^{th} protein does not possess any function. We substituted lncRNAs where $\mathrm{X}_{ie}= 0$ for i^{th} protein not having functions in the network. Therefore, the probability of state x of the network was defined by:

$$P(\boldsymbol{x}|\theta) = \frac{1}{Z(\theta)} \exp(-U(\boldsymbol{x},\theta)) \qquad (1.25)$$

where $Z(\theta)$ is normalising constant which depends on θ, and $-U$ is energy function. The energy function U can be in the following way for homogenous second order MRF:

$$U(\boldsymbol{x}|\theta) = \mathrm{e}\sum_{i=1\mathrm{e}}^{N\mathrm{e}} G_1(x_i) + \mathrm{e}\sum_{i=1}^{N} \sum_{j=i+1\mathrm{e}}^{N\mathrm{e}} G_2(x_i, x_j) \qquad (1.26)$$

where G_1 and G_2 are problem-dependent functions. G_1 takes one value for per state of the network such that $G_1(1) = \alpha$ and $G_1(0) = 0$. The function G_2 becomes zero if lncRNA and proteins do not interact. The Pseudo-Likelihood Function (PLF) is the product of the conditional probabilities across the nodes in the network. The PLF is computed by the following equation:

$$PLF(\boldsymbol{x}|\alpha, \beta^1, \beta^0) \quad \prod_{i=1}^{N} P(x_i|\mathbf{x}_{-i}, \alpha, \beta^1, \beta^0). \qquad (1.27)$$

where $\mathbf{x}_{-i}$ denotes $\mathbf{x}$ without the i^{th} element. PLF possesses properties similar to a full-likelihood function and therefore helps in determining the logistic equation by setting pseudo-score to zero. PLF outperforms over the full-likelihood function as the latter has an intractable normalising constant. The conditional probability of an unannotated lncRNA i is given by a logistic function $(1 + exp(-v_i))^{-1}$. Each state of the unannotated lncRNA is sampled using the logistic function. Once the PLF is computed, Gibbs-Sampling (GS) is performed by iterating over all the states of the unannotated lncRNA sequences. In each iteration, t elements of lncRNA are updated conditionally with parameter values corresponding to α, β^0 and β^1 which are updated conditionally using the Differential Evolution Markov Chain (DEMC) method. This process is repeated until convergence is reached. The Bayesian process averages across the unknown lncRNAs (i.e. lncRNAs with unknown functions) and models the joint posterior distribution of the model parameters and functional states of the lncRNAs which are unannotated. It samples from the joint distribution using the Markov Chain Monte Carlo (MCMC) method (Geyer, 1991).

Therefore, to predict and associate function using the BMRF method, following steps have been performed:

1. Read in the LPI-PPI matrix and protein GO annotation files (Figure 4.6).
2. Execute BMRF for predicting GO annotations of unannotated lncRNAs.
3. Using lncRNA IDs, match the IDs from the output file to extract GO annotations.
4. Filter the lncRNA annotation file by probability value with probability ≥ 0.8. Although the probability cutoff value can be chosen by the user, value of 0.8 is recommended for extracting predicted lncRNA genes having higher probabilities.
5. Using the protein GO annotation file, match the GO IDs of filtered lncRNA annotation file with protein GO annotation file to obtain function description associated with each GO term.

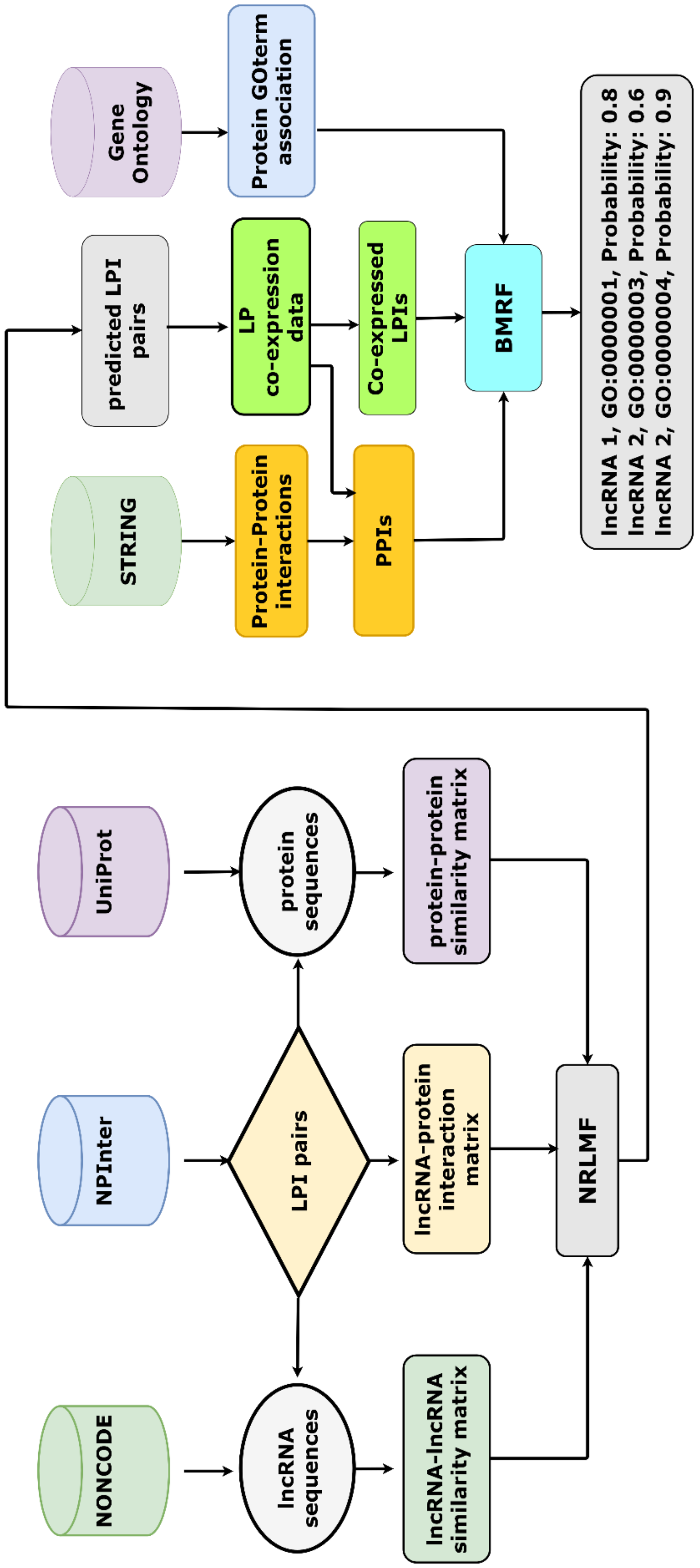

NONCODE
NPInter
UniProt
lncRNA sequences
LPI pairs
protein sequences
lncRNA-lncRNA similarity matrix
lncRNA-protein interaction matrix
protein-protein similarity matrix
NRLMF
STRING
predicted LPI pairs
Gene Ontology
Protein-Protein interactions
LP co-expression data
Protein GOterm association
PPIs
Co-expressed LPIs
BMRF
lncRNA 1, GO:0000001, Probability: 0.8
lncRNA 2, GO:0000003, Probability: 0.6
lncRNA 2, GO:0000004, Probability: 0.9

Figure 2.6: Workflow of the lncRNA function prediction NRLMF-BMRF model.

2.13.5 Function prediction of DE lncRNA genes

To determine the functions of lncRNAs which are DE in RNA-seq time-series samples, lncRNA transcript IDs were extracted from the interaction/co-expression matrix. The IDs were matched with transcript IDs in the Cuffdiff gene expression results file. The Cuffdiff results file consists of the following information for each transcript ID: gene name, transcript ID, genomic coordinate, $\log_2$ Fold Change ($\log_2$FC) value, p-value, q-value and significance status.

Cuffdiff results were filtered to keep transcript ID having q-value (i.e. False Discovery Rate (FDR) value) ≤ 0.05 and $\log_2$FC > 1 and < -1. $\log_2$ Fold Change provides measure of genes significantly up or down-regulated during gene expression process. $\log_2$FC values greater than 1 and less than -1 demonstrates significant up and down-regulation of genes. lncRNA transcript IDs were matched with filtered DEGs to obtain DE lncRNAs for each sample pair analysis. Using filtered GO annotation, PPI and LP interaction/co-expression, molecular functions and GO terms with associated probability was annotated using BMRF.

2.13.6 Gene filtering based on the published experimental data

Results from the NRLMF-BMRF analysis were further filtered based on the published experimental data. A summary of experimentally reported lncRNA-function association data in plants (Liu *et al.*, 2015) (Table 2.9) were used for validation of the results. The function annotation results were filtered based on the dictionary of keywords which were extracted from the experimentally-derived lncRNA regulatory functions.

Table 2.9: Summary of known lncRNA genes in plants.

Gene name	Species	Biological function	Regulatory mechanism	Reference
COLDAIR	*A. thaliana*	Flowering time	Histone modification	(Heo and Sung, 2011)
COOLAIR	*A. thaliana*	Flowering time	Promoter interference	(Swiezewski *et al.*, 2009; Csorba *et al.*, 2014)
LDMAR (P/TMS12-1)	*O. sativa*	Fertility	Promoter methylation	(Ding *et al.*, 2012)
HID1	*A. thaliana*	Photomorphogenesis	Chromatin association	(Wang *et al.*, 2014)
IPS1	*A. thaliana*	Phosphate homeostasis	Target mimicry	(Franco-Zorrilla *et al.*, 2007)
Cis-NAT$_{PHO1;2}$	*O. sativa*	Phosphate homeostasis	Translational enhancer	(Jabnoune *et al.*, 2013)
OsPI1	*O. sativa*	Phosphate homeostasis	Unknown	(Wasaki *et al.*, 2003)
TPS11	*S. lycopersicum*	Phosphate homeostasis	Unknown	(Liu, Muchhal and Raghothama, 1997)
asHSFB2a	*A. thaliana*	Vegetative and gametophytic development	Antisense transcription	(Wunderlich, Groß-Hardt and Schöffl, 2014)
ASCO-lncRNA	*A. thaliana*	Lateral root development	Alternative splicing regulators	(Bardou *et al.*, 2014)
APOLO	*A. thaliana*	Auxin-controlled development	Chromatin loop dynamics	(Ariel *et al.*, 2014)

To filter the lncRNA functions predicted by BMRF, a filter-based approach was constructed. Table 2.10 illustrates the algorithm for filtering the genes based on the known regulatory mechanisms.

Table 2.10: Algorithm for filtering the lncRNA gene functions.

Algorithm 4: Algorithm for filtering the lncRNA gene functions

Input: annotationFile: Input TXT file containing predicted functions, GOTerms and associated with probability values for each lncRNA gene.
expAnnotation: Input TXT file containing list of experimentally confirmed lncRNA regulatory annotation data.
Output: output: Output TXT file containing filtered prediction results.

```
$\mathrm{df} \leftarrow \mathrm{annotationFilee}$
```

```
dfe ← expAnnotatione
dfFunction ← df['Function']
dfGene ← df['genename']
dfeList ← convert the dfe dataframe to list
keyDfe ← split each word in dfeList by space
newFun, newGen ← Initialize lists
ignoreList1 ← construct a list of words containing punctuations, frequent words (e.g. a, and, the, or, nor, it, etc.).
ignoreList2 ← construct a list of non-relevant words (e.g. located, constituent, situated, composed, found, find, etc.).
for i=0 to length(keyDfe) do
    if keyDfe[i] in ignoreList1 then
        continue
    end if
    else
        keyDfeF ← append keyDfe[i]
end for
 for i=0 to length(keyDfeF) do
     for j=0 to length(dfFunction) do
        if keyDfeF[i] in dfFunction[j] then
            newFun ← dfFunction[j]
            newGen ← dfGene[j]
        end if
        else
            continue
    end for
end for
newList ← Join newGen and newFun lists
for i=0 to length(ignoreList2) do
     if ignoreList2[i] in newList then
            Remove element from newListe
     end if
end for
output ← Save the newList file
```

The main steps of the algorithm are as follows:

1. The algorithm requires input function prediction file generated by BMRF analysis (annotationFile) and an input experimental annotation file (expAnnotation) containing the list of regulatory function mechanisms in text format. Each regulatory mechanism should be separated by a newline.
2. Load the annotationFile and expAnnotation input files into df and dfe tables.
3. Extract the 'Function' column from the df table and store it into dfFunction list.
4. Extract the 'genename' column from the df table and store it into dfGene list.
5. Convert the dfe table to a list and store the elements in dfeList list.

6. Split each element of the dfeList to individual words by space delimiter and store the elements in keyDfe list.
7. Construct a list ignoreList1 containing articles, punctuations, frequent words (such as: and, the, or, if, by, for, when, in, etc.).
8. Construct another list ignoreList2 containing non-relevant words (such as located, constituent, situated, composed, found, find, etc.).
9. Match each element of the keyDfe list to the ignoreList elements. If the element matches one of the elements, then continue the loop, else, append the current element of the keyDfe to keyDfeF list.
10. Run a loop over the elements of the keyDfeF list. Within this loop, run another loop over the individual elements of dfFunction list. Check whether each element of the keyDfeFe list can be found in dfFunction elements. If the element occurs, then append the current element of dfFunction to newFun list, and dfGene to newGen list. If it does not match, then continue the loop to the next iteration.
11. Join the newGen and newFun lists as the newList file.
12. Run a loop through each element of ignoreList2. Match each element of ignoreList2e with individual elements of newList. If the element of ignoreList2 is found, then remove that element from the newList, else continue to next element.
13. Save the modified newList as output file.

The output file contains the filtered list of the lncRNA genenames and its corresponding regulatory functions.

2.14 Visualisation of lncRNA sequences using D3.js Javascript library

The web-based visualisation application (D3VizRNA) is a client-side application which is constructed using D3.js Javascript library and is used for viewing the positions of predicted lncRNA and mRNA transcript sequences obtained from prediction using the iRF classifier. The visualisation application also provides information of the genomic location of lncRNA and mRNA genes in the genome; for example, if the lncRNA gene is intergenic, sense-intronic, sense-exonic, bidirectional, etc. The prediction results are stored in a Comma Separated Values (CSV) format file which is then used by the D3.js to generate the visualisation application. For generating the graphical visualisation, the CSV file requires the following input fields:

1. Chromosome name (chrName)

2. Chromosome start value (chrStart)
3. Chromosome end value (chrEnd)
4. Start position of the gene (geneStart)
5. End position of the gene (geneEnd)
6. Width of the position (geneWidth)
7. Position of the gene on y-axis (geneYaxis)
8. Height of the gene (geneHeight)
9. Gene name (geneName)
10. Gene type (geneType)
11. Gene Function (geneFunction)

The D3.js based application requires the creation of the following fields before constructing the visualisation: chrName, chrStart, chrEnd, geneWidth, geneYaxis and geneHeight. These fields are created by a Python-based program which formats the input coordinates data into D3.js compatible format.

The output data resulting from the lncRNA identification, classification and prediction steps is used as input into the Python-based application (Table 2.11) for producing D3.js coordinate file format. For constructing the D3.js file format, the application requires the following fields in CSV format:

1. Chromosome name
2. Start position of the gene
3. End position of the gene
4. Strand value
5. Gene type
6. Gene function

Table 2.11: Input file format of the genes obtained from the classification and prediction analysis.

```
chromosome,start,end,strand,gene_type,gene_function
1,3899,4670,+,protein_coding,glucosinolate biosynthesis
1,4589,8900,-,protein_coding,metabolic function
2,230078,256789,+,sense overlapping lncRNA,DNA damage repair
2,347897,348690,-,Antisense overlapping lncRNA,Functions in chromosome organisation
2,567845,567904,-, protein_coding,biosynthetic function
```

The main steps of the application (Table 2.12) are as follows:

1. Input coordinates and genome index files are loaded in CSV format as data frames.
2. Individual column values are stored as list vectors.
3. Width of each gene sequence is calculated by taking the difference of End value minus Start value
4. Based on the chromosome field, chromosome start and end values are stored in separate vectors
5. Two identical lists $\mathrm{dictList1}$ and $\mathrm{dictList2}$ are created consisting of following format. [{‘chromosome’: 1, ’chrStart’:0, ’chrEnd’:500000, ’start’: 3899, ’end’: 4670, ’strand’:’+’, ’gene_type’:’protein_coding’, ’gene_function’:’glucosinolate biosynthesis’, ’yaxis’:30},{‘chromosome’:1, ’chrStart’:0, ’chrEnd’:500000, ’start’:4589, ’end’:8900, ’strand’:’-’, ’gene_type’:’protein_coding’, ’gene_function’:’metabolic function’, ’yaxis’:30}]. Fixed y-axis values are appended to each gene.
6. An empty list $\mathrm{dictList3}$ is created.
7. An empty list vector $\mathrm{countArray}$ is created. Count is performed and stored for each gene by matching coordinates of each gene sequence in $\mathrm{dictList1}$ to all elements in $\mathrm{dictList2}$e based on the following five conditions:

   1) If $\mathrm{startPosition}_{\mathrm{gen}}^{\mathrm{dictList1}}$ =e$\mathrm{startPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{gen}}^{\mathrm{dictList1}}$ =e $\mathrm{endPosition}_{\mathrm{g\ n\ se}}^{\mathrm{dictList2}}$ and $\mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList1}} = \mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList2}}$

   2) If $\mathrm{startPosition}_{\mathrm{g\ n}}^{\mathrm{dictList1}} \leq \mathrm{startPosition}_{\mathrm{g\ n\ s}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{g\ n}}^{\mathrm{dictList1}} \geq \mathrm{endPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList1}} = \mathrm{strand}_{\mathrm{g\ n}}^{\mathrm{dictList2}}$, push 1 to $\mathrm{countArray}[i]$,

   3) If $\mathrm{startPosition}_{\mathrm{gen}}^{\mathrm{dictList1}} \geq \mathrm{startPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{gen}}^{\mathrm{dictList1}}$ ≤e $\mathrm{endPosition}_{\mathrm{g\ n\ s}}^{\mathrm{dictList2}}$ and $\mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList1}} = \mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList2}}$, push 1 to $\mathrm{countArray[i]}$,

   4) If $\mathrm{startPosition}_{\mathrm{g\ n}}^{\mathrm{dictList1}} \leq \mathrm{startPosition}_{\mathrm{g\ n\ s}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{g\ n}}^{\mathrm{dictList1}} \geq \mathrm{startPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{gen}}^{\mathrm{dictList1}} < \mathrm{endPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList1}} = \mathrm{strand}_{\mathrm{g\ n}}^{\mathrm{dictList2}}$, push 1 to $\mathrm{countArray[i]}$

   5) If $\mathrm{startPosition}_{\mathrm{gen}}^{\mathrm{dictList1}} \geq \mathrm{startPosition}_{\mathrm{gen\ se}}^{\mathrm{dictList2}}$ and $\mathrm{endPosition}_{\mathrm{gen}}^{\mathrm{dictList1}}$ >e $\mathrm{endPosition}_{\mathrm{g\ n\ s}}^{\mathrm{dictList2}}$ and $\mathrm{startPosition}_{\mathrm{g\ n}}^{\mathrm{dictList1}} < \mathrm{endPosition}_{\mathrm{g\ n\ s}}^{\mathrm{dictList2}}$and $\mathrm{strand}_{\mathrm{g\ n}}^{\mathrm{dictList1}}$ =e $\mathrm{strand}_{\mathrm{gen}}^{\mathrm{dictList2}}$, push 1 to $\mathrm{countArray[i]}$
8. An empty list vector $\mathrm{countArray1}$ is created.
9. Values in $\mathrm{countArray}$ are added for each sequence and stored in $\mathrm{countArray1}$e
10. Y-axis values are updated in $\mathrm{dictList1}$ by adding 10 times $\mathrm{countArray1}$ for each yaxis value in

each sequence

11. Updated y-axis values, genomic coordinates and annotation data are stored in dictList3e
12. dictList3 is exported to CSV file which is then used as input for the visualisation application

Table 2.12: Algorithm for formatting the genomic coordinates of the genes.

```
Algorithm 6: Implementation of gene coordinates formatting algorithm
Input: inputCoordinates: input CSV file containing genomic coordinates for each transcript
sequence.
chromosomeIndex: input CSV file containing chromosome lengths
Output: output: output CSV file containing genomic coordinates compatible for visualisation
df ← inputCoordinatese
chrdf ← chromosomeIndexe
start ← row values of start position from dfe
end ← row values of end position from dfe
strand ← row values of end position from dfe
geneName ← row values of gene names from dfe
geneType ← row values of gene types from dfe
geneFunction ← row values of gene functions from dfe
chrStart ← 0e
height ← 5e
width ← list vectore
for i = 0 to length(start) do
    width ← end[i] − start[i]
end for
chrEnd ← row values of chromosome lengths from chrdfe
dictList1, dictList2 ←
{chrStart, chrEnd, start, end, width, height, yaxis, strand, geneName, geneType, geneFunction}
dictList3 ← empty dictionary
countArray ← list vectore
for i = 0 to length(dictList1) do
    for j = 0 to length(dictList2) do
        if strand = strand and start = start and end = end then
            continue
        end if
        if strand = strand and $start(i)^{dictList1} \leq start(j)^{dictList2}$ and $end(i)^{dictList1} \geq start(j)^{dictList2}$ then
            countArray ← 1e
            continue
        end if
        if strand = strand and $start(i)^{dictList1} \geq start(j)^{dictList2}$ and $end(i)^{dictList1} \leq start(j)^{dictList2}$ then
            countArray ← 1e
            continue
        end if
        if strand = strand and $start(i)^{dictList1} \leq start(j)^{dictList2}$ and $end(i)^{dictList1} > start(j)^{dictList2}$ and $end(i)^{dictList1} < end(j)^{dictList2}$ then
```

```
                countArray ← 1e
                continue
            end if
            if strand = strand and $\text{start(i)}^{\text{dictList1}} \geq \text{start(j)}^{\text{dictList2}}$ and $\text{end(i)}^{\text{dictList1}} > \text{end(j)}^{\text{dictList2}}$ and $\text{start(i)}^{\text{dictList1}} < \text{end(j)}^{\text{dictList2}}$ then
                countArray ← 1e
                continue
            end if
        end for
    end for
    countArray1 ← sum values in each sequencee
    dictList3 ← {chrStart, chrEnd, start, end, width, height, yaxis + 10 * $\text{countArray1}^{\text{s quenc}}$, strand, geneName, geneType, geneFunction}
    output ← save dictList3 to CSV filee
```

The resulting output file is then used as an input file for visualisation of lncRNAs in the genome. The Python script generates output files for individual chromosomes. Table 2.13 provides the algorithm for construction of lncRNA sequences using rectangles and viewing functionality of the sequences using D3.js. An "index.html" file is constructed which acts as Graphical User Interface (GUI) for accessing the viewing functionalities defined in the HTML < script > tag.

Table 2.13: Algorithm for construction of lncRNA visualisation.

Algorithm 7: Implementation of lncRNA visualisation algorithm using D3.js

Input: *inputCoordinates*: input chromosome CSV files containing genomic annotation of lncRNA sequences formatted using Algorithm 3.

Scripts: d3.v4.min.js

Output: *output*: output HTML ‘index.html’ file displaying visualisation of lncRNA sequences of individual chromosomes

```
Insert < div > attributes for holding visualisation chart and chromosome selection button
Attach event listener function displayChromosomeView() to option button
Call displayChromosomeView() to trigger the event when the webpage loads
data ← CSV file using d3.csv()e      # Load input CSV file using d3.csv() function
Call chromosome 1 (chr1.csv) CSV file during initial webpage loading
Define SVG dimensions (width and height)
Define menu object containing annotation information from data array
Add SVG element using d3.select("svg")
Define width and height of the visualisation window
Define x, y, x2, y2 variables as scales for constructing the visualisation using d3.scaleLinear() function which takes range of input values lying between 0 and width for x axis and 0 and height for y axis.
Define xAxis, xAxis2 and yAxis variables using d3.axisBottom(x), d3.axisBottom(x2) and d3.axisLeft(y)
Define brush using d3.brushX() function which constructs a small static navigation panel for the visualisation
Define zoom variable using d3.zoom() function which provides zoom functionality by calling zoomed() function
```

14: Define focus and context variables to the svg as class focus and class context.

15: Define domains for the x and y axes using $.\mathrm{domain}()$ function. The x domain takes the chromosome length whereas the y domain takes the minimum and maximum values as range from the yaxis variable from data

16: Define rect variable and append "rect" to SVG with class zoom which xAxis and zoome functions

17: Append "rect" to the focus group class which contains the data for the constructing rectangles for the lncRNA genes. The $\mathrm{x, y, width\ and\ height}$ attributes constructs the rectangles for the genes. Annotation is appended using $\mathrm{d3.contextMenu(menu)}$ function. Gene types are coloured individually. Colours are displayed on the visualisation for each gene using $\mathrm{mouseover}$ and $\mathrm{mouseout}$ functions

18: The rectangle attributes for each lncRNA gene are added to the x and y axes using xAxis function. The $\mathrm{zoom}()$ function is called using $.\mathrm{call(zoom)}$

19: Focus and context group classes are added to the "$\mathrm{axis\ axis} - -\mathrm{x}$" and "$\mathrm{axis\ axis} - \mathrm{e}$ $-\mathrm{y}$" classes and $\mathrm{xAxis, xAxis2\ and\ yAxis}$ variables are called using $.\mathrm{call(xAxis)}, .\mathrm{call(yAxis)}$ and $.\mathrm{call(xAxis2)}$

20: Context group class appends the brush class and calls $\mathrm{brushed}()$ function which provides navigation functionality within $\mathrm{x.range}()$

21: The $\mathrm{brushed}()$ function is defined which implements event listener using $\mathrm{d3.event.sourceEvent}$ function which listens to the zoom function and zoom event trigger.

22: The s variable is defined which contains the selection values from the x axis using $\mathrm{d3.event.selection}$ function. The range of values selected on the $\mathrm{x2}$ axis is passed to the $\mathrm{d3.event.selection}$ which is stored in s variable. The selection is then passed to the focus group class which is further passed to the SVG element which scales and translates the rectangles

23: The $\mathrm{zoomed}()$ function is defined using $\mathrm{d3.event.sourceEvent}$ function which listens to the brush function and brush event trigger. The zoom transforms the rectangles using $\mathrm{d3.event.transform}$ function and rescales the x domain using $.\mathrm{rescaleX(x2).domain}()$ function.

24: When zoom function is called, context class calls $\mathrm{brush.move}$ function with $\mathrm{x.range}()$ which is mapped to the domain using $.\mathrm{map}$ function.

25: To complete the zoom functionality, step 15 is copied to the $\mathrm{zoomed}()$ function which zooms the rectangles with transform and translate functionalities, annotation and colour attributes.

2.15 Summary

This chapter started by summarizing the contents of previous chapter. A short methodology workflow was outlined to describe the contents briefly. It then described various methods for lncRNA identification, classification, prediction and visualisation used in the research which led to the formulation of the research questions addressing significant research gaps. Methods used at each step were explained in detail. The methods used in this research address the research questions and provide a better understanding of lncRNA identification and classification using computational approaches. The next chapter provides the results, analysis and discussion of the identification and classification of lncRNA sequences from Refseq and GENCODE databases.

CHAPTER 3: RESULTS AND ANALYSIS OF LONG NON-CODING RNA CLASSIFICATION

3.1 Introduction

The previous chapter outlined detailed methodology for computational processing and analysis of RNA-seq data. We discussed various tools and statistical methods for obtaining DEGs in RNA-seq data. We also discussed detailed methodology for identification of lncRNAs in Reference and RNA-seq datasets using the iRF classifier. Theoretical details of various features and their implementation in extraction from FASTA sequences were also provided. Furthermore, details about implementation of LASSO method and iRF classifier in feature selection in obtaining optimal features were also discussed. Finally, the methodology for the prediction of functions of lncRNAs in RNA-seq datasets was also presented.

This chapter presents the results obtained using the methodology discussed in Chapter Four, as well as the evaluation of the results using machine learning methods. This chapter focusses on lncRNA identification results from reference datasets, optimisation of the classification approach using the LiRF-FS method and genomic annotation of lncRNA sequences using web-based genomic datasets.

3.2 Reference dataset statistics

Statistical analysis on the transcript sequences obtained from the GENCODE and Refseq datasets shows that in total, 9890 lncRNA and 41219 mRNA transcripts in mammals, whereas 10000 lncRNA and 41219 mRNA sequences in plants species were retrieved (Table 3.1). The statistics show that mammalian transcript sequences contain comparatively higher GC% than plant sequences. The results also show that mammalian sequences are comparatively much larger than plant sequences, which is evident from the maximum length (max len) value. When the mean length is compared, both mRNA and lncRNA sequences have comparable lengths which shows that some of the lncRNA and mRNA transcript sequences have equal lengths. Plant sequences on the other hand, contain higher nucleotide content and nucleotide bases when compared to mammalian sequences which do not contain any nucleotide bases.

Table 3.1: Statistical metrics of reference datasets transcript sequences where TT = Transcript Type; MaxL = Max Length; MeanL = Mean Length; MedianL = Median Length; MinL = Minimum Length; Num_A = Number of A bp; Num_C = Number of C bp; Num_G = Number of G bp; Num_N = Number of N bp; N = Number of bp; NOT_N = Number of bp not N; Num_seq = Number of sequences.

File	TT	GC%	MaxL	MeanL	MedianL	MinL	Num_A	Num_C	Num_G
Mammals	lncRNA	45.3	74456	1209	753	200	3303123	2707171	2713436
	mRNA	49.5	40378	2394	1773	63	6233020	5870791	5985911
Plants	lncRNA	40.8	20103	1350	978	200	15682702	11123101	11566854
	mRNA	44.1	14494	1538	1339	19	17854803	13186523	14786249

File	TT	Num_N	Num_T	Num_bp	NOT_N	Num_seq
Mammals	lncRNA	0	3237737	11961467	11961467	9890
	mRNA	0	5854986	23944708	23944708	10000
Plants	lncRNA	22	17293025	55665704	55665682	41219
	mRNA	2658	17592412	63422689	63419987	41219

Analysis of minimum lengths of lncRNAs and mRNAs shows that the minimum length of lncRNAs is 200, whereas for plants and mammals, the minimum lengths are 19 and 63 of mRNA respectively. The difference in minimum lengths is due to the sequence length cutoff of 200 bp that has been applied on lncRNA sequences to remove non-lncRNA sequences from the analysis. Analysis of nucleotide quantities shows that both mammalian and plant transcript sequences contain higher number of adenine (A), cytosines (C), guanines (G) and thymines (T) in mRNAs when compared to lncRNAs. However, the number of Ts in plant lncRNA sequences is proportional to mRNA sequences.

3.3 lncRNA classification on reference datasets

From reference datasets, mRNA and lncRNA transcript sequences from 10 different species were extracted. Using 73 different ORF-based and codon-bias features described in Chapter 2 Section 2.5, 73 features for each transcript sequence were extracted to construct the feature matrix from which training and test sets were created. Using these training and test sets, classification was performed using RF and iRF classifiers. Classification using RF has been additionally performed for the comparison of the results obtained using iRF. The following section details the results obtained from the classification analysis on all features.

3.3.1 Classification performance evaluation

Using the RF classifier, the performance of the 73 features was measured in 10 species for identification of lncRNAs. According to the metrics discussed in Chapter 2 Section 2.8.3, Accuracy (ACC), Sensitivity (SENS), Specificity (SPEC), Precision (PRES), NPV, F1-Score and MCC have been measured for 10 species obtained from reference datasets (Table 3.2). Results of the classification analysis showed that the prediction performance obtained from RF on 73 features of plant species showed ACC and PRES ≥ 93% with *ZM*, *BNA*, *BRA*, *BOL*, *OS*, *SL* and *ST* achieving ACC and PRES ≥ 95%. For mammalian species *HS* and *MM* demonstrated comparatively lower ACC and PRES as compared to plant species having ACC and PRES values of 91% and 90%, respectively. NPV metrics showed similar performance with differences of 1% in ACC and PRES metrics for all species.

Table 3.2: Classification performance of 73 features using RF classifier.

Species	ACC	SENS	SPEC	PRES	PPV	NPV	F1	MCC
ATH	93.78	94.57	92.97	93.79	93.30	94.29	93.78	0.875
ZM	95.63	94.45	96.84	95.66	96.82	94.48	95.63	0.91
BNA	96.62	95.24	98	96.65	97.96	95.34	96.62	0.93
BRA	96.11	94.98	97.28	96.14	97.31	94.94	96.11	0.92
BOL	96.10	95.61	96.59	96.11	96.46	95.77	96.10	0.92
OS	97.18	97.21	97.16	97.18	97.05	97.31	97.18	0.94
SL	97.30	96.70	97.90	97.31	97.87	96.74	97.30	0.94
ST	96.27	95.34	97.20	96.28	97.15	95.42	96.27	0.92
HS	91.07	89.86	92.37	91.12	92.66	89.48	91.07	0.82
MM	90.13	87.05	93.39	90.34	93.30	87.21	90.13	0.80
6-plants	95.26	94.87	95.64	95.26	95.51	95.01	95.26	0.90
2-mammals	90.57	89.63	91.46	90.57	90.80	90.36	90.57	0.81

F1-score is a weighted average of PRES and SENS, and it showed similar performance as displayed by PRES with only slight differences. SPEC is the measure of identification of true negative rate. Therefore, the results of SPEC showed an overall average of 97% on all plant species with *ATH* as an exception with SPEC of ~93%. Mammalian data also showed SPEC of ~93%. MCC metric illustrates the quality of classification of the binary classes by the classifier, which ranges between 0 and 1. Therefore, from the MCC metrics, all plant species except *ATH* displayed MCC between 0.9 and 1 with *ATH* having MCC of 0.875; whereas mammalian species displayed MCC of 0.82% and 0.8% for *HS* and *MM*, respectively.

Classification performance of 73 features was also measured using iRF which is shown in Table 3.3. On the other hand, iRF demonstrated similar statistics with marginal differences. Accuracy performance using iRF showed similar values as those obtained using RF having ACC ≥ 95% for *ZM*, *BNA*, *BRA*, *BOL*, *OS*, *SL* and *ST* whereas *ATH* displayed a slightly higher ACC of 94.20% using iRF. However, ACC obtained for *HS* and *MM* showed differences between the accuracies from RF by 0.12% and 0.16%, respectively. PRES performance displayed an overall increase of 1-1.5% as compared to RF. With RF, the SENS produced consistent values in multiple species whereas NPV showed slightly higher values for *BRA*, *BOL*, *SL* and *ST* using iRF.

Table 3.3: Classification performance of 73 features using iRF classifier.

Species	ACC	SENS	SPEC	PRES	PPV	NPV	F1	MCC
ATH	94.20	95.02	93.34	93.67	93.67	94.77	94.34	0.88
ZM	95.39	94.45	96.34	96.34	96.34	94.46	95.38	0.91
BNA	96.62	95.09	98.15	98.11	98.11	95.21	96.58	0.93
BRA	96.39	95.52	97.28	97.32	97.32	95.46	96.42	0.93
BOL	96.16	95.96	96.36	96.24	96.24	96.08	96.10	0.92
OS	96.82	96.72	96.92	96.80	96.80	96.85	96.76	0.93
SL	97.05	96.7	97.4	97.38	97.38	96.72	97.04	0.94
ST	96.07	95.47	96.67	96.63	96.63	95.52	96.05	0.92
HS	90.95	89.94	92.04	92.37	92.37	89.51	91.14	0.82
MM	89.97	86.44	93.71	93.56	93.56	86.73	89.86	0.80
6-plants	95.02	94.95	95.08	94.98	94.98	95.06	94.97	0.90
2-mammals	90.33	89.30	91.31	90.62	90.62	90.07	89.95	0.80

The F1-score also showed similar statistical measures showing slightly higher values for *ATH*, *BRA* and *HS* with increase of 0.56%, 0.31% and 0.07%, respectively. MCC values also provided similar statistics, with MCC ≥ 0.9 and < 1 for *ZM*, *BNA*, *BRA*, *BOL*, *OS*, *SL* and *ST* whereas *ATH*, *HS and MM* displayed MCC of 0.88, 0.82 and 0.8, respectively. SPEC on the other hand displayed slighter variations with an increase of 0.37% for *ATH*, decrease of 0.5% for *ZM*, increase of 0.15% for *BNA*, decrease of 0.23% for *BOL*, decrease of 0.24% for *OS*, decrease of 0. 4% for *SL*, decrease of 0.57% for *ST*, decrease of 0.33% for *HS*, and increase of 0.32% for *MM*.

Apart from classifying and measuring the performance on individual species, cross-species analysis has also been performed, in which a feature matrix of 30,000 sequences across 6 plant species (*ATH*, *BNA*, *BRA*, *BOL*, *ZM* and *OS*) and 10,000 sequences of 2 mammalian species (*HS* and *MM*) were used for classification analysis. 73 features extracted from these cross-species datasets were used for measuring the performance using RF and iRF classifiers. Classification performance of 6-plant species using RF (Table 3.2) shows overall ACC, SPEC, PRES, NPV and F1-score of ~95% and MCC of 0.9 whereas 2-mammalian species displayed ACC, PRES, NPV and F1-score of ~90% with slight variations in SENS of 89.63%, SPEC of 91.46% and MCC of 0.81.

On the other hand, iRF also generated similar ACC, SENS, NPV and MCC for plants and mammals (Table 3.3) with minute differences in SENS, SPEC, PRES and F1-score. Regarding computation time, classification analysis of cross-species took slightly longer than classification on individual species. This is mainly due to the number of sequences involved in the training step with generation of 400 forests.

Table 3.4: Classification performance of 73 features using SVM classifier.

Species	ACC	SENS	SPEC	PRES	PPV	NPV	F1	MCC
ATH	94.56	94.66	94.47	94.56	94.66	94.47	94.56	0.89
ZM	95.05	93.87	96.26	95.08	96.23	93.91	95.05	0.90
BNA	95.52	93.66	97.40	95.60	97.32	93.84	95.52	0.91
BRA	96.07	94.83	97.36	96.11	97.38	94.79	96.07	0.92
BOL	95.87	94.85	96.87	95.90	96.71	95.08	95.87	0.91
OS	96.90	97.54	96.29	96.91	96.20	97.60	96.90	0.93
SL	96.30	95.60	97	96.30	96.95	95.66	96.30	0.92
ST	95.60	94.80	96.40	95.61	96.34	94.88	95.60	0.91
HS	90.07	86.99	93.36	90.29	93.35	87.01	90.07	0.80
MM	89.51	85.39	93.86	89.86	93.64	85.87	89.50	0.79
6-plants	94.73	94.23	95.21	94.73	05.07	94.4	94.73	0.89
2-mammals	90.18	88.25	92	90.21	91.2	89.27	90.17	0.80

Classification of lncRNAs and mRNA sequences was also evaluated using a Support Vector Machine (SVM) classifier to benchmark the performance of 73 features and accuracy in determining lncRNAs using SVM. Results of the analysis has been shown in Table 5.4. SVM shows that performance using SVM does not vary significantly but does show slight variations in all the metric values. However, for some metrics, SVM does have lower values than RF and iRF. A comparison of ACC, PRES, SENS and F1 of three classifiers on multiple datasets (Figure 3.1) shows slightly lower performance of SVM for the majority of the datasets. When comparing to ACC (Figure 3.1a), a decrease in ACC using SVM is evident in *ZM*, *SL*, *ST*, *HS* and *MM* datasets whereas ACC obtained using RF and iRF does not vary much. When PRES is compared (Figure 3.1b), again a notable difference is observed for SVM, RF and iRF in *ZM*, *BNA*, *BRA*, *SL*, *ST*, *HS* and *MM* datasets where a decrease in PRES values is observed using SVM. It is also important to note that iRF produces the highest PRES values in these datasets amongst RF and SVM which can be observed in *BNA*, *BRA*, *HS* and *MM* datasets with significantly higher PRES values for iRF. SVM however only performs better in the *ATH* dataset, where it shows an increase of 0.57±0.21 for ACC and 0.83±0.06 for PRES. Results of SENS analysis also shows a decrease in SENS values for *ZM*, *BNA*, *BRA*, *BOL*, *SL*, *ST*, *HS* and *MM* datasets. iRF however, shows an improvement in *ATH*, *BRA* and *BOL* datasets. For the rest of the datasets, similar performance of iRF and RF can be observed, particularly in *SL*, *ST*, *HS*, *MM*, *6-plants* and *2-mammals* datasets. The comparison with respect to F1-score shows similar scores as observed in ACC metrics with a decrease in F1 values in *ZM*, *BNA*, *BRA*, *BOL*, *SL*, *ST*, *HS*, *MM*, *6-plants* and *2-mammals*.

The comparison of speed performance of RF, iRF and SVM shows that SVM performs comparatively much faster while training the transcript sequences.

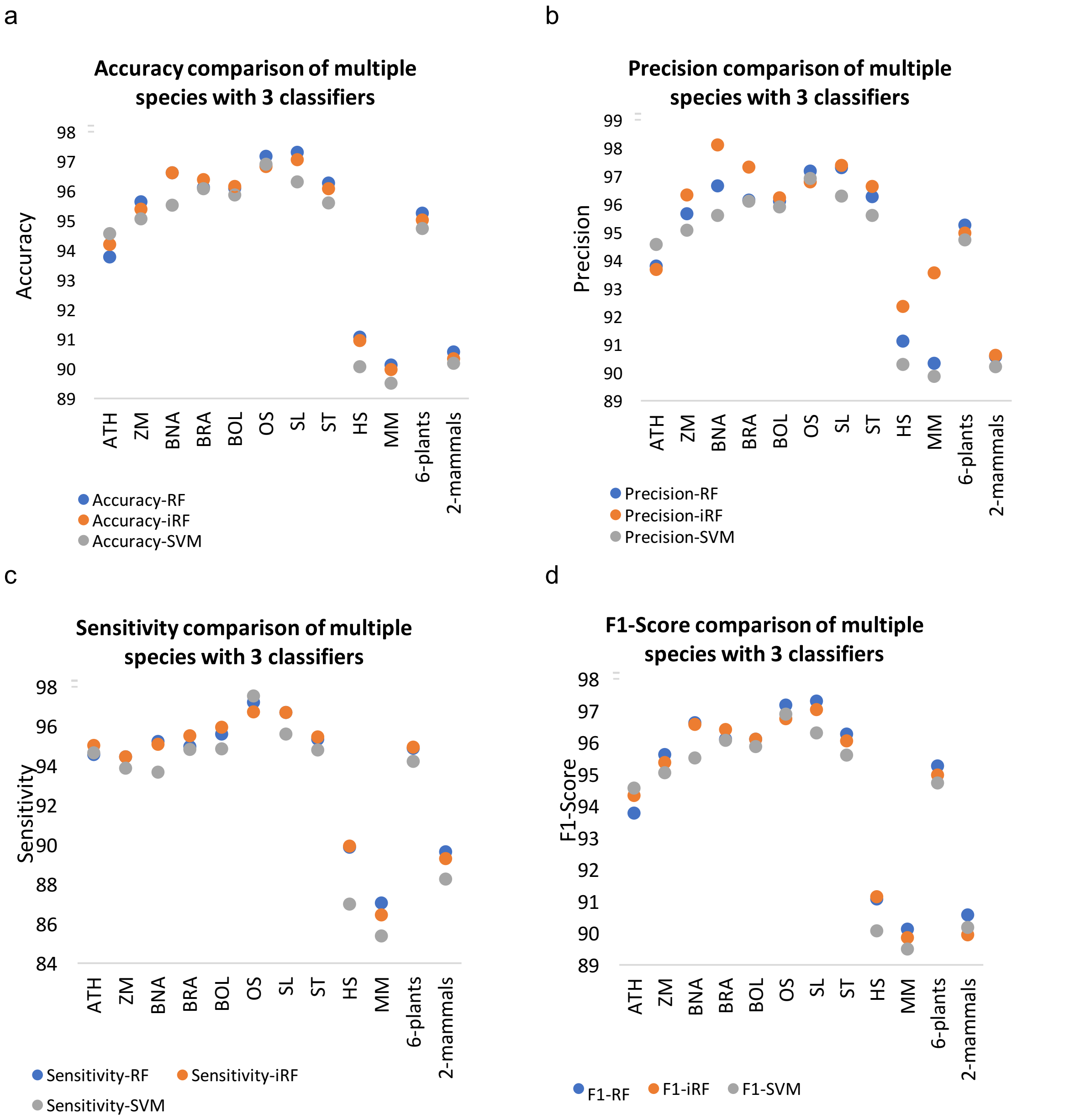


Figure 3.1: Classification performance comparison of RF, iRF and SVM classifiers with 73 features on plants and mammalian species. (a) Accuracy performance, (b) Precision performance, (c) Sensitivity performance and (d) F1-Score performance. Horizontal axis represents various species and vertical axis represents accuracy in percentage.

Comparison of iRF, RF and SVM was also performed by benchmarking their performance with respect to the total time (in seconds) in classifying lncRNAs and mRNA sequences from multiple species (Figure 3.2). Speed comparison results indicate that SVM performs the fastest classification, with less amount of time required for training than RF and iRF with iRF being comparatively slower. However, for *6-plants* datasets, SVM takes 144.26 seconds to train and predict the data, which is almost twice the time taken by iRF classifier. RF on the other hand, performs identical to SVM, with less time for training and prediction steps displaying a faster performance in *6-plants* dataset.

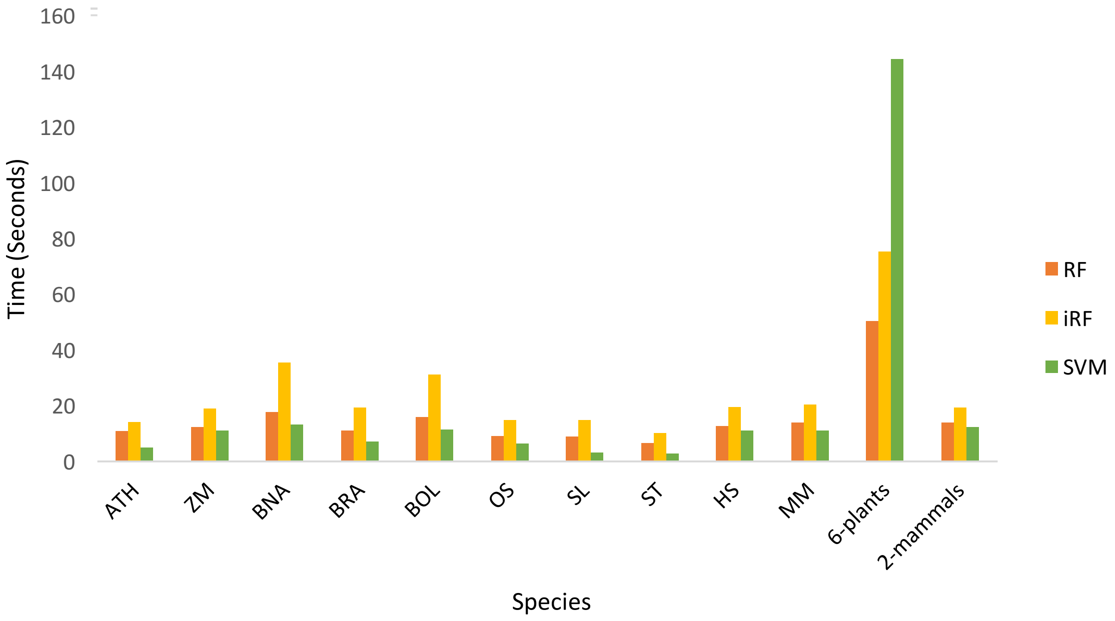


Figure 3.2: Speed comparison of RF, iRF and SVM classifiers in classification of lncRNAs and mRNA transcript sequences on multiple species datasets with horizontal axis representing various species and vertical axis time in seconds.

We further evaluated how the classification performance of multiple species using 73 features by plotting Receiver Operating Characteristics (ROC) curves. Figure 3.3a shows the ROC curves of 6-plant species, whereas Figure 3.3b shows ROC curves of 2-mammalian species where False Positive Fraction (FPF) is plotted against True Positive Fraction (TPF). The ROC curves in Figure 3.3a show that all the plant species exhibit similar performance, except for *ATH*, where the curve slightly dips downwards. The corresponding Area Under the Curve (AUC) score for the ROC curves of plant species (Table 3.5) illustrates an average AUC score of 99.23% by averaging the maximum AUC scores of 8 plant species. On the other hand, *HS* and *MM* datasets also display identical ROC curves except *MM* which displays lower TPF compared to *HS* having a higher TPF.

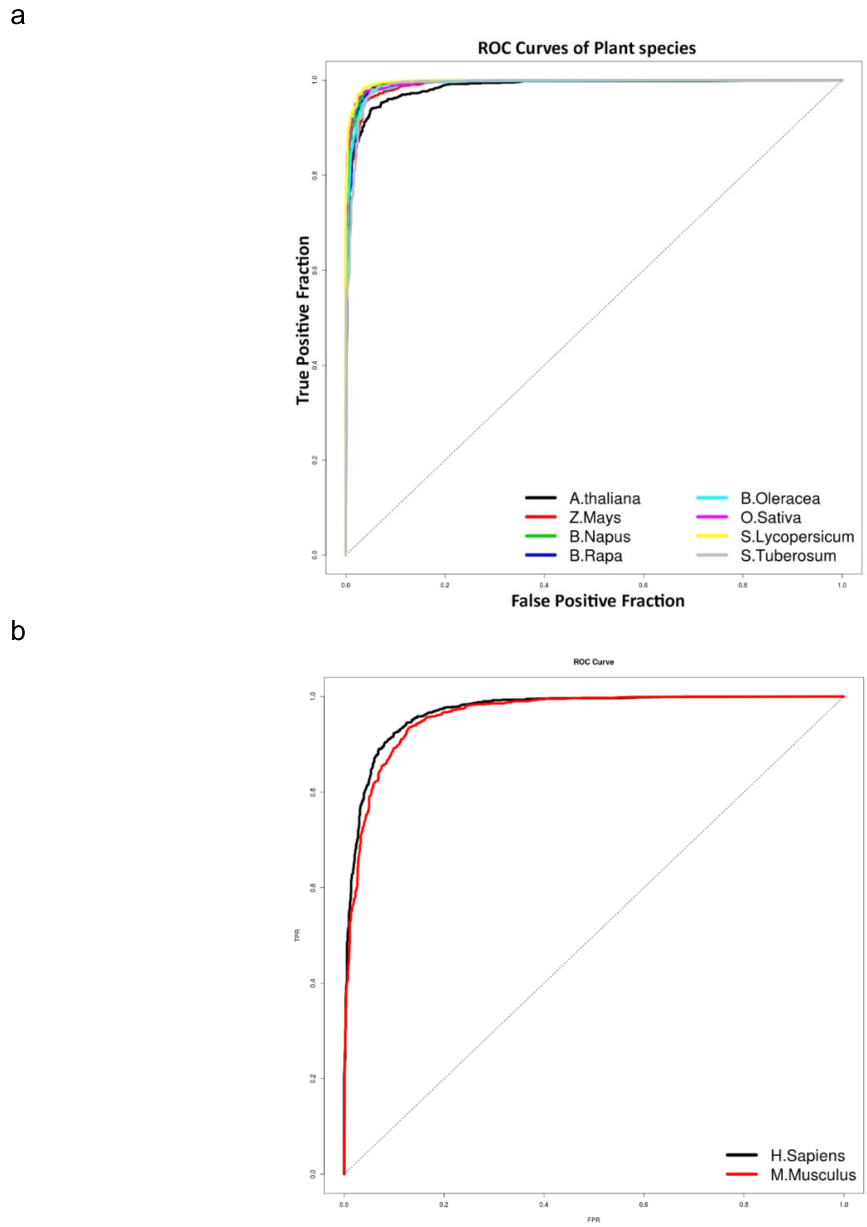


Figure 3.3: ROC curves showing performance comparison of 10 different species involving plants and mammalian datasets where TPR is the True Positive Rate and FPR is False Positive Rate. (a) shows ROC curves of 6 individual plants species and (b) shows ROC curved of 2 individual mammalian species.

HS and *MM* produces an average AUC score of 96.82% with maximum AUC scores over 4 iterations using iRF. For computation of accuracy values, 4 iterations were performed. iRF produced maximum AUC of 97.32% and 96.33% for HS and MM, respectively. AUC scores of 6-plants and 2-mammalian species were also computed. 6-plants cross species produced maximum AUC of 99.07% whereas 2-mammalian species produced an AUC of 96.12%.

Table 3.5: AUC scores of multiple species using iRF classifier with 4 iterations.

Species	**AUC score (Iteration 1)**	**AUC score (Iteration 2)**	**AUC score (Iteration 3)**	**AUC score (Iteration 4)**
ATH	98.58%	98.6%	98.58%	98.58%
ZM	98.9%	99.06%	99.02%	99.06%
BNA	99.46%	99.48%	99.49%	99.48%
BRA	99.2%	99.22%	99.18%	99.19%
BOL	99.22%	99.13%	99.11%	99.07%
OS	99.6%	99.58%	99.52%	99.53%
SL	99.67%	99.66%	99.67%	99.66%
ST	98.89%	98.91%	98.99%	98.91%
HS	97.32%	96.98%	96.96%	96.98%
MM	96.33%	96.24%	96.14%	96.21%
6-plants	99.07%	98.98%	98.95%	98.96%
2-mammals	96.12%	96.12%	96.07%	96.05%

Furthermore, a separate analysis was also conducted using the RF classifier to examine the classification of mRNA and lncRNA transcript sequences using 73 features by classifying into four separate classes: (1) mRNA plants, (2) mRNA mammals, (3) lncRNA plants and (4) lncRNA mammals. This analysis was primarily conducted to evaluate the performance of the features in distinguishing the transcript sequences into its appropriate class. Results of the classification are illustrated in Table 3.6 which shows PRES, SENS and F1-Score of the four classes. Results from multi-class classification analysis show that *mRNA_plants* class achieves PRES of 82 with SENS of 91 and F1 of 0.86 whereas *mRNA_mammals* achieves an average PRES and SENS of 90 and F1 of 0.90. Regarding lncRNA classes, *lncRNA_plants* achieves PRES of 89, SENS of 83 and F1 of 0.86 whereas *lncRNA_mammals* scores higher than plants with PRES of 92, SENS of 89 and F1 of 0.91. From the results, PRES or PPV of mRNA sequences in plants have slightly lower accuracy than other classes due to which the overall PRES decreases to 88 whereas *lncRNA_plants* show SENS of 83 mainly due to misclassification of lncRNA sequences in other classes.

Table 3.6: Classification metrics of lncRNA and mRNA transcript sequences in plants and mammalian species using a RF classifier.

Species	**Precision**	**Recall or Sensitivity**	**F1-Score**
mRNA_plants	82	91	0.86
mRNA_mammals	91	90	0.90
lncRNA_plants	89	83	0.86
lncRNA_mammals	92	89	0.91
Average	88	88	0.88

F1-score metrics which is a harmonic mean of both PRES and SENS shows that mRNA and lncRNA sequences in plants have comparatively lower values as compared to mammalian sequences which display F1-score of 0.9.

3.3.2 Determination of feature interactions

Prevalent feature interactions are the combinations of features which have a higher probability of occurrence and selection during the classification process as these combination of features produces better and improved accuracy. To achieve a prevalent list of feature interactors, an RIT algorithm has been employed, which provides a list of multi-order interactors. From RIT analysis, 2, 3, 4 and order-5 interactors were identified for plant and mammalian cross species datasets (Figure 3.4). The interactions were scored and assigned stability scores, which vary from 0.1 to 1.0. Feature interactions having stability scores ≥ 0.7 were extracted for further analysis. In this analysis, the search space was limited to obtaining order-5 interactions, since no new interactions were obtained with order > 5.

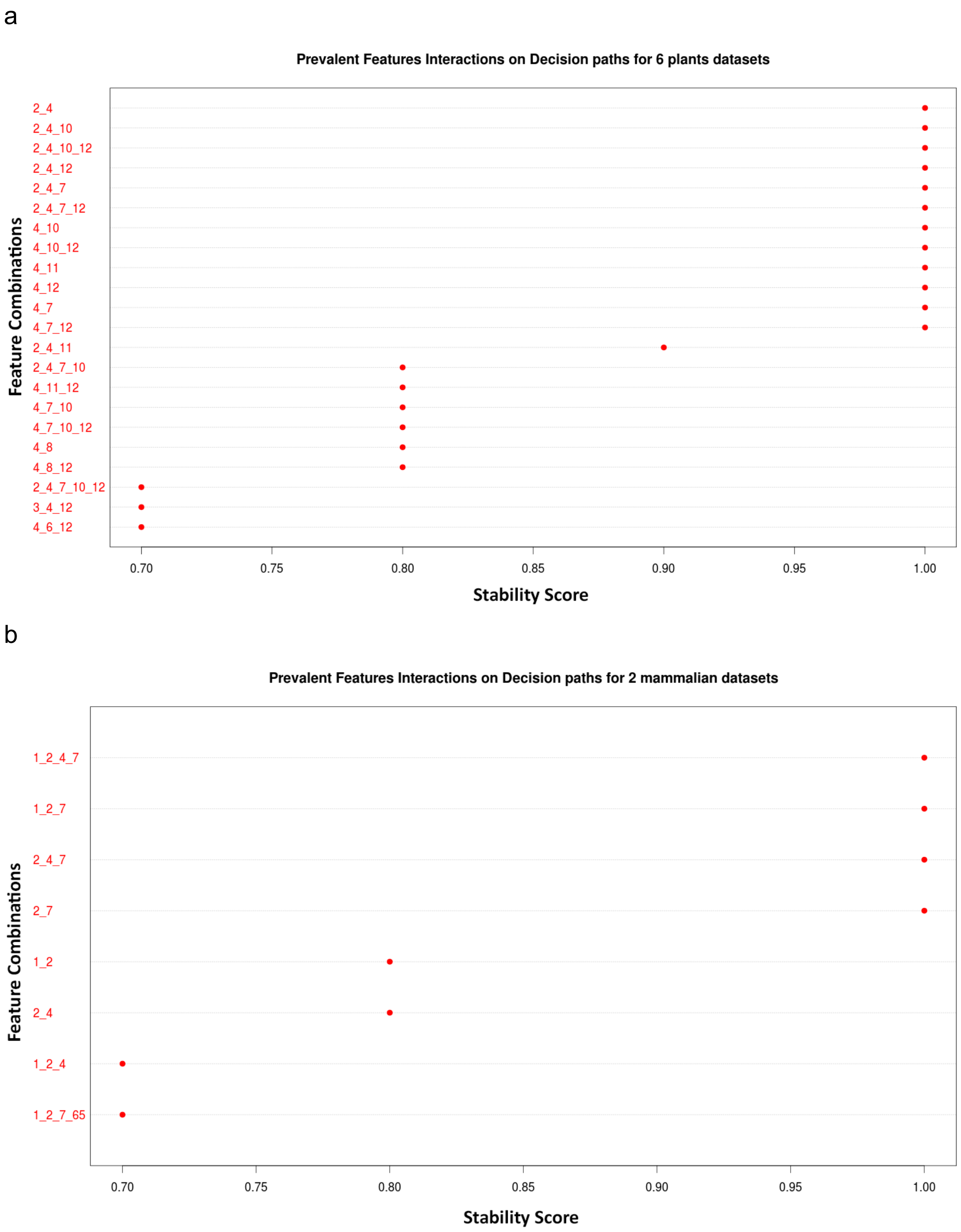


Figure 3.4: Dot chart showing prevalent feature interactions with list of highly prevalent feature combinations on vertical axis and stability scores on horizontal axis for (a) 6-plant species and (b) 2-mammalian species.

Results from RIT analysis (Table 3.7) shows only one order-5 interaction was obtained for 6-plant species with combination of *ORF length*, *ORF coverage*, *Fickett score*, *RCB* and *SCUO* features having stability score of 0.7. For mammalian species, no order-5 interaction was observed. For plants, four order-4 interactions, eleven order-3 interactions having stability scores of 1, and five order-2 interactions were produced with a score of 1.

In mammalian species (Table 3.7), order-4 interactions primarily displayed interactions between *hexamer score*, *ORF length*, *ORF coverage*, *Fickett score* and *CGG*^{RSCU} features having stability scores of 0.7 and 1. Whereas, order-3 interactions of these generated the score of 0.7 and 1. With order-3 interactions, three combinations of these features produced the score of 1 and remaining two combinations exhibited scores of 0.8. The *CGG*^{RSCU} feature was selected only in the order-4 interaction which shows that *CGG*^{RSCU} was preferentially selected with *Hexamer score*, *ORF length* and *ORF coverage* features.

Table 3.7: Prevalent feature interactions with stability scores ≥ 0.7 in plants and mammalian species obtained using the RIT algorithm.

Order	**6-Plants dataset**	**2-Mammals dataset**
Order-5	*ORFLength-ORFCoverage-FickettScore-RCB-SCUO*	None
Order-4	*ORFLength-ORFCoverage-FickettScore-RCB* *ORFCoverage-FickettScore-RCB-SCUO* *ORFLength-ORFCoverage-FickettScore-SCUO* *ORFLength-ORFCoverage-RCB-SCUO*	*HexamerScore-ORFLength-ORFCoverage-FickettScore* *HexamerScore-ORFLength-FickettScore-CGG*^{RSCU}
Order-3	*MeanORFCoverage-ORFCoverage-SCUO* *ORFCoverage-GC-SCUO* *ORFCoverage-Fop-SCUO* *ORFCoverage-FickettScore-RCB* *ORFCoverage-EW-SCUO* *ORFLength-ORFCoverage-EW* *ORFCoverage-FickettScore-SCUO* *ORFCoverage-RCB-SCUO* *ORFLength-ORFCoverage-FickettScore* *ORFLength-ORFCoverage-SCUO* *ORFLength-ORFCoverage-RCB*	*HexamerScore-ORFLength-FickettScore* *ORFLength-ORFCoverage-FickettScore* *HexamerScore-ORFLength-ORFCoverage*
Order-2	*ORFCoverage-Fop* *ORFCoverage-FickettScore* *ORFCoverage-EW* *ORFCoverage-RCB* *ORFLength-ORFCoverage*	*ORFLength-FickettScore* *HexamerScore-ORFLength* *ORFLength-ORFCoverage*

Apart from feature interactions having higher stability scores, other feature interactions with scores less than 0.7 were also obtained to study combination of features which are less stable. From this analysis, 52 feature combinations were obtained for 6-plant species (Figure 3.5a) having stability scores less than 0.7 out of which nine order-5 feature interactions were also obtained. In total, 26 feature interactions were observed for 6-plant species in which *hexamer score*, *ORF Length*, *Fickett score*, *ORF coverage*, *SCUO*, *EW*, *GC*, *Fop* were selected to form order-4 feature combinations producing scores of 0.5, 0.4, 0.3, 0.2 and 0.1. order-3 interactions, 11 combinations were produced with scores ranging between 0.1 and 0.6. For order-2 interactions, only four interactions were observed. For mammalian species (Figure 3.5b), four order-5 interactions were observed having stability scores less than 0.7. The order-4 feature combinations produced much lower scores between 0.1 and 0.4, whereas order-2 interactions displayed scores of 0.1.

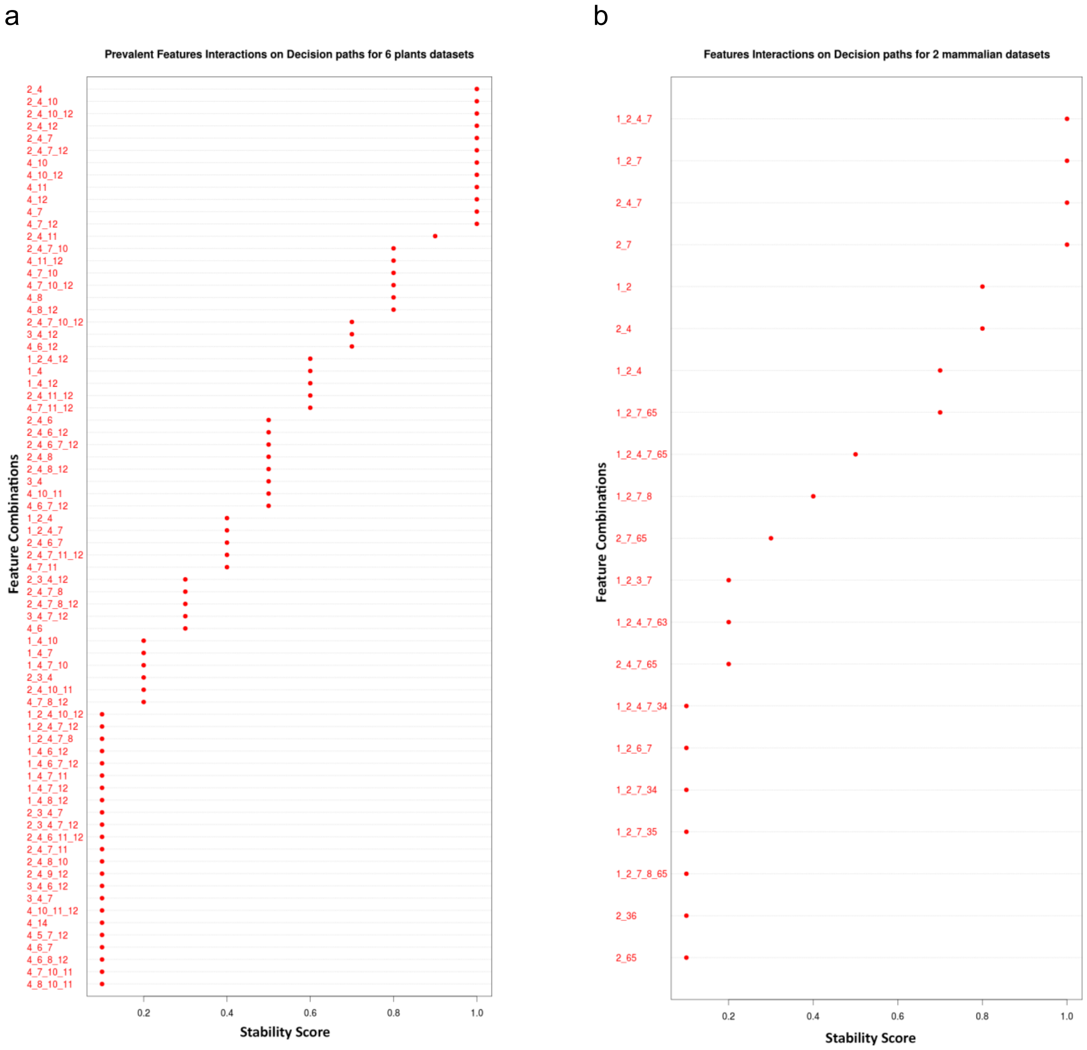


Figure 3.5: Dot chart showing prevalent feature interactions with list of all feature interactions for (a) 6-plant species and (b) 2-mammalian species.

Results from the RIT analysis indicate selection of certain features in plants and mammals. RIT generated higher number of feature combinations in plants as compared to mammals. 6-plant dataset generated 74 feature combinations whereas the 2-mammalian dataset generates 21 combinations. The combinations produced stability scores ≥ 0.7 in plants, whereas mammalian dataset generates 8 combinations with higher stability scores. As the search space is limited to order-5 combinations, plants species generate only one order-5 combination with the selection of *ORF length*, *ORF coverage*, *Fickett score*, *RCB* and *SCUO* features with stability score of 0.7. Contrastingly, mammalian dataset does not generate any order-5 feature combination with score ≥ 0.7. The rest of the order-5 combinations in plants and mammals display stability scores of 0.5, 0.3, 0.2 and 0.1. Regarding order-4 combinations, plants display

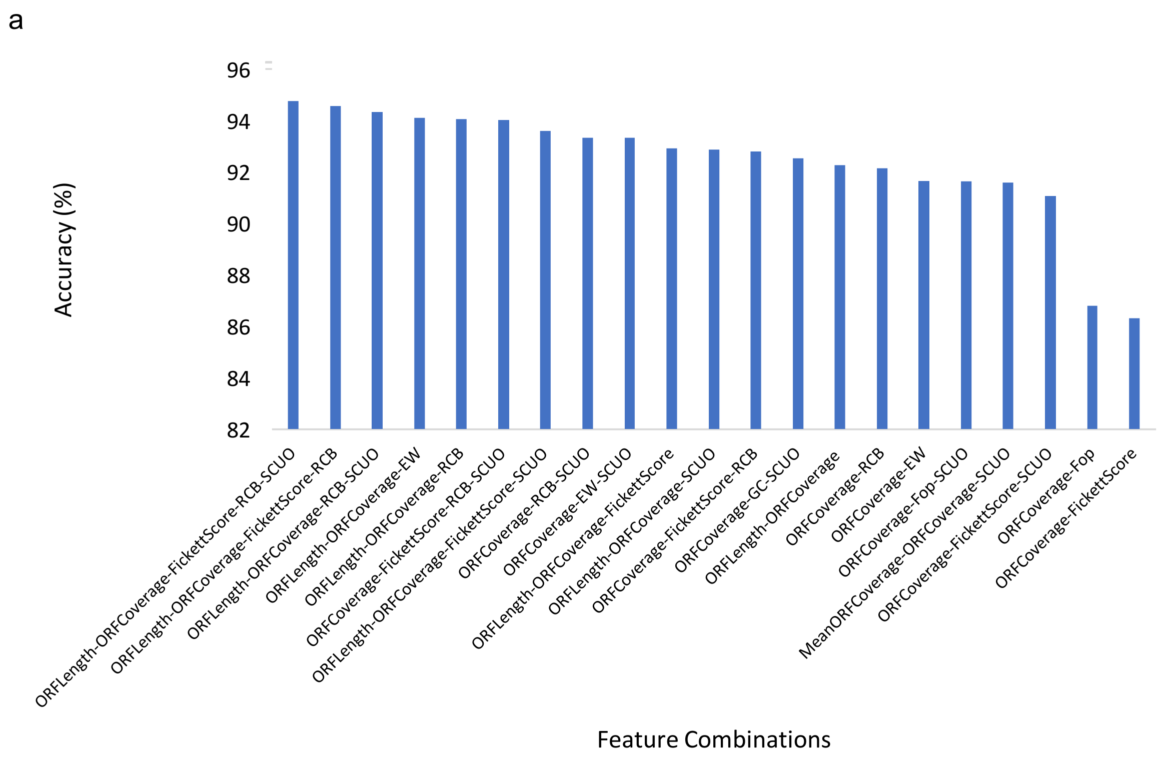
4 feature combinations with 50% having scores of 1 and the rest 50% having scores of 0.8. For mammals, two order-4 feature combinations display scores of 1 and 0.7. For order-3 combination of features, 8 out of 11 features have stability scores ≥ 0.8. Mammals however display two order-3 combinations with scores of 1 and one with score of 0.7. Order-2 feature combinations however contain the majority of combinations with scores of 1 in both the species.

A separate analysis was also undertaken to observe feature combinations with order > 5. However, no higher order feature combinations were generated using iRF.

3.3.3 Prediction performance of prevalent features

Based on the feature combination obtained in Table 3.5, performance of each combination was evaluated on 6-plants and 2-mammals dataset. The prediction accuracies were computed to evaluate their performance. Results of the performance evaluation can be observed from Figure 3.6 which shows a line chart of prediction accuracies in 6-plants and 2-mammals using the iRF classifier.

In 6-plants (Figure 3.6a), the 5 and 4-order feature combinations generated highest prediction accuracies ranging between 94.77% and 93.61%. The accuracy showed a decrease of $\sim 2\%$ in 3-order combinations. However, some feature combinations demonstrated an increase in the accuracy bringing the value to 94.1%. The increase in the accuracy was mainly observed in the combinations which contains *EW* and *SCOU* features. This confirms the importance of *EW* and *SCUO* as their selection improved the overall prediction accuracy in plants. Accuracy of 2-order combinations exhibited a sudden drop from $\sim 92\%$ to $\sim 86\%$. Apart from *EW* and *SCUO* features, *ORF length* also improved the accuracy significantly which confirmed its ability to distinguish the sequences.


a
96
94
92
90
88
86
84
82
Accuracy (%)
ORFLength-ORFCoverage-FickettScore-RCB-SCUO
ORFLength-ORFCoverage-FickettScore-RCB
ORFLength-ORFCoverage-RCB-SCUO
ORFLength-ORFCoverage-EW
ORFLength-ORFCoverage-RCB
ORFCoverage-FickettScore-RCB-SCUO
ORFLength-ORFCoverage-FickettScore-SCUO
ORFCoverage-RCB-SCUO
ORFCoverage-EW-SCUO
ORFLength-ORFCoverage-FickettScore
ORFLength-ORFCoverage-SCUO
ORFCoverage-FickettScore-RCB
ORFCoverage-GC-SCUO
ORFLength-ORFCoverage
ORFCoverage-RCB
ORFCoverage-EW
ORFCoverage-Fop-SCUO
MeanORFCoverage-ORFCoverage-SCUO
ORFCoverage-FickettScore-SCUO
ORFCoverage-Fop
ORFCoverage-FickettScore
Feature Combinations

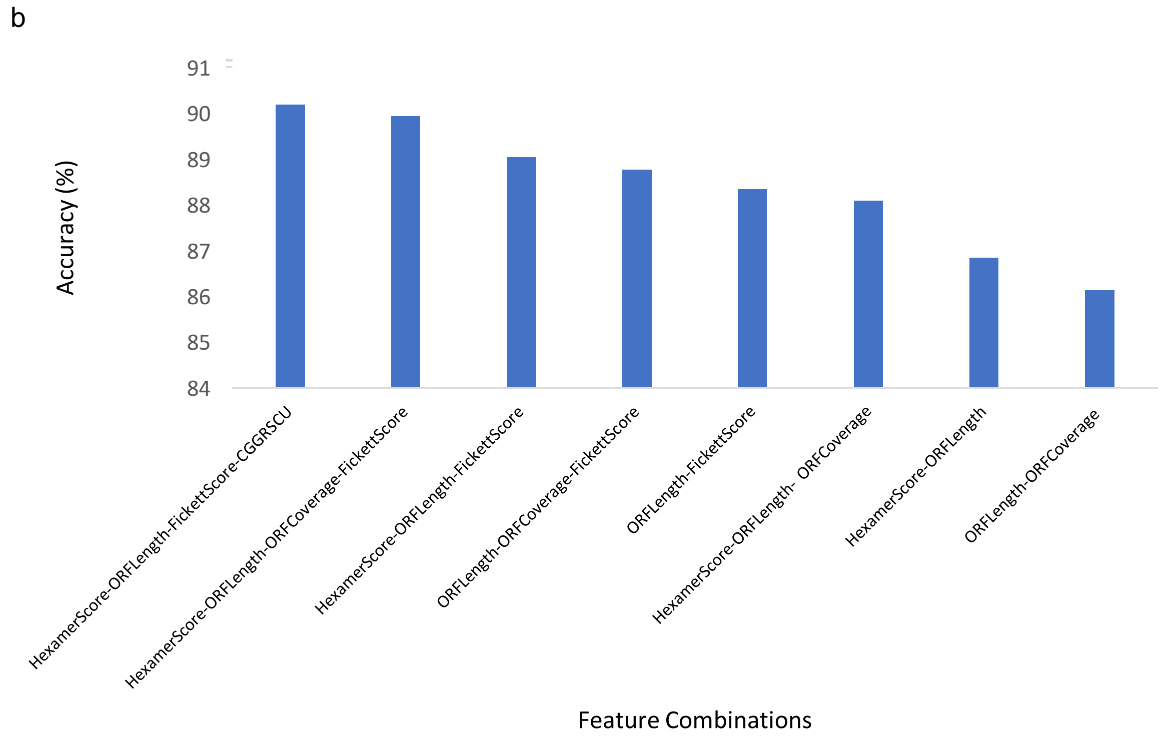


Figure 3.6: Performance of feature combinations based on prediction accuracy for (a) 6-plants, and, (b) 2-mammals datasets.

In contrast to plants, mammalian sequences exhibited dissimilar pattern of accuracies with feature combinations. The highest prediction accuracy of 90.18% was observed for the order-4 combination where *CGG*^{RSCU} was observed in combination with *hexamer score*, *ORF length* and *Fickett score*. The second lowest accuracy of 89.94% was observed with selection of *ORF coverage* instead of *CGG*^{RSCU}. The order-3 combinations generated lower accuracies than the order-4 combinations and steady decline in the accuracy was observed. Based on the non-selection of essential features in order-3 features and decrease in the accuracy, *ORF coverage* was ranked 1 followed by *hexamer score* with rank 2 and *Fickett score* with rank 3. From the order-2 features, the highest accuracy of 88.33% was produced by *ORF length* and *Fickett score*. The non-selection of *Fickett score* produced lower prediction accuracy values. This shows that in mammals, the most important features are the order-4 features: *hexamer score*, *ORF length*, *Fickett score* and *CGG*^{RSCU}.

3.3.4 Performance evaluation of individual features

Features extracted from the transcript sequences were also used for the evaluation of their individual performance in the classification of lncRNAs in 6-plants and 2-mammalian species transcript sequences. Figure 3.7 displays the performance of each feature classified using an iRF classifier. Results of the analysis in plants (Figure 3.7a) show that ORF length achieves the highest accuracy of 90.44%, followed by *ORFCoverage* with 82.32%. *Fickett score* achieves accuracy of 73.75% followed by GC with 69.57%. Performance of *hexamer score* becomes slightly lower when compared to other features in distinguishing lncRNAs in plants. Individual performance of codon bias features generates an average ACC of 61.55%. The bar plot below also illustrates that codon-bias features such as *CUB*, *EW* and *SCUO* individually.

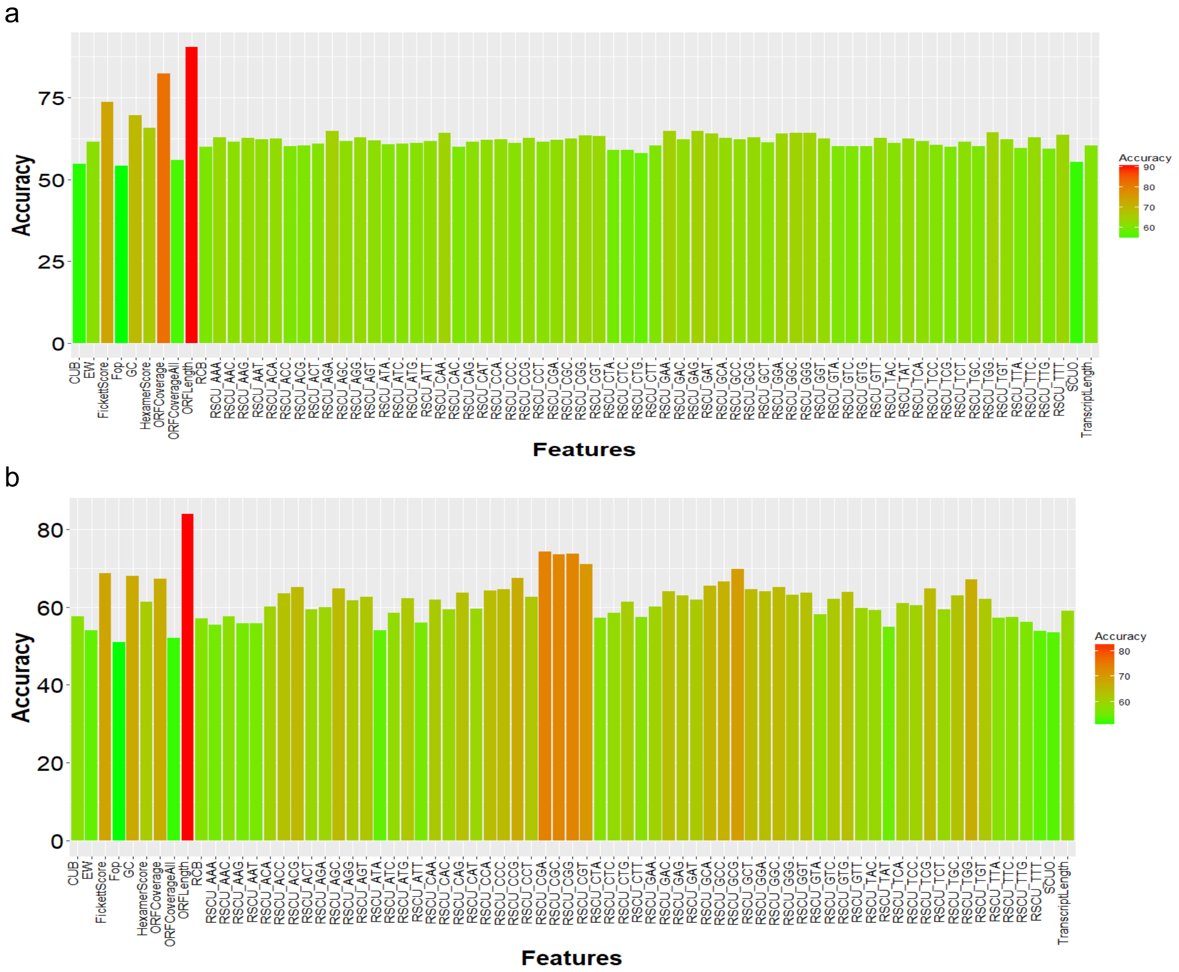


Figure 3.7: Bar charts displaying individual feature performance using iRF classifier in (a) 6-plant species transcript sequences and (b) 2-mammalian species transcript sequences.

Individual feature performance in mammalian sequences (Figure 3.7b) illustrates higher prediction accuracy of 83.85% by *ORF length* which is followed by *CGA*^{RSCU}, *CGG*^{RSCU}, *CGC*^{RSCU}, *CGT*^{RSCU}, *GCG*^{RSCU} with ACC of 74.27%, 73.68%, 73.48, 71.09% and 69.83% respectively. Sequence-based features such as *Fickett score*, *GC content* and *ORF coverage*, display slightly lower accuracy values of 68.73%, 67.95% and 67.23%, respectively. From the Figure 3.7b it can be clearly observed that some of the RSCU codon-bias features display higher accuracy in mammals when compared with plant data. This signifies that the RSCU features in mammals and ORF length in plants as well as mammals play significant roles in distinguishing the transcript sequences.

3.3.5 10-Fold Cross Validation performance on reference datasets

The performance of the iRF classifier with 73 features on 6-plants and 2-mammalian GENCODE datasets was evaluated using 10-Fold Cross-Validation (CV) feature sets. 10% of the features from the overall feature set were selected as a test set in each fold. The mammalian dataset consists of 10000 sequences; 9000 were selected as the training set and the remaining 1000 were selected as the test set. The plant dataset consists of 30000 protein-coding and lncRNA sequences, out of which 27000 were selected for the training set and 3000 as the test set. With unique test set values in each fold, AUC and accuracy values were computed (Figure 3.8).

Results from the 10-Fold CV show that the AUC scores for plants fluctuates between 98.7 and 99.4, whereas for mammals it fluctuates between 95.5 and 98. At fold 4 in mammals, the AUC score peaks to 97.77. This significant increase in the value suggests that the sequences chosen as test values at fold=4 show significant differentiation. In plants, the AUC metric reaches a peak value of 99.29 at fold 10. The same inference is generated suggesting the lncRNA and protein-coding sequences chosen as test set at the fold value of 10 can be easily distinguished with higher accuracy.

Comparison of accuracy values shows a similar pattern observed in the AUC scores. Accuracy values in plants show that the highest accuracy of 95.7% was obtained in fold 9 when compared to AUC value at fold=10. Since AUC is obtained by computing the fractions of true positive and negative rates, the tradeoff in the PPV and NPV values might affect the overall AUC, as well as the accuracy. On the other hand, accuracy values of mammalian dataset do not show significant differences with AUC scores. However, minor variations in folds=1, 2 and 8 can be observed which is attributed to the fractions of PPV and NPV values. The datasets display an average AUC of 99.04 for plants and 96.54 for mammals, whereas the average accuracy for plants is 95.37% and 90.99% for mammals.

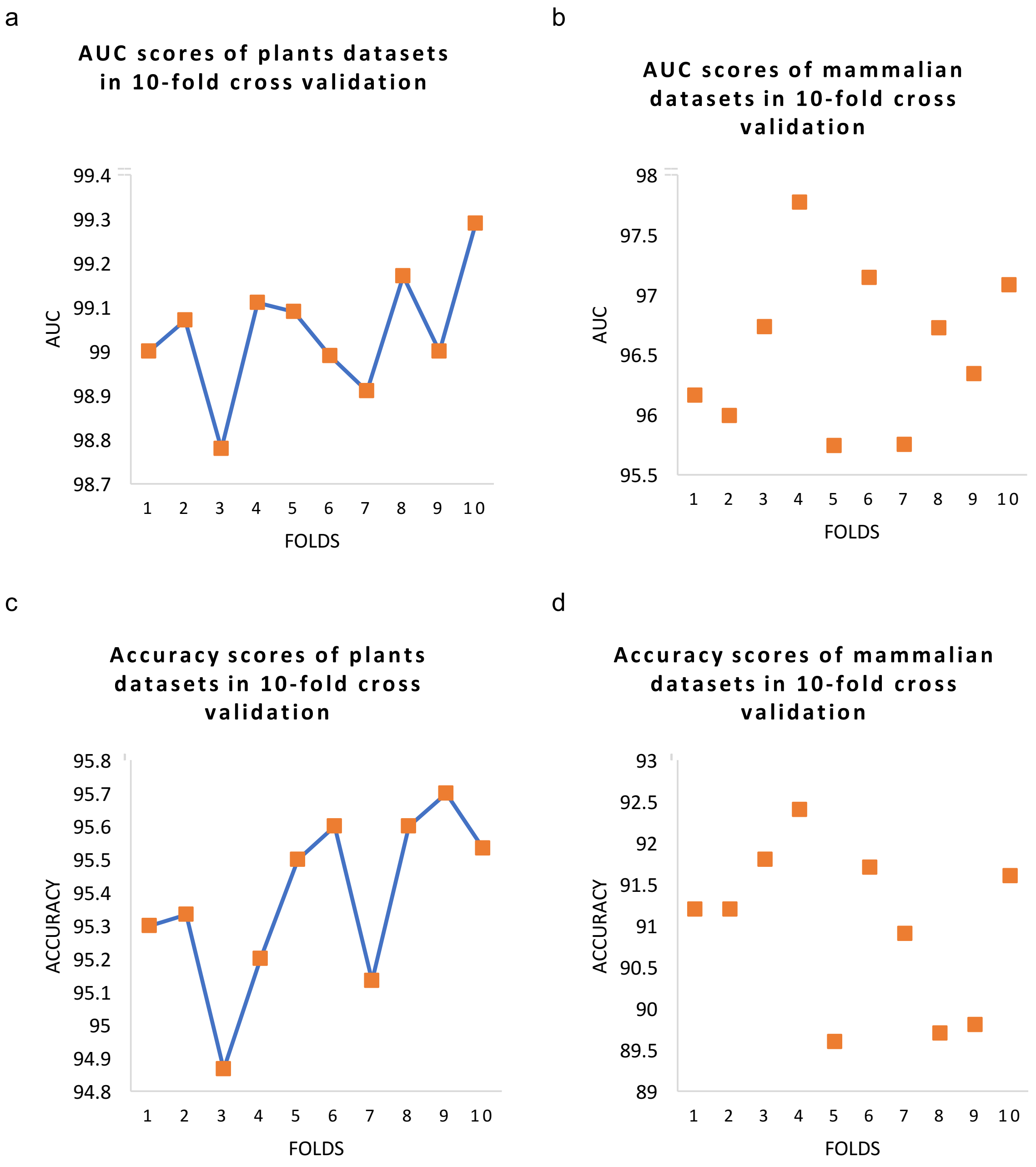


Figure 3.8: 10-Fold cross-validation performance of plants and mammalian GENCODE datasets using iRF classifier with (a) and (b) showing AUC scores, and (c) and (d) showing Accuracy scores.

3.4 Selection of optimal features using LiRF-FS on reference datasets

After performing classification using 73 features, the LiRF-FS algorithm was implemented to search for optimal features on the reference datasets. The feature selection method was applied on 6-plants and 2-mammals species to search for the feature set producing the highest accuracy in plants and mammals. Using the above approach for 6-plant species (Figure 3.9a), the lower λ value was kept to 10^{-6} and upper λ value was kept to 0.1. Using LASSO, the first feature was identified at $\lambda = 0.1$ which is *ORF coverage*. *ORF coverage* produced classification accuracy of 82.34% as a single feature. As λ further decreased from 0.1 to 0.032, two features were selected, namely, *ORF coverage* and *Fickett score*. With two features, the accuracy increased to 86.34%. A further decrease in λ value from 0.032 to 0.026 generated three features with additional selection of RCB feature, which increased the accuracy to 92.8%. With a decrease in λ from 0.026 to 0.0092, the *mean ORF coverage* gets selected which further increased the accuracy to 94.09%. With gradual decrease in λ value, the accuracy gets increased to the maximum value of 95.22% with the selection of 11 features, namely: *Hexamer score*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *GC content*, *Fop*, *RCB*, *EW*, *SCUO*, TAT^{RSCU}, GAT^{RSCU} at $\lambda = 3\times10^{-4}$. Further shrinking the λ decreases the accuracy to 94.45% with selection of 72 features at $\lambda = 10^{-6}$. However, the accuracy seems to fluctuate with mean accuracy of 94.83±0.385%. By applying the feature selection criteria mentioned in the methodology, the LiRF-FS method provides 7 optimal features: *hexamer score*, *mean ORF coverage*, *ORF coverage*, *Fickett score*, *Fop*, *RCB* and *SCUO* with accuracy of 94.91% at $\lambda = 6.9\times10^{-4}$.

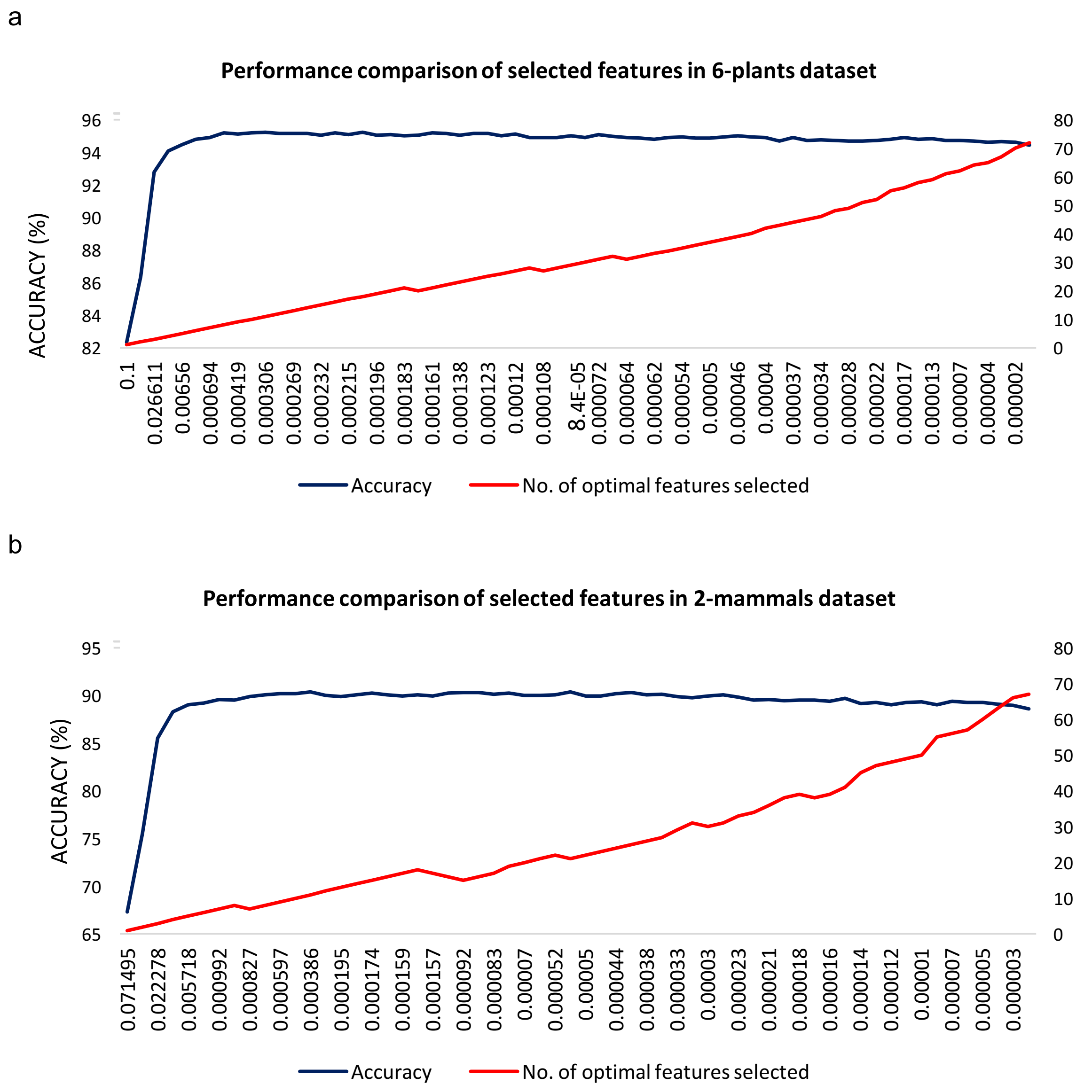


Figure 3.9: Line chart showing corresponding accuracy and optimal number of features selected with decrease in λ value using LiRF-FS method for (a) 6 plant species and (b) 2 mammalian species. Horizontal axis represents various λ values with accuracy in (%) representing primary vertical axis and number of features selected in secondary vertical axis.

In the case of the mammalian species, a similar pattern of selection was observed, as shown in Figure 3.9b. Unlike plants, the first feature (*ORF coverage*) gets selected at $\lambda = 0.071$ yielding an accuracy of 67.32%. Decrease in λ from 0.071 to 0.046 selects two features, *ORF coverage* and *Fickett score* which generate accuracy of 75.53%. By further decreasing λ to

0.0087, the accuracy drastically increases to 88.29% with selection of four features *ORF coverage*, *Fickett score*, *EW* and *SCUO*. At $\lambda = 0.0057$, *Mean ORF coverage* gets selected which increases the accuracy by 0.75%. The accuracy further increases and reaches peak value of 90.37% with the selection of 11 features, namely, *Hexamer score*, *ORF length*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *Fickett score*, *CUB*, *RCB*, *EW*, *SCUO*, *ACC^{RSCU}* at $\lambda = 3.8\times10^{-4}$. Further decrease in λ from 3.8×10^{-4} to 2.7×10^{-5} stabilizes the value at 90.06%, with the selection of 31 features after which the accuracy starts to decrease to 89.86% and starts to fluctuate downwards. At $\lambda = 10^{-6}$, 67 features are selected producing an accuracy of 88.61%. The trend of selection of features is quite like the one observed in plants species, with stability in $\lambda = 3\times10^{-4}$ to 1.2×10^{-4} with selection of 31 features producing an accuracy of 95.11%. From $\lambda = 2.3\times10^{-5}$, the accuracy decreases and remains constant till $\lambda = 10^{-6}$. This demonstrates that the optimal features should be selected based on the λ value where the prediction accuracy matches closely with λ value producing highest prediction accuracy.

Selection of features in plants and mammalian species can be explained by plotting the trace path of LASSO coefficients at various λ values. Figure 3.10 shows the result of applying LASSO on the training set with λ values on the x-axis. The x-axis is scaled so that maximal bound corresponding to OLS estimate is one. From the plots it can be clearly observed that most of the coefficient values of features range between 0 and 0.3 and are not selected when λ is higher. This LASSO behavior can be explained by the geometry underlying the ℓ_1 constraint which can be better understood by looking at the contour plot of LASSO and Ridge regression as shown in Figure 3.11. The contour plot illustrates coefficient values for feature x plotted using LASSO and Ridge regressions. Ridge regression employs the ℓ_2 norm which shrinks the values due to the circular nature of the constraint regions. LASSO on the other hand, employs the ℓ_1 norm creating squared constraint regions. The red ellipses represent the contours of the residual sum of squares (RSS) function of the OLS estimate and $\hat{\beta}$ represents the unconstrained least-squares estimate. The RSS has elliptical contours where the constraint region for Ridge is $|\beta|_1^2 + |\beta|_2^2 \leq t^2$ and the constraint region for LASSO is $|\beta_1| + |\beta_2| \leq t$. Due to the circular constraint region in Ridge, the coefficients for the features rarely reach zero. In other words, the probability of coefficients approaching zero is low, therefore the coefficient values of features shrink but do not reach zero. Contrastingly, due to the diamond shape of the contour, the RSS contour of the OLS or β coefficients of the features (β_j) touch either of the four corners of the diamond constraint, where the values become zero. Due to this nature, LASSO tends to yield sparse features having fewer non-zero coefficients whereas ridge does not.

When the trace path was constructed for the plants and mammalian species dataset, the majority of the features had zero coefficient values between 0 and 0.3 in plants (Figure 3.10a) and therefore these features do not contribute in improving the accuracy of the lncRNA identification. The trace path from 0.3 to 1 shows the selection of rest of the features when the $\lambda = 3\times10^{-4}$ to $\lambda = 2\times10^{-6}$. The λ is represented by $|\mathrm{coef}|/\max|\mathrm{coef}|$ value. Similarly, for mammalian species the trace path shows that the larger number of features produces zero coefficient values from 0 to 0.3 where the λ is optimal at 3.8×10^{-4}. A further decrease in λ does not alter the accuracy which is evident from the trace plot in Figure 3.10b, when the λ value increases from 0.3 to 1. With an increase in the number of features having non-zero coefficient values leading to decrease in sparsity, the accuracy does not improve significantly when the λ changes from 4×10^{-4} to 3×10^{-6}. The selection of sparse features can be clearly observed in Figure 3.12 which shows the trace path of λ values from 0 to 0.4 for plants and 0 to 0.16 in mammals. Comparison of the accuracies of the selected features shows that features such as *ORF coverage* and *Fickett score* produces non-zero coefficient values when the λ value is lower. However, optimal features can be obtained when the λ value is 0.3 (Figure 3.12a) with an accuracy of 95.22% and the selection of 11 features in plants. When the value approaches 0.4, the prediction accuracy remains constant at 95.11% with selection of 27 features. In mammals, the prediction accuracy reaches the maximum at 0.07 λ value with ACC of 90.37%. Similar to the trace path in plants, a further increase in λ value from 0.07 to 0.16 does not alter the accuracy and keeps it constant with accuracy of 90.06%. An increase in λ value to 1 produces no improvement in the accuracy with further selection of features.

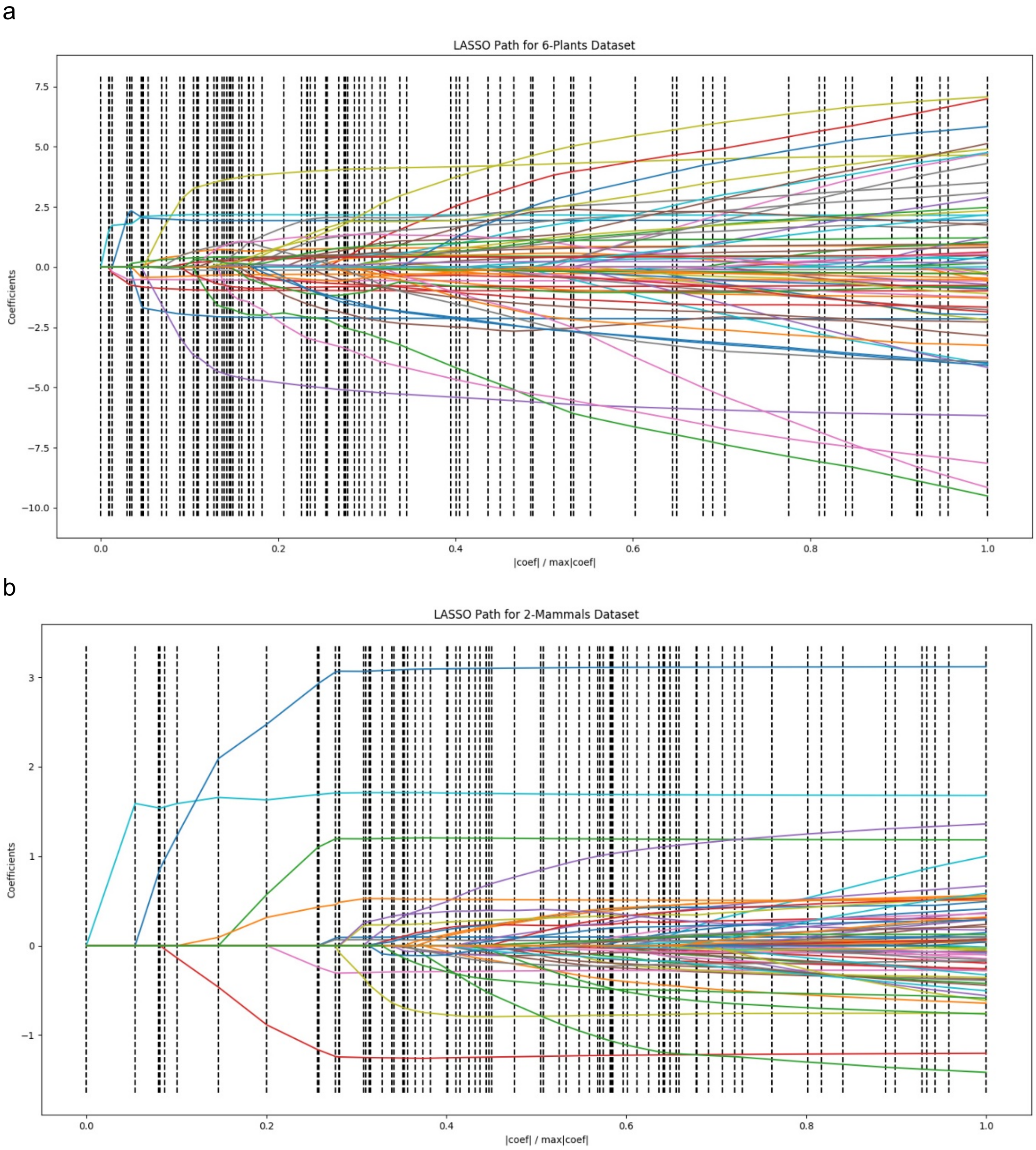


Figure 3.10: LASSO trace path of the coefficients against the ℓ_1-norm of the coefficient vectors as λ varies from 0 to 1. (a) and (b) shows the complete LASSO trace path for all values of λ for 6-plants and 2-mammalian datasets, respectively with $|\text{coef}|/\max|\text{coef}|$ (i.e. λ) values on horizontal axis and β coefficient values for selected features on vertical axis.

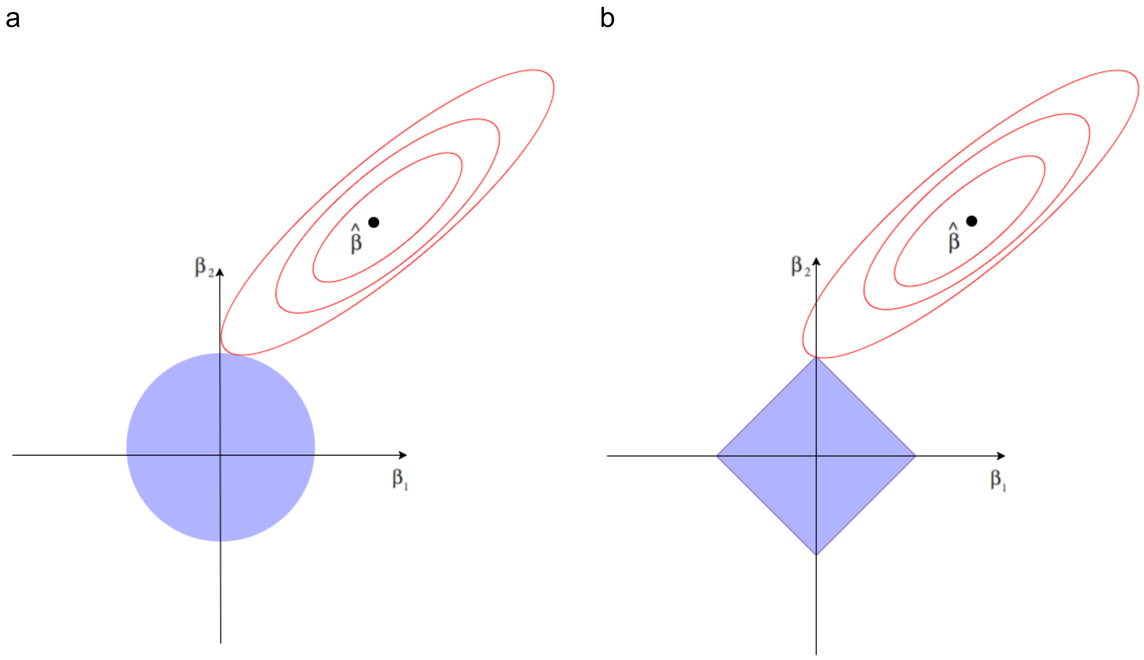


Figure 3.11: An estimation picture for (a) Ridge regression and (b) LASSO regression showing the contours of error and constraint functions. Solid blue regions are the constraint regions $|\beta|_1^2 + |\beta|_2^2 \leq t^2$ and $|\beta_1| + |\beta_2| \leq t$, respectively.

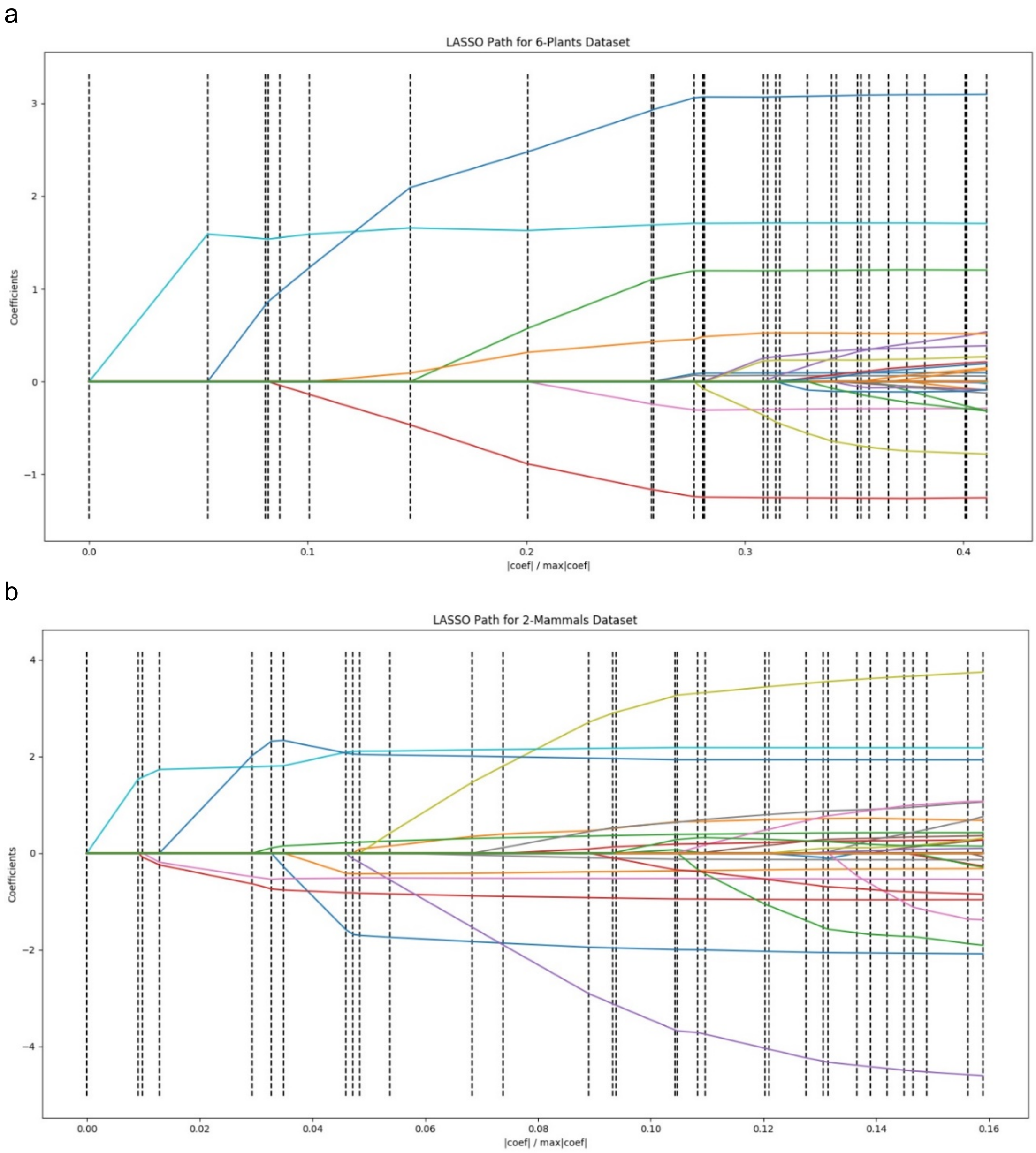


Figure 3.12: LASSO trace path of the coefficients against the l_1-norm of the coefficient vectors showing selection of features with increase in accuracy over the selected λ values. (a) and (b) shows the LASSO trace path for selected λ values for 6-plants and 2-mammalian datasets, respectively when the accuracy increases, with λ values ranging from 0.1 to 1.2×10^{-4} for plants and 0.071 to 3.5×10^{-5} for mammals on the horizontal axis and β coefficient values for selected features on vertical axis.

Feature selection using *LiRF-FS* methodology was applied on 6-plants and 2-mammalian species. By varying the λ from 1 to 10^{-5} with step size of 10^{-5} and tolerance value of 0.5, 7 optimal features are generated, namely, *hexamer score*, *mean ORF coverage*, *ORF coverage*, *Fickett score*, *Fop*, *RCB* and *SCUO* at $\lambda = 6.9 \times 10^{-3}$ with prediction accuracy of 94.95%. Since the tolerance value provided was 0.5, the difference from the maximum accuracy was 0.33%. The prediction accuracy was highest at $\lambda = 3.9 \times 10^{-3}$ with the selection of 10 features, namely: *hexamer score*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *Fickett score*, *Fop*, *RCB*, *Ew*, *SCUO* and TAT^{RSCU} producing accuracy of 95.28%. For mammals, the optimal features were selected at $\lambda = 7.2 \times 10^{-3}$ producing an accuracy of 90.10% with the selection of 8 features, namely: *hexamer score*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *Fickett score*, *RCB*, *Ew*, *SCUO*. The accuracy at this λ value differs from the maximum accuracy value of 90.37% with a difference of 0.27%. When the accuracy reaches its maximum at $\lambda = 3.8 \times 10^{-3}$, 11 features are selected, namely: *hexamer score*, *ORF length*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *Fickett score*, *CUB*, *RCB*, *EW*, *SCUO*, ACC^{RSCU} and CGA^{RSCU}.

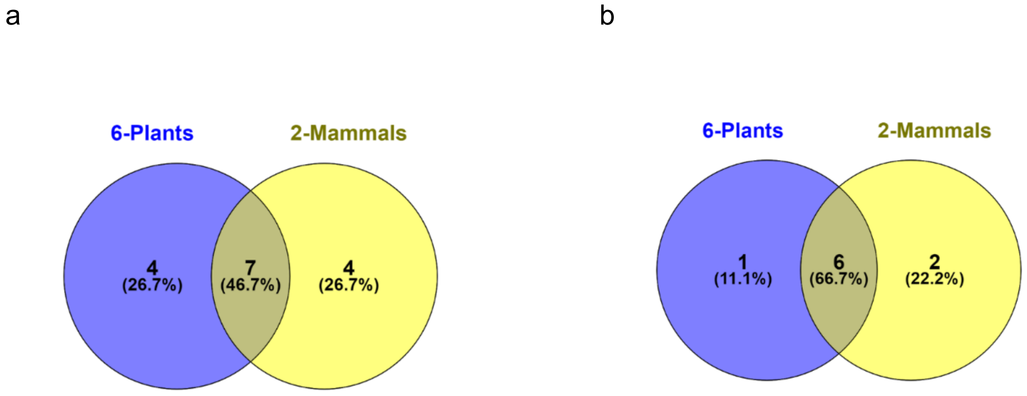


Figure 3.13: Venn diagram showing the number of common and exclusive features from 6-plants and 2-mammals datasets with (a) features producing the maximum prediction accuracy, and (b) optimal features obtained from LiRF-FS.

Resulting features obtained from plants and mammalian datasets using LiRF-FS method were compared to obtain the features which are commonly selected in both the datasets (Figure 3.13). Two comparisons were made, namely: (1) features producing maximum prediction accuracy among all the features in both the species (Figure 3.13a), and (2) features obtained from LiRF-FS optimisation producing accuracy within the given threshold value (Figure 3.13b). Results from the intersection of features show an overlap of 7 features, namely, *hexamer score*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *RCB*, *EW* and *SCUO* which are commonly selected in both species, whereas 4 and 2 features were exclusively selected in plants and mammals, respectively. *GC content*, *Fop*, *TAT*^{RSCU}, *GAT*^{RSCU} were exclusively selected among plants and *ORF length, Fickett score*, *CUB* and *ACC*^{RSCU} were exclusively selected among mammals. Another analysis of results obtained from the LiRF-FS optimisation method shows an overlap of 6 features, namely, *hexamer score*, *mean ORF coverage*, *ORF coverage*, *Fickett score*, *RCB* and *SCUO*. Among the exclusive features, *Fop* was selectively found among plants whereas *transcript length* and *EW* were selected among mammalian datasets.

To study the feature importance obtained from the LiRF-FS on plants and mammals, frequencies of individual features were computed over the iterations of λ values selected in plants and mammals. Results of the feature importance in plants and mammals has been plotted in Figure 3.14. The results show the frequency of selection of individual features across the overall values of λ. Results demonstrate that in both the species, the sequence and ORF-based features are highly selected as compared to codon-biased features. However, some of the codon-biased features do display selection in selected λ values.

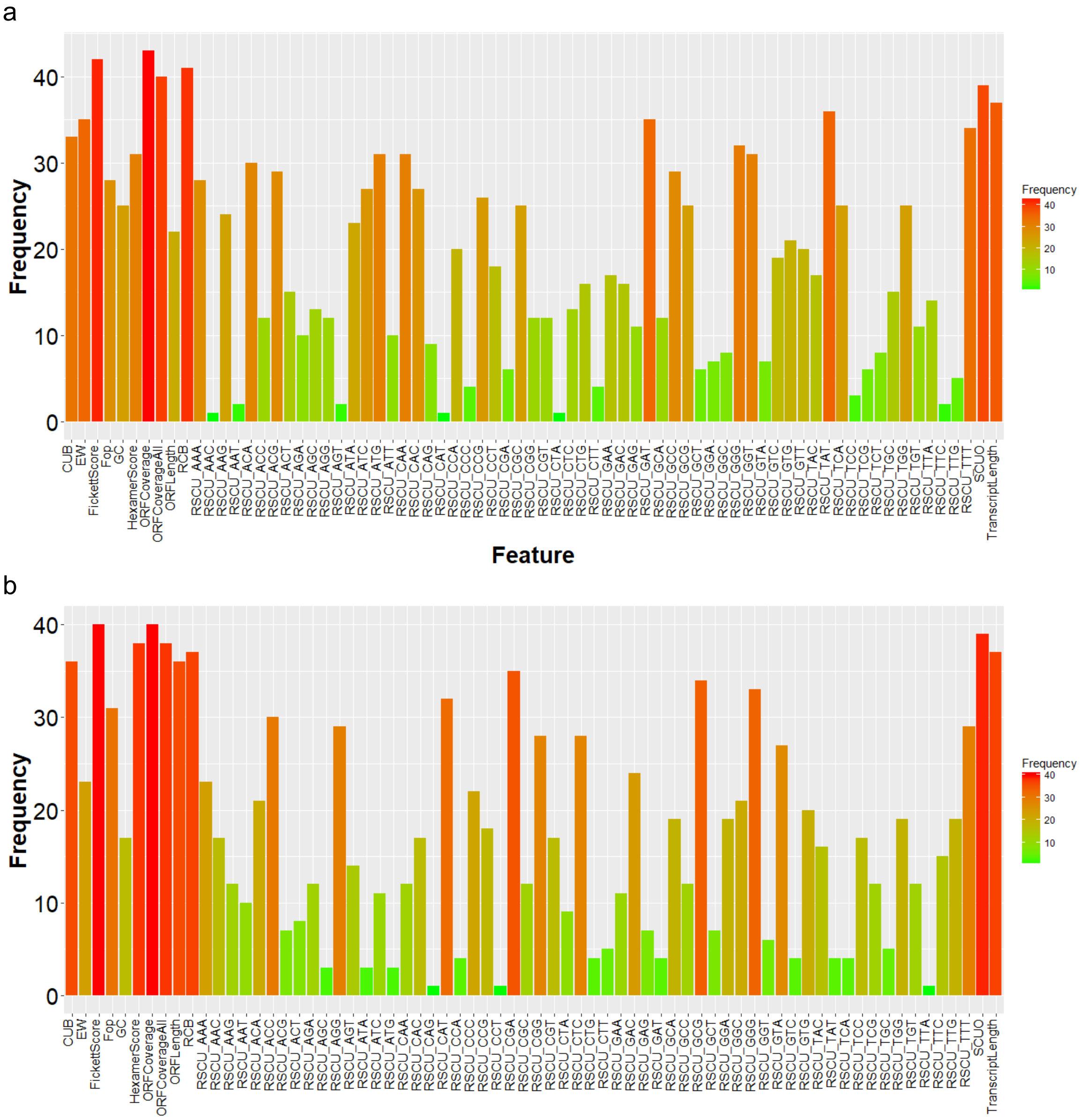


Figure 3.14: Frequency of selection of individual features across various λ values in (a) 6-plants, and (b) 2-mammalian GENCODE datasets.

Results from the datasets can be divided into three clusters:

1. Cluster-1: Features displaying highest frequencies (Frequency > 30)
2. Cluster-2: Features displaying moderate frequencies (Frequency < 30 and ≥ 15)
3. Cluster-3: Features displaying lower frequencies (Frequency < 15)

Table 3.8: Distribution of features based on individual performance in plants and mammals.

Species	Cluster-1	Cluster-2	Cluster-3
Plants	*ORF coverage*, *mean ORF coverage*, *Fickett score*, *RCB*, *SCUO*, *EW*, *CUB*, *transcript length*, *hexamer score*, TTT^{RSCU}, ATG^{RSCU}, ACA^{RSCU}, TAT^{RSCU}, CAA^{RSCU}, GAT^{RSCU}, GGT^{RSCU}, GGG^{RSCU}	*ORF length*, *GC content*, *Fop*, CTG^{RSCU}, ATC^{RSCU}, ATA^{RSCU}, GTT^{RSCU}, GTC^{RSCU}, GTG^{RSCU}, TCA^{RSCU}, CCT^{RSCU}, CCA^{RSCU}, CCG^{RSCU}, ACT^{RSCU}, ACG^{RSCU}, GCC^{RSCU}, GCG^{RSCU}, TAC^{RSCU}, CAC^{RSCU}, AAA^{RSCU}, AAG^{RSCU}, GAC^{RSCU}, GAA^{RSCU}, TGC^{RSCU}, TGG^{RSCU}, CGC^{RSCU}	TTC^{RSCU}, TTA^{RSCU}, TTG^{RSCU}, CTT^{RSCU}, CTC^{RSCU}, CTA^{RSCU}, ATT^{RSCU}, GTA^{RSCU}, TCT^{RSCU}, TCC^{RSCU}, TCG^{RSCU}, CCC^{RSCU}, ACC^{RSCU}, GCT^{RSCU}, GCA^{RSCU}, CAT^{RSCU}, CAG^{RSCU}, AAT^{RSCU}, AAC^{RSCU}, GAG^{RSCU}, TGT^{RSCU}, CGT^{RSCU}, CGA^{RSCU}, CGG^{RSCU}, AGT^{RSCU}, AGC^{RSCU}, AGA^{RSCU}, AGG^{RSCU}, GGC^{RSCU}, GGA^{RSCU}
Mammals	*Hexamer score*, *ORF length*, *mean ORF coverage*, *ORF coverage*, *transcript length*, *Fickett score*, *Fop*, *CUB*, *RCB*, *SCUO*, ACC^{RSCU}, GCG^{RSCU}, CAT^{RSCU}, CGA^{RSCU}, GGG^{RSCU}	*GC content*, *EW*, TTT^{RSCU}, TTC^{RSCU}, TTG^{RSCU}, CTC^{RSCU}, GTA^{RSCU}, GTG^{RSCU}, TCC^{RSCU}, CCC^{RSCU}, CCG^{RSCU}, ACA^{RSCU}, GCA^{RSCU}, TAC^{RSCU}, CAC^{RSCU}, AAC^{RSCU}, AAA^{RSCU}, GAC^{RSCU}, TGG^{RSCU}, CGT^{RSCU}, CGG^{RSCU}, AGG^{RSCU}, GGC^{RSCU}, GGA^{RSCU}	TTA^{RSCU}, CTT^{RSCU}, CTA^{RSCU}, CTG^{RSCU}, ATC^{RSCU}, ATA^{RSCU}, ATG^{RSCU}, GTC^{RSCU}, TCA^{RSCU}, TCG^{RSCU}, CCT^{RSCU}, CCA^{RSCU}, ACT^{RSCU}, ACG^{RSCU}, GCT^{RSCU}, GCC^{RSCU}, TAT^{RSCU}, CAA^{RSCU}, CAG^{RSCU}, AAT^{RSCU}, AAG^{RSCU}, GAT^{RSCU}, GAA^{RSCU}, GAG^{RSCU}, TGT^{RSCU}, TGC^{RSCU}, CGC^{RSCU}, AGT^{RSCU}, AGC^{RSCU}, AGA^{RSCU}, GGT^{RSCU}

Results from plants datasets (Figure 3.14a) suggests that 17 features with frequencies above 30 show higher selection (Table 3.8). 8 RSCU features were selected in this category. Second cluster consists of 2 sequence-based and 24 codon-bias features. Whereas the third cluster included 30 RSCU codon-biased features. From the mammalian datasets (Figure 3.14b), 15 features were selected in the first cluster displaying higher frequencies (Table 3.8). These included 6 sequence-based 9 codon-bias features. The second cluster consisted of only 1 sequence-based and 23 codon-bias features. The third cluster was comprised of 31 RSCU features.

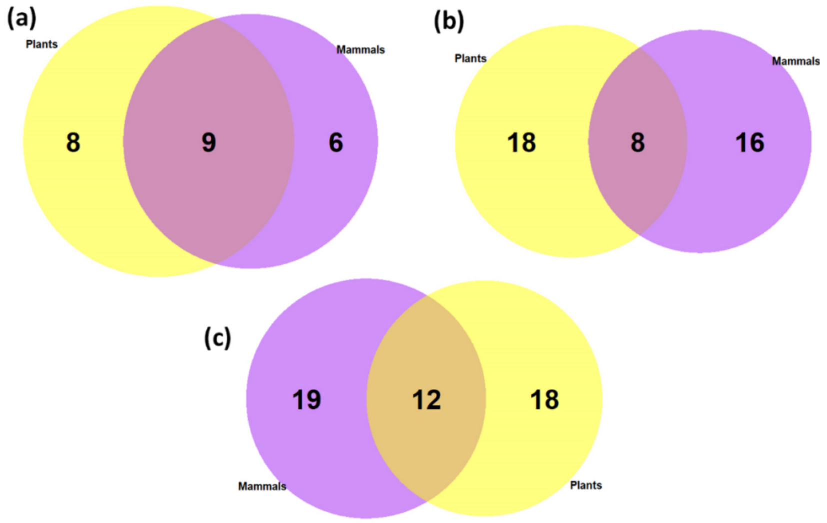


Figure 3.15: Venn diagram of the unique and shared features between plants and mammals in (a) Cluster-1, (b) Cluster-2 and (c) Cluster-3.

Results from the intersection of the features from the Venn diagram (Figure 3.15a) show that in Cluster-1, 9 features were found to be overlapping in both plants and mammalian datasets along with 8 and 6 species-specific non-overlapping features. Whereas in Cluster-2 (Figure 3.15b), only 8 features had overlaps. A large proportion was found to have selectively moderate frequencies as non-overlapping features. Cluster-3, however, displayed a large fraction of features as shared feature sets from both species (Figure 3.15c). From the overlaps obtained between the three clusters, approximately 50% of the features were selected with high, moderate and lower frequencies.

3.5 Comparison of different feature selection methods

Comparison of prediction performance of the LiRF-FS method was performed with five other feature selection methods, namely, mRMR (Peng et al., 2005b), Chi-square (Chen and Chen, 2011), Information Gain (IG) (Lee and Lee, 2006a), ReliefF (Durgabai, 2014), and UDFS (Yang *et al.*, 2011). Comparison was performed on the 6-plants and 2-mammalian species datasets. Classification of lncRNA and mRNA was performed using the RF classifier. The feature selection was performed on the training set. Using a threshold value of 10, only the top 10 features were selected from the ranked list of features (Table 3.9). Using the above criteria, IG selected 7 features *ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *mean ORF coverage*, *SCUO* and *Fop* in plants (Table 3.9) with prediction accuracy of 94.88%. Whereas for mammals, it produced *ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *transcript length*, *EW*, *SCUO*, *mean ORF coverage* with accuracy of 89.67%. Chi-square on the other hand selected *ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *EW*, *ORF length*, *TAT*^{RSCU}, *CCA*^{RSCU}, *CAC*^{RSCU} and *GC content* producing accuracy of 95.37%. For mammals, it produced *ORF coverage*, *Fickett score*, *hexamer score*, *SCUO*, *EW*, *GC content*, *RCB*, *transcript length*, *ORF length* and *CUB* with accuracy of 89.9%.

Table 3.9: Comparison of features selected using different feature selection methods.

Method	Features selected in Plants	Features selected in Mammals
mRMR	*hexamer score*, *ORF coverage*, *Fickett score*, *Fop*, *ATG*^{RSCU}, *GCT*^{RSCU}, *TAT*^{RSCU}, *CAG*^{RSCU}, *TGT*^{RSCU} and *CGG*^{RSCU}	*mean ORF coverage*, *GC content*, *Fickett score*, *CUB*, *CCA*^{RSCU}, *CAG*^{RSCU}, *GAT*^{RSCU}, *GAA*^{RSCU}, *TGC*^{RSCU} and *CGT*^{RSCU}
Chi-square	*ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *EW*, *ORF length*, *TAT*^{RSCU}, *CCA*^{RSCU}, *CAC*^{RSCU} and *GC content*	*ORF coverage*, *Fickett score*, *hexamer score*, *SCUO*, *EW*, *GC content*, *RCB*, *transcript length*, *ORF length* and *CUB*
Information Gain	*ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *mean ORF coverage*, *SCUO* and *Fop*	*ORF coverage*, *Fickett score*, *hexamer score*, *RCB*, *transcript length*, *EW*, *SCUO* and *mean ORF coverage*
ReliefF	*Fickett score*, *ORF coverage*, *SCUO*, *Fop*, *RCB*, *hexamer score*, *mean ORF* coverage, *GC content*, *CUB* and *transcript Length*	*mean ORF coverage*, EW, RCB, *ORF coverage*, Fop, *hexamer score*, *ORF length*, *transcript length*, *TCA*^{RSCU} and *TGC*^{RSCU}
UDFS	*ORFLength, TranscriptLength, GC, TTA*^{RSCU}*, TTG*^{RSCU}*, CTA*^{RSCU}*, CTG*^{RSCU}*, GCT*^{RSCU}*, TGT*^{RSCU} and *TGC*^{RSCU}	*TTA*^{RSCU}*, TTG*^{RSCU}*, TCT*^{RSCU}*, TCA*^{RSCU}*, AAG*^{RSCU}*, GAT*^{RSCU}*, GAC*^{RSCU}*, GAA*^{RSCU}*, GAG*^{RSCU} and *AGC*^{RSCU}

For selection of features using mRMR, Mutual Information Difference (MID) was used as method for selection of features on training dataset. The discretization threshold value was kept to 0 and 1. Using discretization threshold of zero for plants, *hexamer score*, *ORF coverage*, *Fickett score*, *Fop*, *ATG*^{RSCU}, *GCT*^{RSCU}, *TAT*^{RSCU}, *CAG*^{RSCU}, *TGT*^{RSCU} and *CGG*^{RSCU} were selected (Table 3.9) from 73 feature set and performed the binary classification producing accuracy of 94.59%. Using similar parameters, mammalian dataset produced *mean ORF coverage*, *GC content*, *Fickett score*, *CUB*, *CCA*^{RSCU}, *CAG*^{RSCU}, *GAT*^{RSCU}, *GAA*^{RSCU}, *TGC*^{RSCU} and *CGT*^{RSCU} exhibiting accuracy of 86.45%. Changing the discretization threshold parameter to 1 produced *hexamer score*, *ORF coverage*, *Fop*, *TTG*^{RSCU}, *ATG*^{RSCU}, *CCG*^{RSCU}, *CAG*^{RSCU}, *GAT*^{RSCU}, *TGT*^{RSCU} and *CGC*^{RSCU} giving accuracy of 94.41% in plants whereas the same parameter generated *mean ORF coverage*, *Fickett score*, *CUB*, *ACA*^{RSCU}, *GCA*^{RSCU}, *CAG*^{RSCU}, *AAG*^{RSCU}, *GAT*^{RSCU}, *GAA*^{RSCU} and *GAG*^{RSCU} producing accuracy of 85.62%.

Feature selection using ReliefF in plants produced *Fickett score*, *ORF coverage*, *SCUO*, *Fop*, *RCB*, *hexamer score*, *mean ORF* coverage, *GC content*, *CUB* and *transcript Length* produced accuracy of 95.27% (Table 3.9). For mammals, ReliefF selected *mean ORF coverage*, EW, RCB, *ORF coverage*, Fop, *hexamer score*, *ORF length*, *transcript length*, TCA^{RSCU} and TGC^{RSCU} with 88.57% accuracy. The results from the analysis show that Chi-square, ReliefF and IG performed similar to the LiRF-FS method, exhibiting similar prediction accuracies as displayed by LiRF-FS method. mRMR on the other hand produced similar prediction accuracy to LiRF-FS in plant species, but the accuracy dropped by 3-4% in mammalian species. The reason for the decrease in accuracy is the non-selection of essential features such as *hexamer score*, *SCUO*, *EW*, *GC content*, *RCB*, *transcript length* and *ORF length*.

Unsupervised Discriminative Feature Selection (UDFS) was also applied on the unified datasets for testing the prediction accuracies and evaluate the selection of optimal features. Implementation of UDFS in plants produced 3 sequence-based and 7 codon-biased features (Table 3.9) predicting lncRNA sequences with an accuracy of 92.76%. In mammalian species, UDFS selected all codon-biased features as optimal set generating an accuracy of 75.05%. As discussed above, the accuracy of identifying the transcripts require selection of principle features such as *hexamer score*, *SCUO*, *EW*, *GC content*. UDFS is an unsupervised approach with $\ell_{2,1}$-norm regularisation which is particularly suitable for finding correlations between samples. Results from UDFS suggest non-sensitivity and non-specificity for FASTA sequence derived features.

3.6 Performance comparison evaluation

To evaluate the predictive power of the framework in lncRNA identification, its performance was measured on four other popular coding-potential alignment-free tools i.e. PLEK (Li, Zhang and Zhou, 2014a), CPAT (Wang *et al.*, 2013), lncScore (Zhao, Song and Wang, 2016) and CPC2 (Kang *et al.*, 2017) (Table 3.10). The comparisons were made for 8 plants and 2 mammalian species. Prediction on test set data in individual species shows that in general, *LiRF-FS* achieves higher accuracy and presents better performance than other tools in individual species prediction. Specifically, the framework performed exceptionally accurate on ZM, OS and ST datasets and comparatively better on ATH and SL datasets with marginal differences in specificity, sensitivity and NPV metrics. Results from the BRA, BOL and ST datasets show marginal differences in the metric values with a difference of 0.5-1% when compared with CPAT. The framework exhibited highest precision and accuracy values in 5 plant species when compared with PLEK and CPAT. When compared with CPC2, the framework displayed superior performance in all the species except ATH where higher metrics were observed for CPC2. An average prediction accuracy difference of 1 – 4% between the framework and CPC2 was detected in ZM, BNA, BRA, BOL, SL and ST species. OS displayed an accuracy difference of 47.17% whereas an average difference of 6.49% was observed in mammalian species between the framework and CPC2.

The prediction accuracies of the framework were comparable with CPAT and lncScore in BRA, BNA, BOL, SL, HS and MM datasets. However, the framework in the ATH dataset generated lower accuracy as compared to CPAT and lncScore. A difference in the accuracy of 2.53% and 2.77% was observed against lncScore and CPAT, respectively. Comparison of the prediction accuracy on ZM and HS datasets shows highest accuracy, precision, sensitivity and specificity in ZM dataset when compared with other tools whereas for HS, the framework displayed highest accuracy, specificity, F1 and MCC against PLEK and CPAT tools. LncScore exhibited the highest performance in the mammalian species. Performance of PLEK in the mammalian and plants datasets was significantly lower in multiple species. Accuracy difference between the framework and PLEK showed an average difference of ~30 – 40% in BNA, BRA, BOL and ZM datasets, ~7 – 15% in HS, MM and ATH, whereas a significant difference of 72.34% was observed in OS species.

Table 3.10: Performance comparison of the framework with PLEK, CPAT, lncScore and CPC2 tools on multiple species.

Tools	**Species**	**ACC**	**SENS**	**SPEC**	**F1-Score**	**NPV**	**MCC**
Framework	ATH	94.51	94.91	94.12	94.51	94.93	0.89
PLEK	ATH	80.82	68.58	92.89	78.91	74.98	0.63
CPAT	ATH	97.28	97.25	97.3	97.28	97.21	0.94
lncScore	ATH	97.04	95.58	99.23	97.37	95.76	0.94
CPC2	ATH	95.99	94.62	97.34	95.96	94.83	0.92
Framework	ZM	94.71	93.6	95.95	94.72	93.11	0.89
PLEK	ZM	65.8	67.03	64.43	65.71	63.8	0.31
CPAT	ZM	94.71	95.21	94.28	94.74	95.75	0.89
lncScore	ZM	94.36	92.63	96.36	94.46	92.18	0.88
CPC2	ZM	91.82	94.27	89.10	91.61	93.34	0.83
Framework	OS	96.95	97.23	96.66	96.95	97.13	0.93
PLEK	OS	24.61	26.92	39.7	32.08	58.79	-0.29
CPAT	OS	93.7	98.29	90.38	94.17	98.49	0.88
lncScore	OS	19.63	99.78	2.03	4	96.15	0.06
CPC2	OS	49.78	13.74	86.9	23.73	49.44	0.009
Framework	BNA	96.73	95.75	97.76	96.73	95.67	0.93
PLEK	BNA	56.77	45.68	68.32	54.75	54.73	0.14
CPAT	BNA	96.86	97.14	96.58	96.86	97.27	0.93
lncScore	BNA	96.35	95.61	97.16	96.38	95.5	0.92
CPC2	BNA	94.64	95.08	94.19	94.63	94.85	0.89
Framework	BRA	95.77	94.54	97.01	95.78	94.65	0.91
PLEK	BRA	61.29	54.93	67.68	60.64	59.91	0.22
CPAT	BRA	96.9	96.39	96.42	96.9	97.43	0.93
lncScore	BRA	96.34	95.74	97.09	96.41	95.78	0.92
CPC2	BRA	94.73	94.30	95.16	94.73	94.33	0.89
Framework	BOL	96.35	95.38	97.32	96.35	95.47	0.92
PLEK	BOL	54.98	44.71	65.31	53.08	54.23	0.1
CPAT	BOL	96.78	96.86	96.75	96.8	96.86	0.93
lncScore	BOL	96.43	94.51	98.51	96.47	94.74	0.93
CPC2	BOL	92.45	89.56	95.33	92.35	90.14	0.85
Framework	SL	97.25	97.63	96.87	97.25	97.58	0.94
PLEK	SL	67.94	70.67	65.17	67.81	68.68	0.35
CPAT	SL	97.98	98.62	97.36	97.98	98.66	0.95
lncScore	SL	97.92	97.53	98.33	97.92	97.51	0.96
CPC2	SL	95.85	97.53	94.16	95.81	97.41	0.91
Framework	ST	95.69	95.55	95.83	95.69	95.32	0.91
PLEK	ST	62.29	54.94	70.06	61.59	59.52	0.25
CPAT	ST	95.36	95.78	94.98	95.38	96.06	0.9
lncScore	ST	95.43	93.78	97.18	95.45	93.66	0.9
CPC2	ST	93.73	95.30	92.08	93.66	94.88	0.87
Framework	HS	91.67	89.82	93.59	91.67	89.87	0.83
PLEK	HS	84.22	72.04	98.11	83.08	76.99	0.72
CPAT	HS	91.47	91.45	91.49	91.47	91.78	0.82
lncScore	HS	93.03	91.46	95.97	93.66	91.46	0.87

CPC2	HS	85.06	74.88	95.62	83.99	78.6	0.71
Framework	MM	89.49	87.91	91.12	89.49	87.95	0.79
PLEK	MM	78.63	66.2	92.1	77.03	72.4	0.6
CPAT	MM	90.63	91.68	89.67	90.66	92.17	0.81
lncScore	MM	93.15	91.31	95.73	93.47	91.39	0.87
CPC2	MM	83.11	72.03	94.56	81.77	76.60	0.68

When F1 values are compared, the framework displays higher F1 score than CPAT and PLEK with marginal differences of 0.004, 0.05, 0.018, 0.019 and 0.05 in BNA, BRA, BOL, SL, ST and MM datasets. Comparison of metric values in mammalian species demonstrates higher accuracy, F1 and MCC values in the HS dataset. Performance in MM dataset illustrates minor differences in the metric values where CPAT performs better than other tools.

The higher sensitivity and specificity values obtained from the comparison indicate that the framework can correctly identify the proportion of true mRNA and lncRNA sequences using sensitivity analysis and it can also identify true negative values with greater accuracy. Results from MCC analysis indicate a near perfect prediction of lncRNA sequences in the test dataset, whereas PLEK displays a random prediction of observation values. CPAT and lncScore on the other hand, performed similarly to the performance obtained from the framework. F1 measure provides information on the accuracy of the classifier with values between 0 and 1. Since the values obtained from the analysis have been scaled to 100, the indication of the performance can be best identified at this scale. In general, comparison of F1 score shows higher F1 values in 5 out of 10 species with minor differences in BNA, BRA, BOL, ST and MM datasets with difference ranging between 0.1-3.12%. Comparison of the results indicate that the features selected and extracted from mRNA and lncRNA sequences from multiple species provide good classification potential. Usage of the iRF classifier in conjunction with the features extracted improves the prediction of lncRNA sequences and hence can be employed for the identification of lncRNA sequences in other FASTA-based datasets.

3.7 lncRNA sub-classification

The classification of lncRNA sequences is important to identification of functional mechanism in the genome. Due to the lower gene expression of lncRNA sequences in RNA-seq datasets, the biological roles of lncRNAs becomes difficult to interpret. Therefore, by identifying the various sub-classes of lncRNA sequences based on their genomic position, valuable insights into the functional mechanism can be understood. Sub-classification analysis was performed on *H. sapiens* (HS) and *M. musculus* (MM) GENCODE datasets based on the rules outlined in Section 2.12. The analysis was performed based on two rules:

1) Rule 1 states that the classification performed for five classes follows the following rules:
    a. Overlap of lncRNA exons with mRNA exons on '+' strand for SOE class.
    b. Overlap of lncRNA introns with mRNA exons on '+' strand for SOI class.
    c. Overlap of lncRNA exons lying on '-' strand with mRNA exons lying on '+' strand for AOE class.
    d. Overlap of lncRNA introns lying on '-' strand with mRNA exons lying on '+' strand for AOE class.
2) Rule 2 states that the classification performed for five classes follows the following rules:
    a. Overlap of lncRNA exons with mRNA exons on '+' strand OR overlap of lncRNA exons with mRNA exons on '-' strand for SOE class.
    b. Overlap of lncRNA introns with mRNA exons on '+' strand OR overlap of lncRNA introns with mRNA exons on '-' strand for SOI class.
    c. Overlap of lncRNA exons lying on '-' strand with mRNA exons lying on '+' strand OR overlap of lncRNA exons lying on '+' strand with mRNA exons lying on '-' strand for AOE class.
    d. Overlap of lncRNA introns lying on '-' strand with mRNA exons lying on '+' strand or overlap of lncRNA introns lying on '+' strand with mRNA exons lying on '-' strand for AOE class.

Here SOE is "Sense Overlap Exonic", SOI is "Sense Overlap Intronic", AOE is "Antisense Overlap Exonic", AOI is "Antisense Overlap Intronic", BDP is "Bidirectional Promoter" and INT is "Intergenic" classes.

Results were obtained from the classification of the sequences using Rules 1 and 2 which have been presented in Table 3.11 and 3.12. 27908 sequences were classified based on 61022 protein-coding sequences scattered across 24 chromosomes in humans. Based on Rule-1, 124 lncRNAs were classified as SOE, whereas 121 were classified as SOI. Classification into antisense category indicates that 181 sequences were classified as AOE whereas 175 were classified as AOI class. On the other hand, 648 sequences were classified into BDP class and 12884 were classified into INT class based on Rule-1. Based on Rule-2, the number of SOE and SOI sequences increased to 152 and 149, respectively. Whereas sequences belonging to

AOE and AOI classes showed significant increase with 431 sequences classified in AOE class and 421 classified as AOI class. Sequences belonging to BDP and INT classes using Rule-2 did not showed significant differences as 556 sequences were classified as BDP and 10542 were classified in INT class. Those sequences belonging to the “antisense_RNA” (ANT) category were larger than the overlap categories. Since the ANT classification is purely based on the strand annotation, identical number of ANT sequences were obtained in both rule-based classification algorithms. A total of 4972 ANT sequences were obtained from the HS dataset which is almost half the number of the INT sequences. ANT contributes as the second highest category, based on the number of sequences classified.

Classification results obtained from MM lncRNA sequence classification using Rule-1 shows a comparatively smaller number of lncRNA sequences, with 54 sequences classified in SOE and SOI classes. Antisense category included 138 sequences classified as AOE class and 135 classified as AOI class. BDP class included 431 sequences, whereas INT included 6923 sequences altogether. Results based on Rule-2 showed similar performance to HS dataset with an increase of sequences classified in sense and antisense overlap categories. 74 sequences were classified in SOE and SOI classes, 286 and 278 were classified in AOE and AOI classes, respectively, whereas the number of sequences classified as BDP and INT remained similar to those obtained by Rule-1. When compared to the HS data, MM generated a total of 1725 ANT sequences using PBC. Although, the number of sequences classified as ANT in MM is much lower than that obtained in HS, the number of ANT sequences is still greater than sense-overlap, antisense-overlap and bidirectional classes.

Table 3.11: Statistics of lncRNA sequences annotated in various sub-classes with GENCODE datasets using Rule 1.

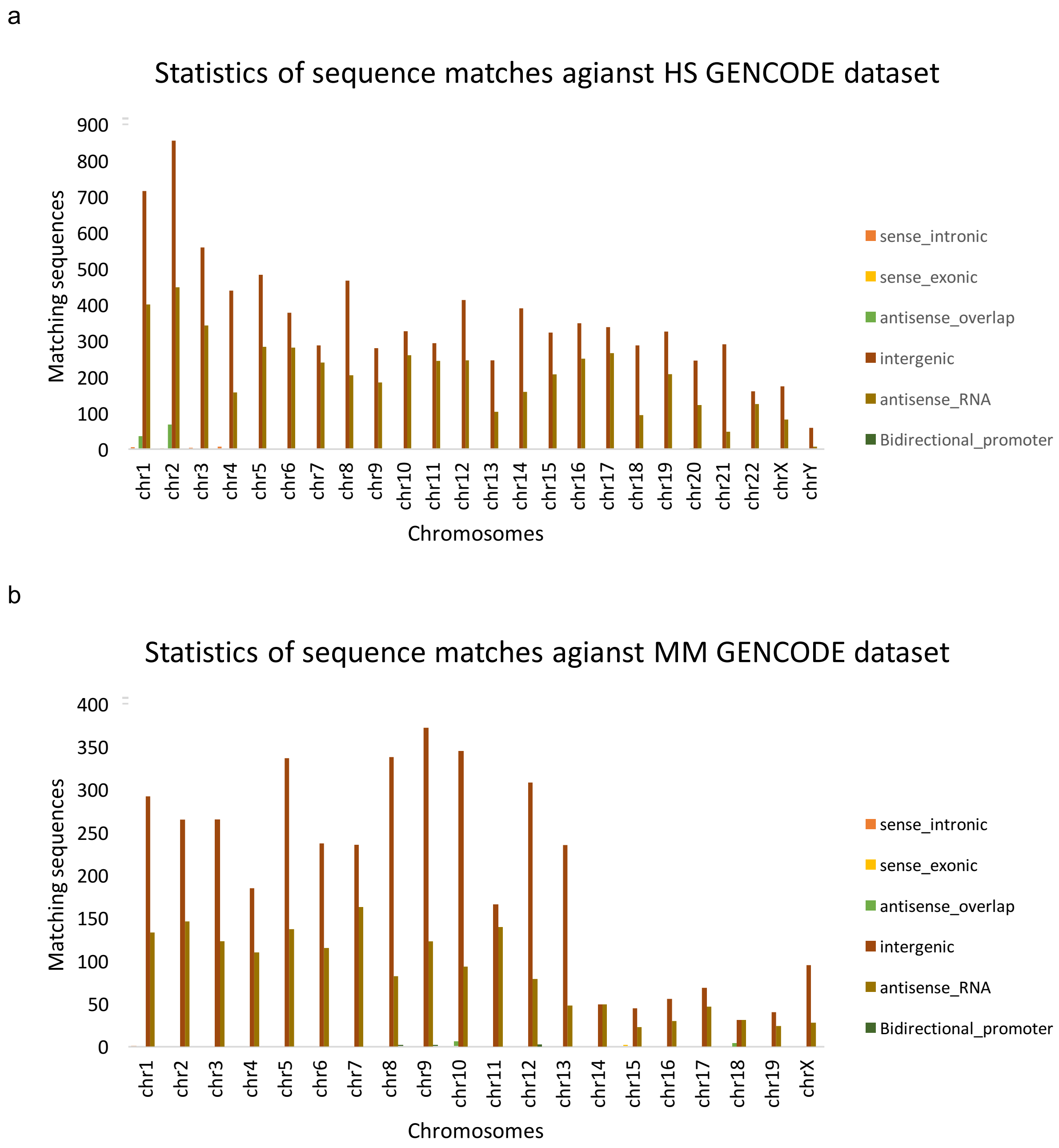
Species	**SOE**	**SOI**	**AOE**	**AOI**	**ANT**	**BDP**	**INT**
HS	124	121	181	175	4972	648	12884
MM	54	54	138	135	1725	431	6923

Table 3.12: Statistics of lncRNA sequences annotated in various sub-classes with GENCODE datasets using Rule 2.

Species	**SOE**	**SOI**	**AOE**	**AOI**	**ANT**	**BDP**	**INT**
HS	152	149	431	421	4972	556	10542
MM	74	74	286	278	1725	431	6923

Results from the classification using two different rules show differences in the number of lncRNA sequences classified. As compared to the sense and antisense overlap classes, a significant number of sequences are classified into INT classes, with approximately 50-60 times more sequences than in the rest of the classes. Sequences which have been classified in the BDP class show a comparatively higher number than sense and antisense classes. Using Rule-2 in the HS dataset, a higher number of antisense classes were observed, with 852 sequences classified in antisense category, as compared to 556 sequences classified as BDP, showing a difference of 296 sequences. Results from the MM dataset do not show significant increase in sequences categorised in antisense class, where 564 sequences were classified as antisense class and 431 sequences classified as BDP class, showing a minor increase in the classified sequences.

The increase in the number of sequences in sense and antisense classes using Rule-2 is due to scanning of sequences in '+' and '-' strands for searching sense and antisense overlaps. Experimental studies suggest that lncRNA sequences are often transcribed from the antisense '-' strand of the DNA (Ma, Bajic and Zhang, 2013) and the protein-coding genes are transcribed from the sense '+' strand. Based on these principles, the classification shows fewer genes classified into these classes based on Rule-1. However, Rule-2 on the other hand, considers overlap of lncRNA sequences on both the strands, with respect to protein-coding genes. According to the lncRNA sequence annotation obtained from GENCODE database, the SOE and SOI lncRNA sequences can overlap on '-' strand of DNA which will still be considered as sense overlaps. On the other hand, the antisense RNA sequences from GENCODE shows overlap on either strand. This means that if the mRNA sequence occurs on '+' strand and the lncRNA sequence overlaps the position of mRNA sequence on the '-' strand or the mRNA sequence lies on '-' strand and lncRNA sequence overlaps the position on '+' strand, it will be considered as antisense lncRNA.


a

Statistics of sequence matches agianst HS GENCODE dataset

b

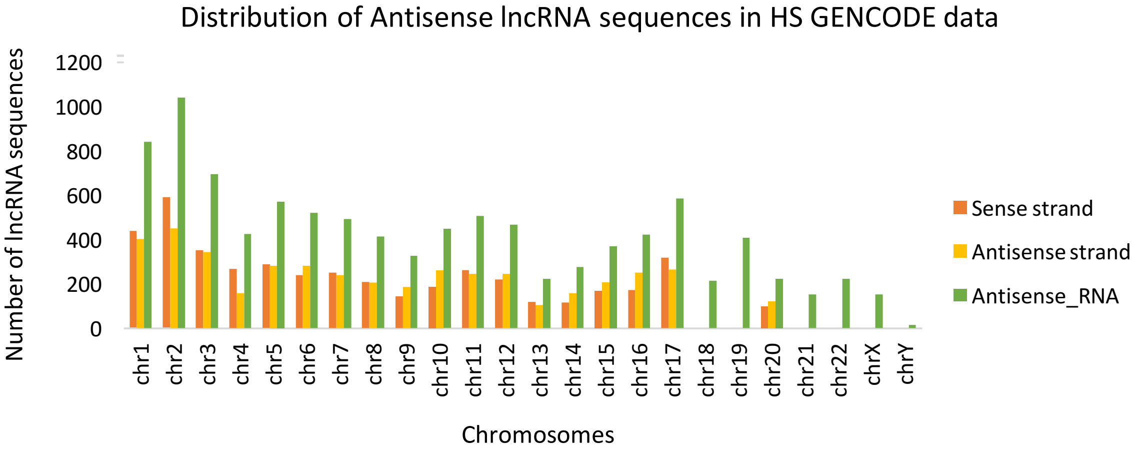
Statistics of sequence matches agianst MM GENCODE dataset

Figure 3.16: Comparison of classification results obtained from PBC versus GENCODE results for (a) *H. sapiens* dataset, and (b) *M. musculus* dataset.

Results of the sub-classification were compared to the annotation information obtained from the GENCODE datasets. Figure 3.16 shows the statistics of the total number of annotation matches of the PBC approach, with HS and MM GENCODE data. The results of the classification were performed on individual chromosomes to obtain the total matching annotation results. Class matching based on chromosome provides a clear picture of the overall classification analysis. Results from the analysis clearly show that intergenic lncRNAs produced the highest number of matches followed by antisense lncRNA, sense intronic and sense overlapping classes. No matching bidirectional lncRNA sequences were detected in the result set, however, bidirectional lncRNA sequences were annotated using PBC approach.

In the HS dataset, $\sim 60\%$ of the classified lncRNA sequences displayed identical match based on Rule-2 classification. The highest matches were produced by the intergenic class. Antisense lncRNA on the hand, produced a comparatively lower match than the intergenic class. Sense intronic and exonic overlap classes showed little overlap of sequence classification with GENCODE data. On the other hand, classification using PBC on MM produced a matching percentage of 52-55% on different chromosomes. As illustrated in Figure 3.16b, the highest matches have been attributed to intergenic lncRNA sequences, followed by the antisense lncRNA class. Sense intronic overlap class contributed minimally to the overall sequence match, whereas sense exonic overlap class did not contribute at all. Similar to HS data, results from MM data did not exhibited bidirectional lncRNA matches.

a



b

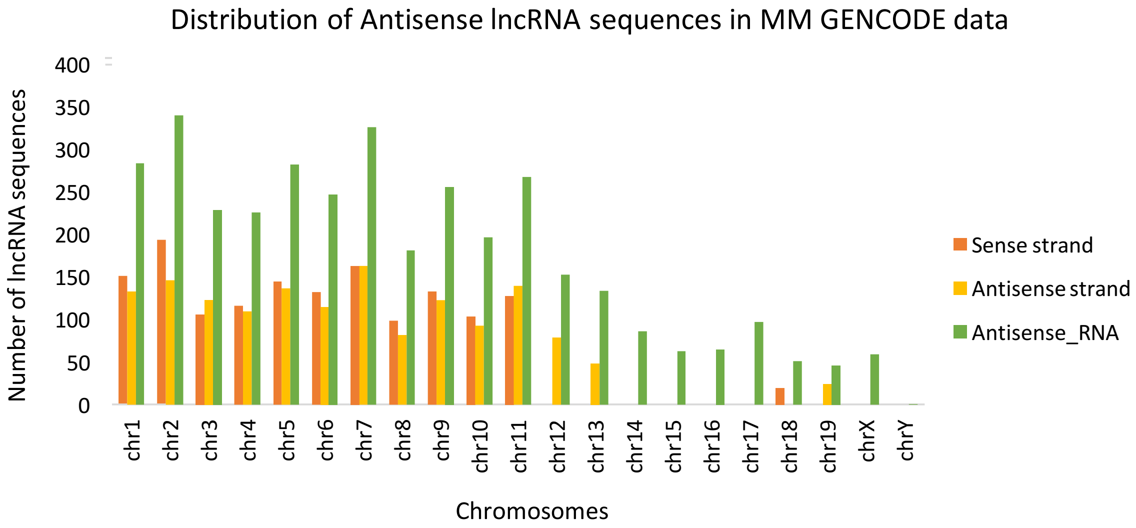


Figure 3.17: Genome-wide density distribution of antisense lncRNA sequences on sense and antisense strands in (a) HS GENCODE, and (b) MM GENCODE datasets.

An analysis of the distribution of antisense lncRNA sequences on sense and antisense strands of DNA was performed in the HS and MM GENCODE datasets (Figure 3.17). From the figure, a large proportion of sequences have been identified as antisense lncRNAs in both datasets. The distribution of antisense sequences shows that sequences are equally distributed on both the strands for the majority of the chromosomes. In HS dataset (Figure 3.17a), only a few chromosomes namely, *chr2*, *chr4* and *chr17* have a comparatively greater number of lncRNA sequences distributed on the sense strand, whereas in most of the cases the number of lncRNA sequences annotated on antisense strand is comparatively higher. However, the difference is marginal and therefore can be considered as equal.

In the MM dataset (Figure 3.17b), a similar distribution pattern can be observed with equal dispersion of antisense RNA sequences on both the strands. ANT sequences distributed on sense strand in *chr2* and *chr13* display higher proportion than the ANT sequences distributed on the antisense strand, whereas the rest of the chromosomes display equal distribution. No antisense lncRNA sequences were observed in *chrY* from the MM GENCODE data.

The density distribution of the antisense lncRNA sequences in GENCODE datasets suggests that the lncRNA sequences annotated as “antisense_RNA” on the sense strand of the DNA does not depend on its position on the DNA strand. The results suggest that their distribution is completely independent, and the majority of the sequences annotated on sense strand may overlap the coordinates of protein-coding sequences. Such sequences may be classified as antisense overlaps and hence are hidden from the data available. Analysis of the pattern indicate that the distribution follows Rule-2 implemented in the PBC approach, where a greater number of lncRNAs were annotated as antisense lncRNAs. Results obtained from the GENCODE data demonstrate a dissimilar distribution of the sense and antisense overlapping sequences compared to those obtained from the PBC approach. These results indicate a contradiction to published research studies (Ma, Bajic and Zhang, 2013).

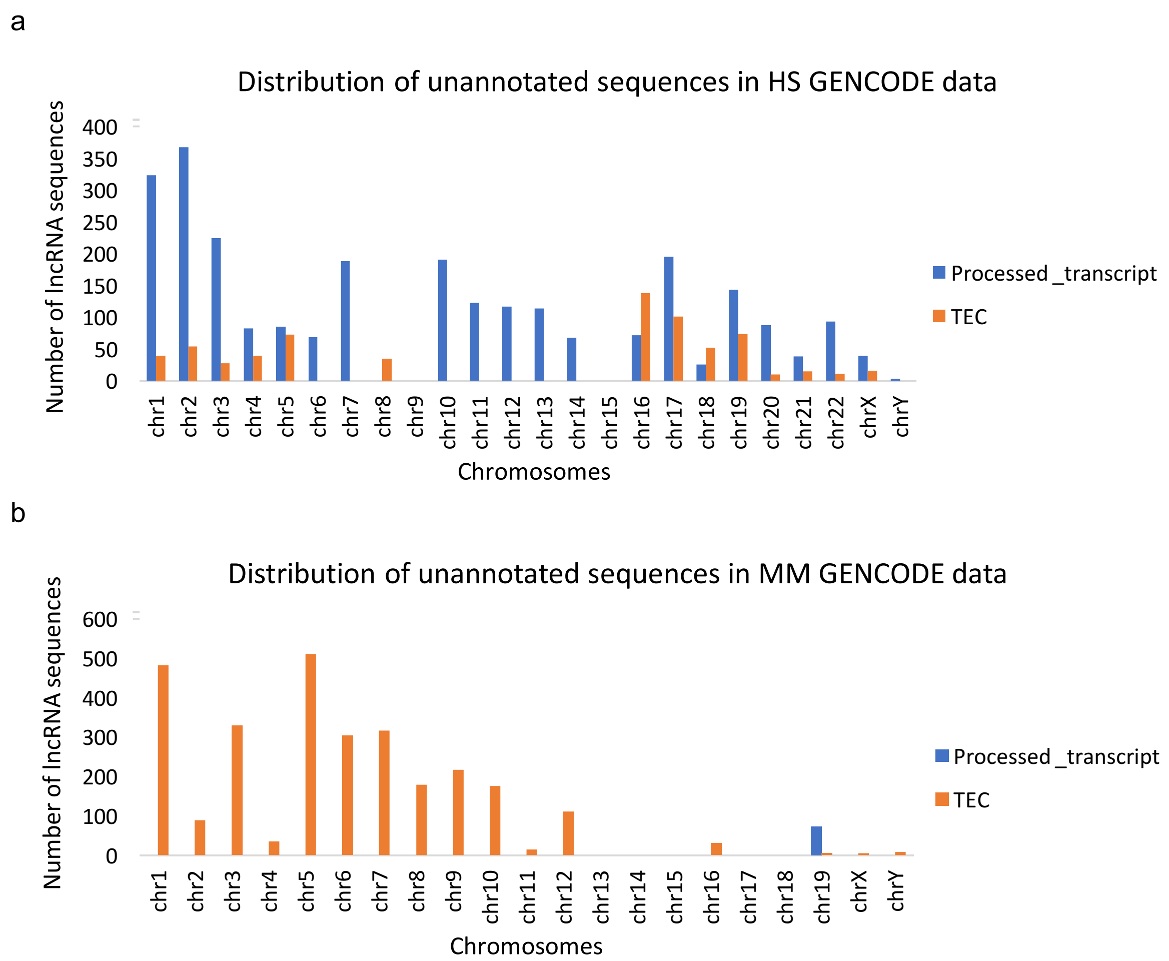


Figure 3.18: Genome-wide density distribution of processed and TEC lncRNA transcript sequences in (a) HS GENCODE, and (b) MM GENCODE datasets.

The GENCODE dataset obtained from HS and MM species contains a large proportion of unannotated sequences classified as “Processed Transcript” (PT) and “TEC” where TEC stands for “To be Experimentally Confirmed”. An analysis of the genome-wide distribution of these sequences was performed (Figure 3.18). Sequence distribution in HS dataset (Figure 3.18a) show that sequences annotated as “processed transcript” are significantly higher than

TEC sequences particularly in *chr1*, *chr2*, *chr3*, *chr7*, *chr10*, *chr17*, *chr20* and *chr22*. Sequences distributed on the remaining chromosomes are also higher, but the difference is moderate. TEC sequences annotated in *chr16* show higher proportion than “processed transcript” sequences.

The distribution of lncRNA classified sequences on the MM dataset (Figure 3.18b) show a dissimilar pattern to the PBC-based class distribution observed on HS data. Distribution of the GENCODE annotation demonstrates significantly higher proportion of TEC sequences and lower proportion of “processed transcript” sequences in MM. The proportion is significantly higher in *chr1*, *chr3*, *chr5*, *chr6*, *chr7*, *chr8*, *chr10* and *chr13*. The “processed transcript” sequences display higher proportion only on *chr2*, *chr11*, *chr17*, *chr18*, *chr19* and *chrX*. *chrY* displayed extremely lower proportion of unannotated transcripts in MM data.

Sub-classification analysis of “processed transcript” and TEC sequences using the PBC approach was performed. These sequences were classified either as: processed transcript intergenic (PT-INT), processed transcript sense overlapping exonic (PT-SOE), processed transcript sense overlapping intronic (PT-SOI), processed transcript antisense overlapping exonic (PT-AOE), processed transcript antisense overlapping intronic (PT-AOI), processed transcript bidirectional promoter (PT-BDP), processed transcript antisense (PT-ANT), TEC intergenic (TEC-INT), TEC sense overlapping exonic (TEC-SOE), TEC sense overlapping intronic (TEC-SOI), TEC antisense overlapping exonic (TEC-AOE), TEC antisense overlapping intronic (TEC-AOI), TEC bidirectional promoter (TEC-BDP) and TEC antisense (TEC-ANT).

Results of the sub-classification analysis on HS dataset (Table 3.13) show that greater number of unannotated sequences were classified as PT-ANT, TEC-ANT, PT-INT and TEC-INT, whereas only a fraction of the sequences were classified as SOE, SOI, AOE and AOI. Sequences classified as PT-BDP and TEC-BDP displayed the third higher proportion that sense and antisense overlaps occupying approximately 15% of the total “processed transcript” and TEC sequences in HS data. GENCODE data also contains “Blank” annotation where lncRNA sequences are not classified in any class and therefore have been left unannotated. Analysis of this group suggests that sequences distributed in 3 chromosomes namely, *chr11*, *ch12* and *chr22* were annotated as intergenic (INT) class using PBC. However, the proportion of “Blank” category is low when compared to other categories.

Sub-classification analysis on MM dataset (Table 3.14) also shows a similar distribution pattern of PT-ANT, PT-INT, TEC-ANT and TEC-INT classes, as observed in the HS dataset displaying a significantly higher proportion of lncRNA sequences classified as “antisense_RNA” and “intergenic”. As observed from the genome-wide density distribution diagram (Figure 3.14b), the number of sequences annotated as TEC is significantly higher than PT sequences. Sequences annotated as PT-INT display a higher proportion than the rest of the sequences annotated in chromosomes *chr1* to *chr9*, *chr11* to *chr17*, *chr19* and *chrX*. *Chr10* and *chr18* shows equal distribution in all the classes. Sequences annotated as PT-ANT display the highest proportion of annotated sub-class in all the chromosomes. In contrast to HS data, sequences annotated as PT-BDP in MM data are comparatively equally distributed as the rest of the classes i.e. (PT-SOE, PT-SOI, PT-AOE and PT-AOI). Distribution of TEC sequences however, display a slightly different pattern. Sequences annotated as TEC-SOE, TEC-SOI, TEC-AOE, TEC-AOI and TEC-BDP can be observed in chromosomes *chr1* to *chr9*. Distribution of the sequences annotated as the above-mentioned classes cannot be observed in the rest of the chromosomes, where 90-95% of the sequences are classified as TEC-ANT and TEC-INT. Unannotated sequences “Blanks” can only be observed in chr4 and chr13 classified in INT class.

Table 3.13: Sub-classification statistics of unannotated sequences of HS GENCODE dataset using PBC approach.

Chromosomes	PT-INT	PT-SOE	PT-SOI	PT-AOE	PT-AOI	PT-BDP	PT-ANT	TEC-INT	TEC-SOE	TEC-SOI
1	128	5	4	0	0	6	182	24	1	1
2	268	1	1	1	1	3	287	37	0	0
3	109	3	3	2	2	5	122	16	0	0
4	28	0	0	0	0	0	44	24	0	0
5	24	9	9	1	1	11	43	49	2	2
6	34	1	1	2	2	3	58	24	1	1
7	64	11	11	0	0	7	102	28	0	0
8	35	0	0	2	2	1	42	19	1	1
9	46	1	1	0	0	1	23	16	0	0
10	37	1	1	18	18	8	52	17	0	0
11	92	2	2	0	0	2	86	52	0	0
12	62	6	5	0	0	2	61	51	1	1
13	39	1	1	0	0	1	12	16	0	0
14	39	2	2	0	0	1	26	14	0	0
15	54	2	2	0	0	8	21	28	0	0

16	17	9	9	0	0	3	40	61	1	1
17	97	7	7	1	1	9	102	33	2	2
18	9	2	2	0	0	2	12	19	1	1
19	79	5	5	2	2	6	76	28	1	1
20	69	0	0	0	0	0	57	5	0	0
21	2	0	0	0	0	0	18	12	0	0
22	32	0	0	1	1	1	25	4	0	0
X	16	9	9	0	0	0	25	11	0	0
Y	2	0	0	0	0	0	1	0	0	0

Chromosomes	TEC-AOE	TEC-AOI	TEC-BDP	TEC-ANT	PT	TEC	Blanks
1	0	0	1	15	143	27	
2	1	1	3	25	275	42	
3	1	1	1	17	124	19	
4	0	0	1	19	28	25	
5	0	0	0	34	55	53	
6	0	0	1	12	43	27	
7	0	0	1	21	93	29	
8	2	2	2	18	40	27	
9	1	1	1	7	49	19	
10	0	0	0	14	83	17	
11	1	1	1	35	98	55	1 INT
12	2	2	1	52	75	58	3 INT
13	1	1	1	12	42	19	
14	1	1	0	9	44	16	
15	1	1	0	28	66	30	
16	5	5	3	56	38	76	

17	1	1	4	43	122	43	
18	1	0	1	21	15	23	
19	0	0	6	41	99	36	
20	0	0	0	4	69	5	
21	0	0	0	7	2	12	
22	2	2	2	6	35	10	3 INT
X	1	1	1	6	34	14	
Y	0	0	0	0	2	0	

Table 3.14: Sub-classification statistics of unannotated sequences of MM GENCODE dataset using PBC approach.

Chromosomes	PT-INT	PT-SOE	PT-SOI	PT-AOE	PT-AOI	PT-BDP	PT-ANT	TEC-INT	TEC-SOE	TEC-SOI
1	38	2	3	2	2	5	69	200	3	3
2	70	5	4	4	4	2	95	36	3	3
3	58	2	2	4	3	5	57	169	3	3
4	43	1	1	1	1	5	48	16	0	0

5	55	1	1	0	0	9	56	236	10	10
6	34	2	2	2	2	5	43	144	1	1
7	49	3	3	4	4	7	86	144	3	3
8	36	0	0	0	0	1	36	75	0	0
9	49	2	2	3	2	6	63	90	0	0
10	8	0	0	2	2	5	27	103	0	0
11	49	3	3	2	2	2	77	4	0	0
12	49	1	1	1	1	2	44	52	1	1
13	19	0	0	0	0	0	20	65	0	0
14	44	0	0	0	0	0	65	14	0	0
15	10	1	1	1	1	2	15	12	0	0
16	27	6	6	5	5	4	25	23	0	0
17	34	2	2	2	2	4	50	9	0	0
18	4	4	4	2	2	5	37	11	0	0
19	26	0	0	3	2	2	21	1	0	0
X	67	1	1	0	0	1	58	4	0	0

Chromosomes	TEC-AOE	TEC-AOI	TEC-BDP	TEC-ANT	PT	TEC	Blanks
1	2	2	7	261	52	217	
2	0	0	2	47	89	44	
3	1	1	4	147	74	181	
4	2	2	3	19	52	23	2 INT
5	6	6	10	262	66	278	
6	3	3	8	140	47	160	
7	6	6	10	131	70	172	
8	2	0	2	84	37	79	
9	3	3	3	108	64	99	
10	0	0	1	88	17	104	
11	0	0	0	9	61	4	
12	1	1	1	61	55	57	
13	0	0	3	50	19	68	1 INT
14	0	0	0	16	44	14	
15	0	0	0	10	16	12	
16	0	0	1	7	53	24	

17	0	0	0	12	46	9	
18	0	0	0	6	21	11	
19	0	0	0	3	33	1	
X	0	0	0	4	70	4	

3.8 Summary

This chapter has discussed the results, analysis and evaluation of lncRNA classification in plants and mammalian reference datasets. Comparison was performed between different machine learning classifiers on various evaluation measures. Comprehensive analysis of plants and mammals was performed by combining transcript sequences from multiple species. Performance of individual features was also evaluated on unified datasets of plants and mammals. Comparison of the framework against known coding potential computation tools was also conducted on various evaluation metrics on sequence derived from reference datasets. Demonstration of LiRF-FS analysis was performed on the unified datasets for obtaining optimal feature sets in plants and mammalian species. RIT analysis was performed for obtaining prevalent feature interactions which were compared against results from LiRF-FS analysis. Results from LiRF-FS were assessed against other feature selection tools/methods.

This chapter also presented results of the lncRNA sub-classification based on two rules. Classification analysis was performed on mammalian GENCODE transcript sequences. Results of the classification analysis obtained from the framework were compared against the GENCODE annotations of humans and mouse sequences. Results from the classification generated moderate number of matching sequences based on Rule-2. Additionally, large proportion of unannotated sequences were found to be annotated with PBC into sense, antisense, bidirectional and intergenic classes.

CHAPTER 4: CASE STUDY ANALYSIS

4.1 Introduction

This chapter presents the results on the identification of novel flowering DE protein-coding genes from the *A. thaliana* apical-shoot dataset. Following this, it presents the results obtained from the identification of lncRNA sequences in plant RNA-seq datasets. It also presents results of sub-classification analysis of lncRNA data predicted from three plant RNA-seq datasets. Furthermore, results from the function prediction of lncRNA sequences using a Bayesian method is also discussed. This chapter also displays results from performance benchmarking analysis with known CPC tools. An analysis of the results obtained has been discussed in each section.

This chapter provides details and the implementation of the framework on two plant RNA-seq time-series transcriptome datasets: *A. thaliana* and *Z. mays*. The lncRNA and protein-coding sequences are extracted from the RNA-seq datasets are used for classification and function prediction. Results from the analysis are validated against the annotated lncRNA sequences and the performance of the classification is compared against the known coding potential computation tools. The functions of lncRNA sequences predicted are validated against the experimentally determined functions from plant transcriptome studies.

4.2 Case study 1: *A. thaliana* apical shoot meristem RNA-seq dataset

4.2.1 Differential Expression (DE) analysis

Results obtained from the DGE of five sample pairs (S10-S14) were computed in “Against S7” and “Step analysis” manner as detailed in Table 2.2 of Chapter 2. When samples from transition phase were compared with sample 7, 5266 DEGs were obtained for S7-S10, 2841 genes for S7-S11, 4760 for S7-S12, 6337 for S7-S13 and 2532 genes for S7-S14 sample pair (Figure 4.1a). DGE using “Step analysis” was performed to identify DEGs from the previous day which yielded fewer genes as compared to that obtained from the “Against S7” sample pairs (Figure 4.1b). However, significantly greater number of genes were obtained using Cuffdiff in S9-S10, S10-S11, S13-S14 and S15-S16. DESeq on the other hand, produced higher genes than edgeR in S11-S12, S12-S13, S13-S14 and S15-S16.

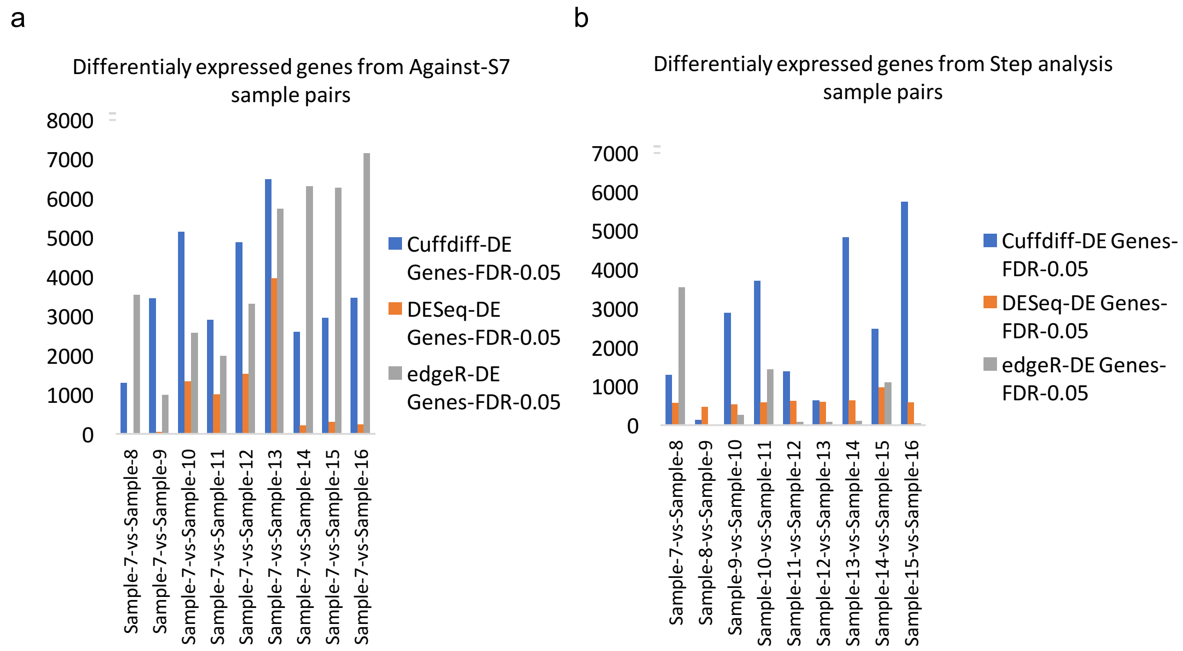


Figure 4.1: Density distribution of DE genes observed using Cuffdiff, DESeq and edgeR tools in (a) Against S7 sample pairs, and (b) Step analysis sample pairs.

To retrieve the true positive values from the analysis, DEGs obtained from Cuffdiff, DESeq and edgeR were overlapped and intersection of the DEG overlaps were obtained for each "Against S7" and "Step analysis" sample pairs. By overlapping Cuffdiff, DESeq and edgeR, 418 genes were found for S7-S10 with FDR <= 0.05. Using the same cutoff, S7-S11 generated 277 genes, S7-S12 produced 520 genes, S7-S13 gave 1,534 genes and S7-S14 gave 150 genes (Table 4.1). On the other hand, 28 genes were found for S9-S10, 3 genes for S10-S11, 7 genes for S11-S12, 38 genes for S12-S13 and 74 genes were found for S13-S14. Overlapping genes were also found for Cuffdiff-edgeR, DESeq-edgeR and Cuffdiff-edgeR-DESeq pairs. From Cuffdiff-DESeq-edgeR overlap, 690 genes were identified in "Against S7" and 19 genes in "Step analysis" which are significantly expressed in more than one sample pairs. This set of common genes is referred to as CGenes in the following analysis.

Results show that both Cuffdiff and edgeR display significant numbers of DEGs in S7-S10, S7-S12 and S7-S13. Overlapping of genes can be visualized by Venn diagrams constructed for transition phase samples. Results from the intersection of Cuffdiff-DESeq-edgeR, Cuffdiff-DESeq and Cuffdiff-edgeR show that the number of DE genes decreases in "Step analysis" sample pairs as compared to "Against S7" sample pairs. edgeR additionally displays a large number of DE genes in S7-S14, S7-S15 and S7-S16 which are not notably identified by Cuffdiff or edgeR.

Table 4.1: Number of overlapping DEGs found in Cuffdiff, DESeq and edgeR results with FDR <= 0.05.

Against S7					**Step analysis**				
Sample pairs	**Cuffdiff-DESeq-edgeR**	**Cuffdiff-DESeq**	**Cuffdiff-edgeR**	**edgeR-DESeq**	**Sample pairs**	**Cuffdiff-DESeq-edgeR**	**Cuffdiff-DESeq**	**Cuffdiff-edgeR**	**edgeR-DESeq**
S7-S10	418	497	870	1170	S9-S10	28	32	241	37
S7-S11	277	348	550	887	S10-S11	3	4	1347	5
S7-S12	520	544	997	1455	S11-S12	7	38	74	8
S7-S13	1534	1674	2138	3630	S12-S13	38	90	78	41
S7-S14	150	150	931	219	S13-S14	74	101	105	77

On the contrary, Cuffdiff displays the maximum number of DE genes from “Step analysis” results as compared to DESeq and edgeR. By comparing the results of Cuffdiff with DESeq and edgeR, it can be clearly observed that the overlap from Cuffdiff-edgeR was more significant than Cuffdiff-DESeq or DESeq-edgeR. This difference can be clearly observed in “Step analysis” for S10-S11 where 1347 genes were found to be common for Cuffdiff-edgeR as compared to 4 genes obtained from Cuffdiff-DESeq results. Thus, the total number of common genes was significantly reduced for Cuffdiff-DESeq-edgeR intersection which is primarily due to a smaller gene count obtained in Cuffdiff-DESeq. Thus, only 1% of the genes were found to be common for Cuffdiff-DESeq-edgeR confirming that the decrease in the overlap is mostly due to DESeq results.

4.2.2 GO enrichment and pathway analysis of DE genes during transition phase

Results of the GO enrichment analysis applied to CGenes were classified in three categories: Biological Process (BP), Molecular Function (MF) and Cellular Component (CC). Results from GO enrichment (Figure 4.2) of common genes obtained from “Against S7” sample pairs show 664 genes were significantly enriched in BP and CC ontologies with p-values < 0.05. Whereas those obtained from “Step Analysis” sample pairs show 18 genes significantly enriched only in the BP ontology with p-value < 0.05. From the pathway analysis of “Against S7” DEGs, 30 genes have been found to be involved in Glucosinolate Biosynthesis, 2-Oxocarboxylic acid metabolism, Sulfur metabolism, Cysteine and methionine metabolism with FDR ≤ 0.05 whereas for “Step analysis” only 4 genes were found to be involved in 2-Oxocarboxylic acid metabolism, C5-Branched dibasic acid metabolism, Valine, leucine and isoleucine biosynthesis.

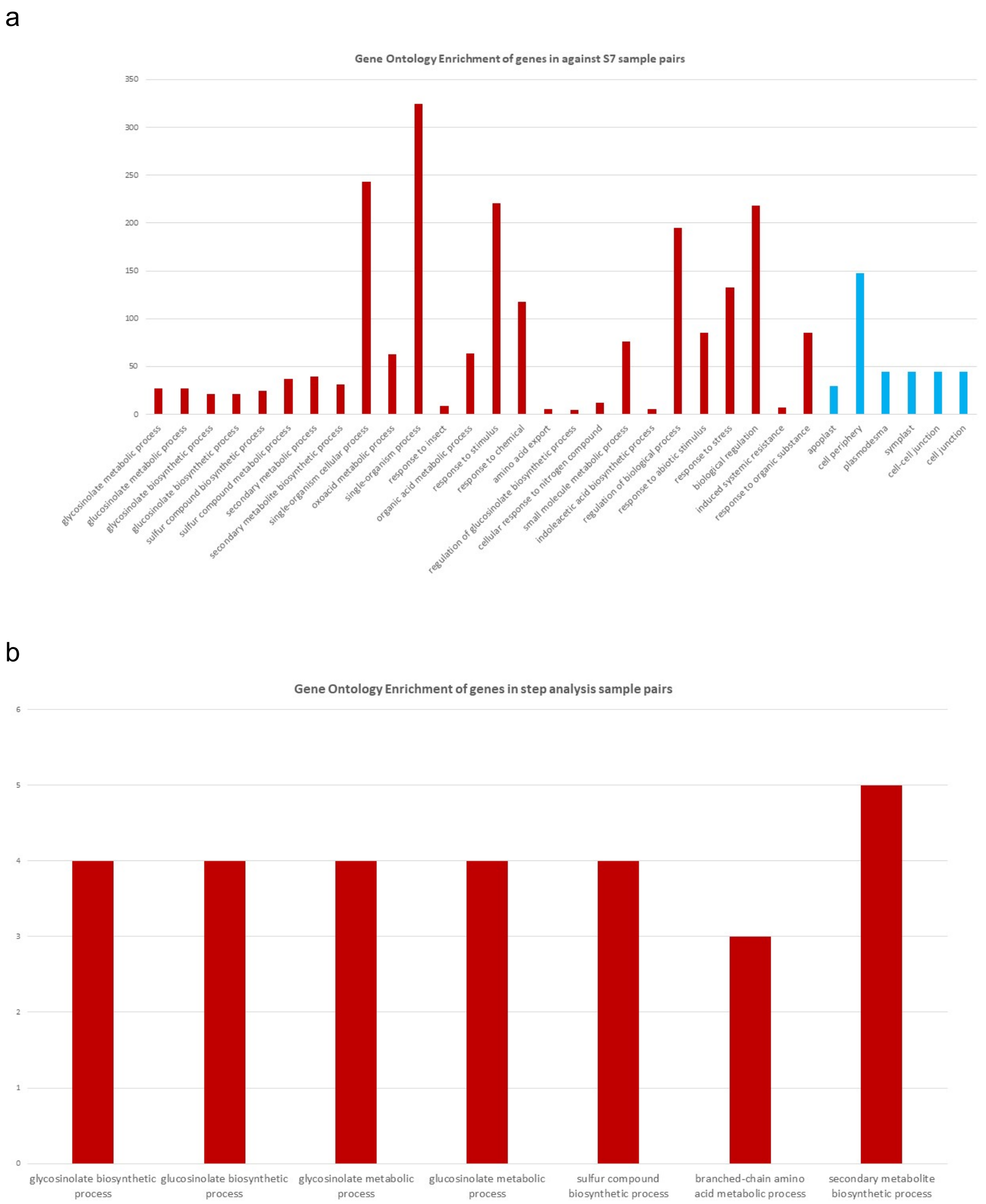
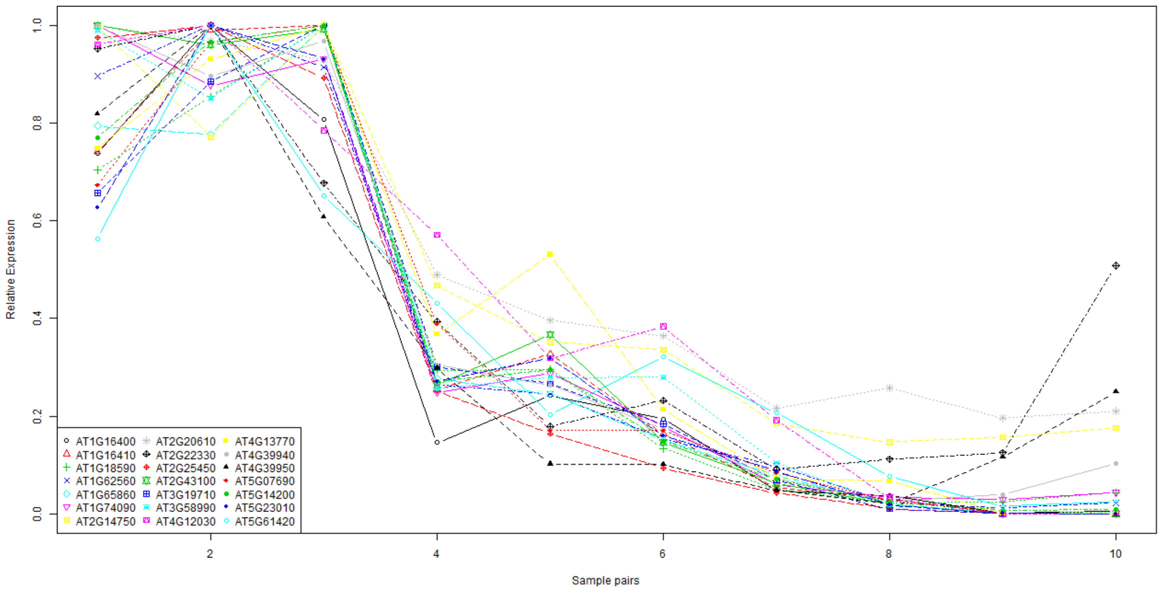


Figure 4.2: GO enrichment functional classification results of common genes from (a) “Against S7” sample pairs, and (b) “Step analysis” sample pairs. Bars coloured in red represent genes enriched in “Molecular Function” whereas bars coloured in blue represent genes enriched in “Cellular Component”.

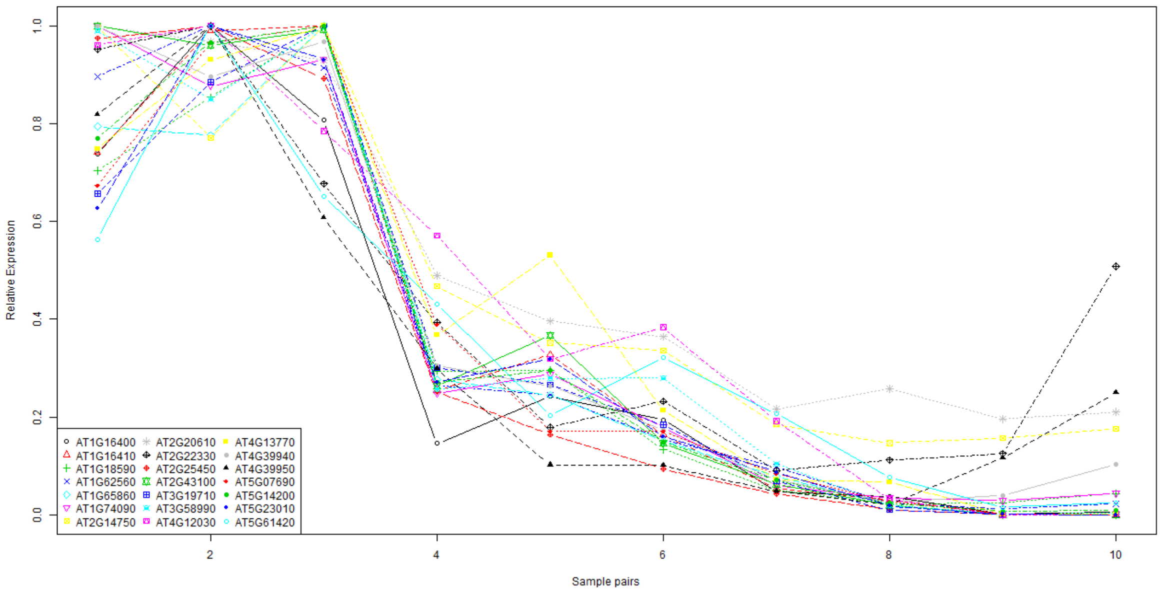
Results from GO enrichment analysis were used to obtain expression profiles of the genes involved in metabolic processes involved in plant defense. Figure 4.3 shows the relative expression profiles of the genes expressed in “Against S7” and “Step analysis” sample pairs that play major roles in Glucosinolate Biosynthetic Process (GluBP), Glycosinolate Biosynthetic

Process (GlyBP), Glucosinolate Metabolic Process (GluMP), Glycosinolate Metabolic Process (GlyMP), Sulfur Compound Biosynthetic Process (SCBP) and Sulfur Compound Metabolic Process (SCMP). 21 genes have been found to be associated with GluBP and GlyBP, 27 associated with GluMP and GlyMP, 25 associated with SCBP and 37 have been found to be associated with SCMP. From the expression profiles in "Against S7": *ACO1*, *ACO2*, *APS1* and *AT4G05090* display different behavior where expression varies between 1 and 0.6 for SCMP. In SCBP, *CYSD1* expression value remains constant whereas for *CYP83B1*, the expression display "zig-zag" pattern. In GBP, only *CYP83B1* shows variable expression. Apart from these genes, certain other genes such as *TGG1* and *TGG2* show a "zig-zag" expression pattern which encodes myrosinase enzymes and helps in the breakdown of glucosinolates (Barth and Jander, 2006). As compared to these genes, *CYP83B1* and *CYP83A1* are expressed in the SCMP, SCBP and GluBP. These encode non-redundant enzymes which also metabolise oximes in glucosinolate biosynthesis (Naur *et al.*, 2003). Similarly, *ACO1* and *ACO2* in the SCMP also differ in their expression profiles despite being similar in structure and function.

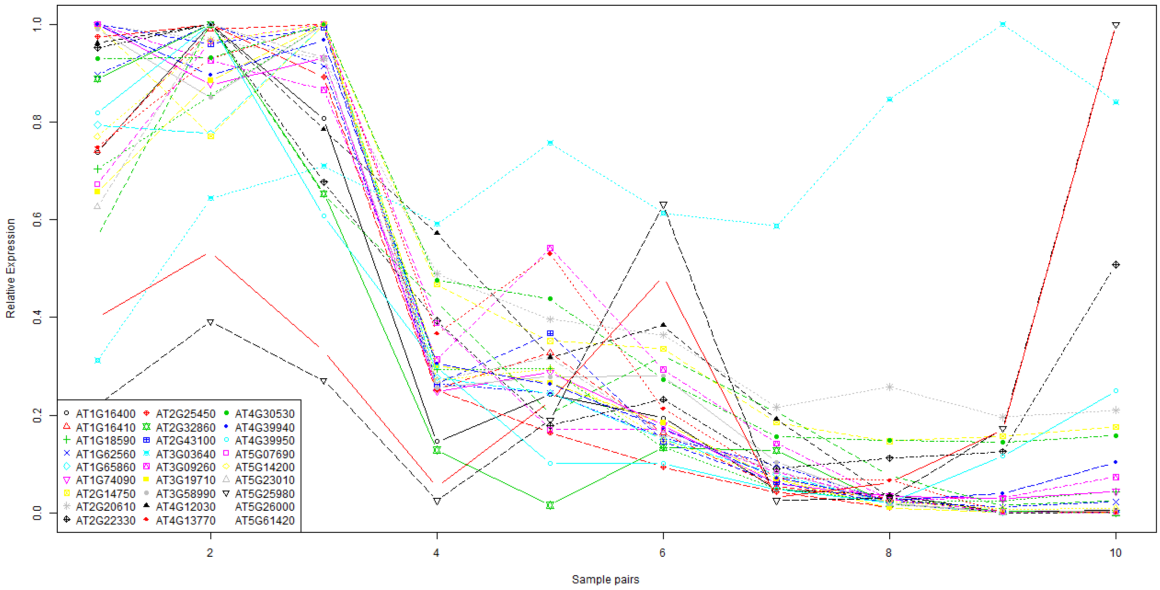
a


Relative Expression
1.0
0.8
0.6
0.4
0.2
0.0
2
4
6
8
10
Sample pairs
AT1G16400
AT1G16410
AT1G18590
AT1G62560
AT1G65860
AT1G74090
AT2G14750
AT2G20610
AT2G22330
AT2G25450
AT2G43100
AT3G19710
AT3G58990
AT4G12030
AT4G13770
AT4G39940
AT4G39950
AT5G07690
AT5G14200
AT5G23010
AT5G61420

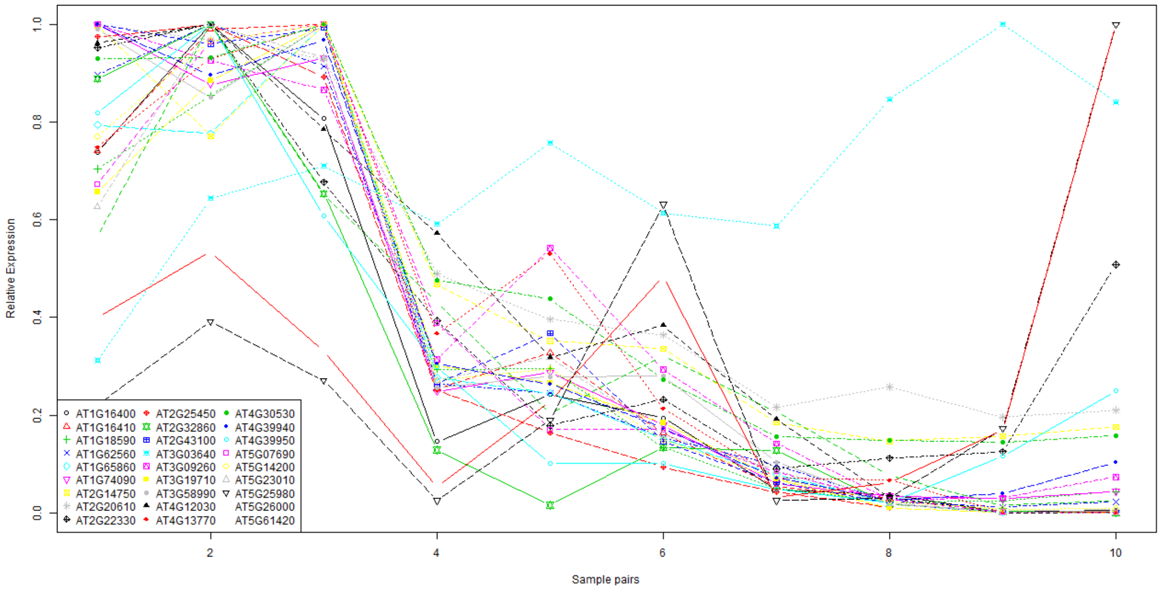
b


Relative Expression
1.0
0.8
0.6
0.4
0.2
0.0
2
4
6
8
10
Sample pairs
AT1G16400
AT1G16410
AT1G18590
AT1G62560
AT1G65860
AT1G74090
AT2G14750
AT2G20610
AT2G22330
AT2G25450
AT2G43100
AT3G19710
AT3G58990
AT4G12030
AT4G13770
AT4G39940
AT4G39950
AT5G07690
AT5G14200
AT5G23010
AT5G61420

c


Relative Expression
0.0
0.2
0.4
0.6
0.8
1.0
Sample pairs
2
4
6
8
10
AT1G16400
AT1G16410
AT1G18590
AT1G62560
AT1G65860
AT1G74090
AT2G14750
AT2G20610
AT2G22330
AT2G25450
AT2G32860
AT2G43100
AT3G03640
AT3G09260
AT3G19710
AT3G58990
AT4G12030
AT4G13770
AT4G30530
AT4G39940
AT4G39950
AT5G07690
AT5G14200
AT5G23010
AT5G25980
AT5G26000
AT5G61420

d


Relative Expression
0.0
0.2
0.4
0.6
0.8
1.0
Sample pairs
2
4
6
8
10
AT1G16400
AT1G16410
AT1G18590
AT1G62560
AT1G65860
AT1G74090
AT2G14750
AT2G20610
AT2G22330
AT2G25450
AT2G32860
AT2G43100
AT3G03640
AT3G09260
AT3G19710
AT3G58990
AT4G12030
AT4G13770
AT4G30530
AT4G39940
AT4G39950
AT5G07690
AT5G14200
AT5G23010
AT5G25980
AT5G26000
AT5G61420

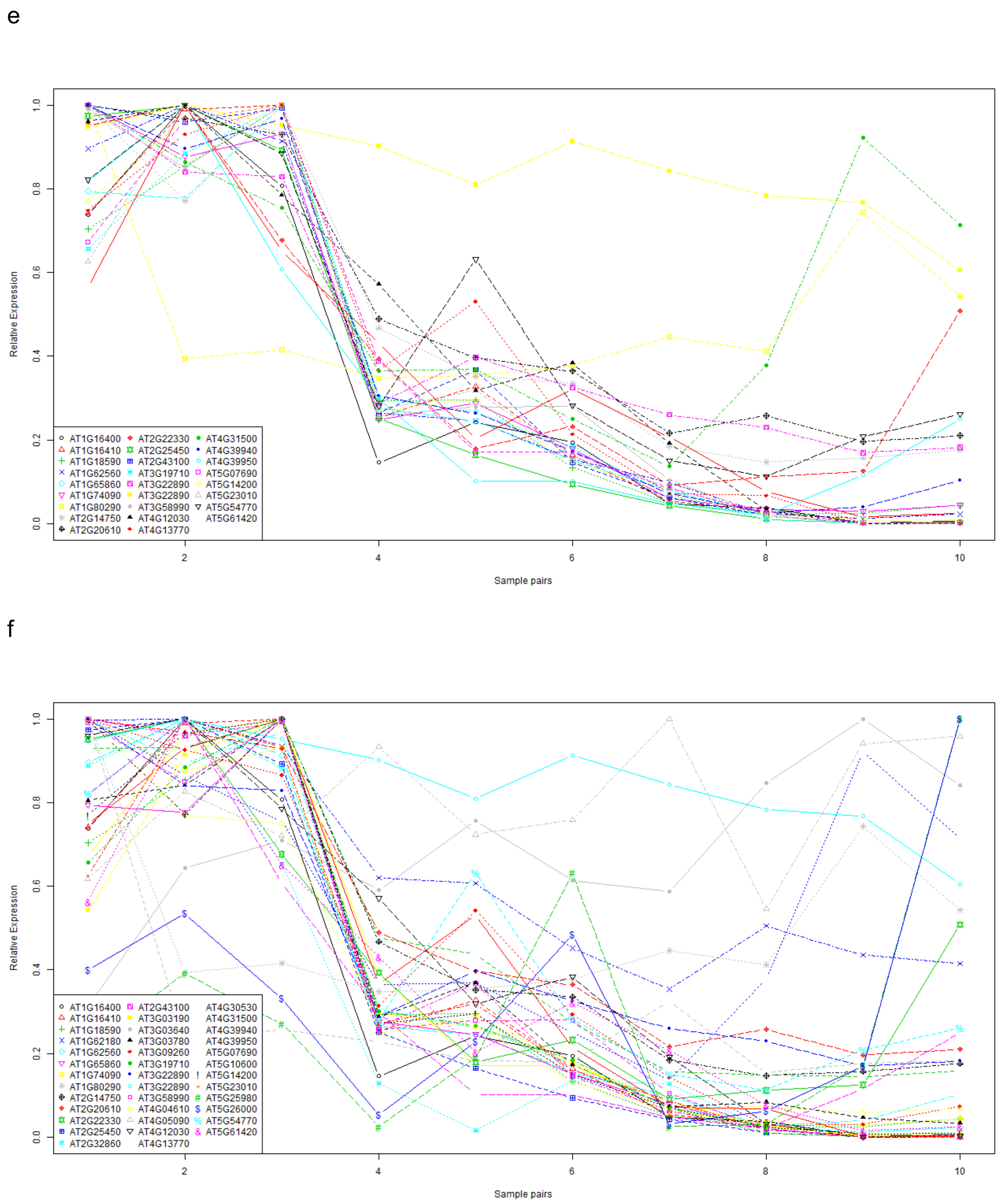


Figure 4.3: Expression profiles of common genes from Cuffdiff-DESeq-edgeR overlap. The above graphs show expression profiles of genes enriched in GluBP, GlyBP, GluMP, GlyMP, SCBP and SCMP. (a) to (f) shows expression profiles of gene clusters in “Against S7” sample pairs. Common genes were obtained by overlapping DEGs from Cuffdiff, DESeq and edgeR and expressed in more than one sample pair.

4.2.3 Identifying important regulators using PPI network analysis

Interactions between DEGs were studied for identifying the most prevalent interacting genes and their regulation on neighbouring genes. A protein-protein interaction (PPI) network was constructed for identifying highly connected genes and their most prevalent interactions. From PPI network analysis, 18 genes were found to have the highest interactions with edges ≥ 100 and thus be significantly involved in GluBP (Appendix A; Figure A.1). Apart from these 14 genes, 114 genes were involved in induced systemic resistance, sulphur compound biosynthetic process, cellular biogenic amine metabolic process, sulphur metabolism and biosynthesis, anion transport, organic acid transport and cellular response to external stimulus. Results show that most of the DEGs during the transition phase regulate other DEGs which provide induced resistance and protection against external factors such as stress, pathogens, herbivores, temperature variations, etc. A recent study on the relationship of glucosinolates to flowering in *A. thaliana* suggests that the presence of the *MAM1* gene affects glucosinolate accumulation and flowering time in the absence of *APOP2* and *APOP3* genes and leads to the production of C3 glucosinolates (Jensen *et al.*, 2015).

Results from the PPI network analysis clearly show that *MAM1* regulates several other genes in glucosinolates and displays a high expression profile correlation of 0.75 to *FLC* which supports the hypothesis of glucosinolate production and protection during flowering phase. Glucosinolates are sulphur and nitrogen-rich chemical compounds in plants that provide defense against pathogens and herbivores by forming a toxic compound upon herbivore attack when the cell wall is ruptured (Jensen *et al.*, 2015; Mohammadin *et al.*, 2017). Glucosinolates play a crucial role in flowering time regulation during transition from vegetative to reproductive phase and provide protection from herbivores and pathogens for the plant's vegetative and generative tissues during the transition phase. Therefore, differential expression of glucosinolates during the transition phase becomes essential.

4.2.4 Expression profiles of DE flowering genes

From CGenes, genes responsible for flowering and involved in regulation of flower development were identified. 5 genes were found to be involved in "Flowering". 18 were found to be associated with "Flower Development", 8 with "Regulation of Flower Development" and 3 with "Negative Regulation of Flower Development" (Figure 4.4). In "Against S7" sample pairs, many experimental genes such as *FLC*, *SOC1*, *EMS1* and *FD* have also been identified by enrichment analysis. Expression profiles of flowering genes shows that *SOC1*, *FCA*, *SAP* and

AGL31 increase in expression as compared to *FLC* which decreases in expression in “Against S7”. In the “Flower Development” process, a large cluster of genes in “Against S7” sample pairs display a “zig-zag” pattern of expression. There are four gene clusters observed in this process. The first cluster consists of *ATX1*, *RDR6*, *SOC1*, *KAN2*, *BPE*, *SRS2*, *FCA*, the expression values of which increase in S7-S9, decrease in S7-S11 and increase again in S7-S12. The second cluster consists of *ATX1*, *NAC054*, *NGA1* and *F-ATMBP* shows a decrease in expression followed by an increase in S7-S15 and S16. The third cluster consists of *EMS1*, *KAN2*, *ABCB19*, *SOC1* and *SAP1* shows a peak in expression value from S7-S14. The fourth cluster of genes consists of *SPT*, *SRS2*, *ATX1* and *FCA* in S7-S14 where the expression varies between 0.7 and 0.8. In the “Regulation of Flower Development” process, *POLA*, *FD*, *ATX1*, *SOC1*, *AGL31* and *FCA* show a decrease in expression in S7-S11 whereas *ATX1* shows an increase in expression in S7-S11. In the “Negative Regulation of Flower Development” process, only *FLC*, *AGL31* and *POLA* are expressed.

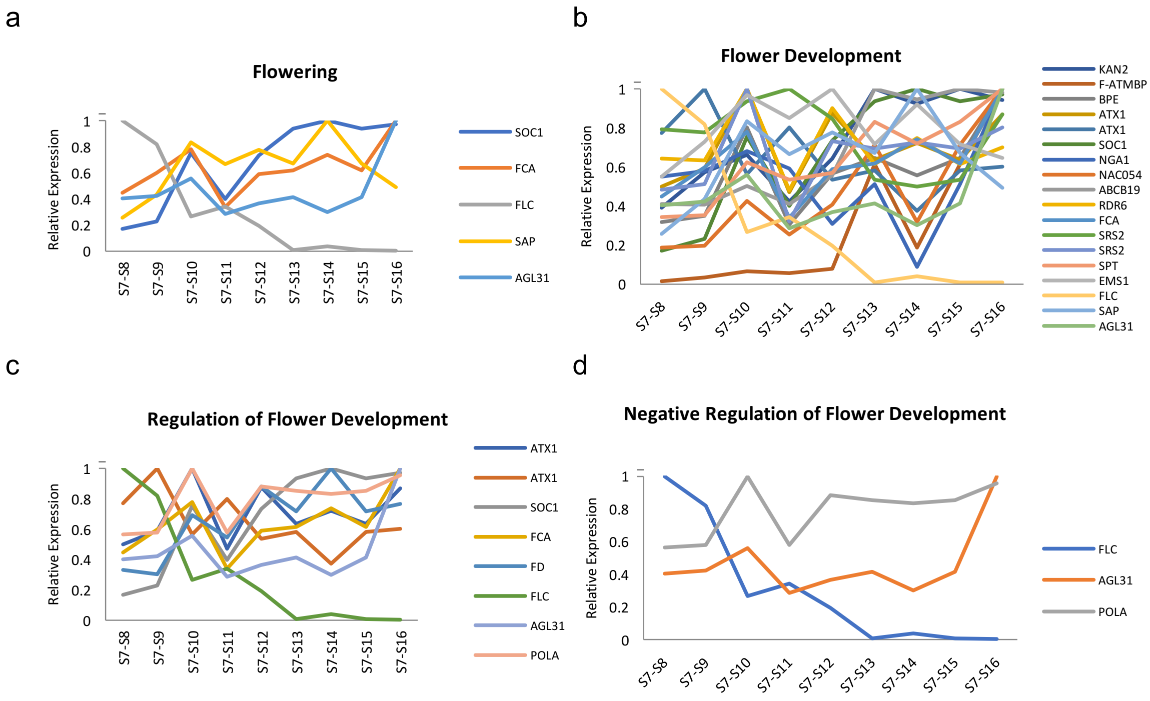
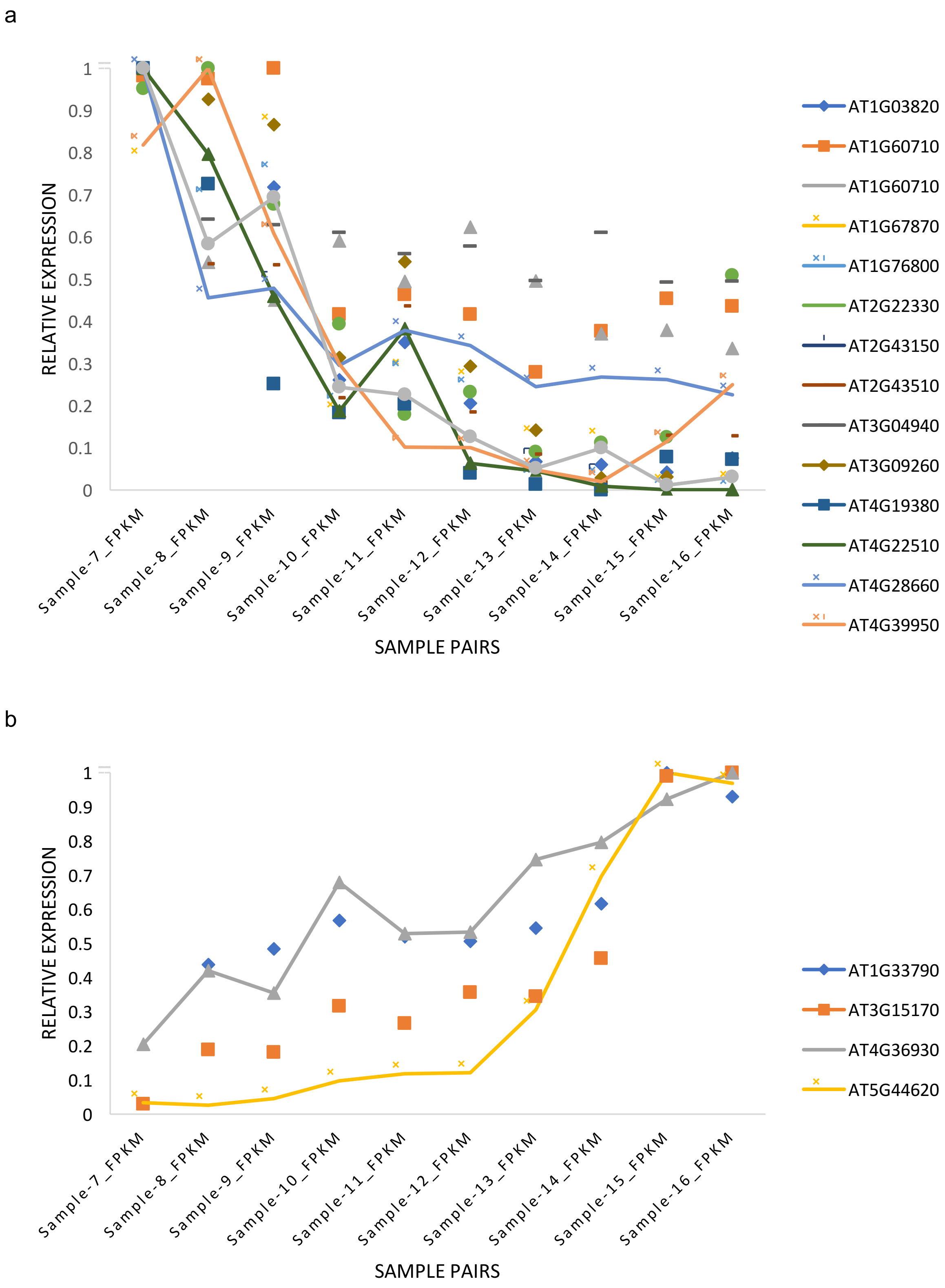


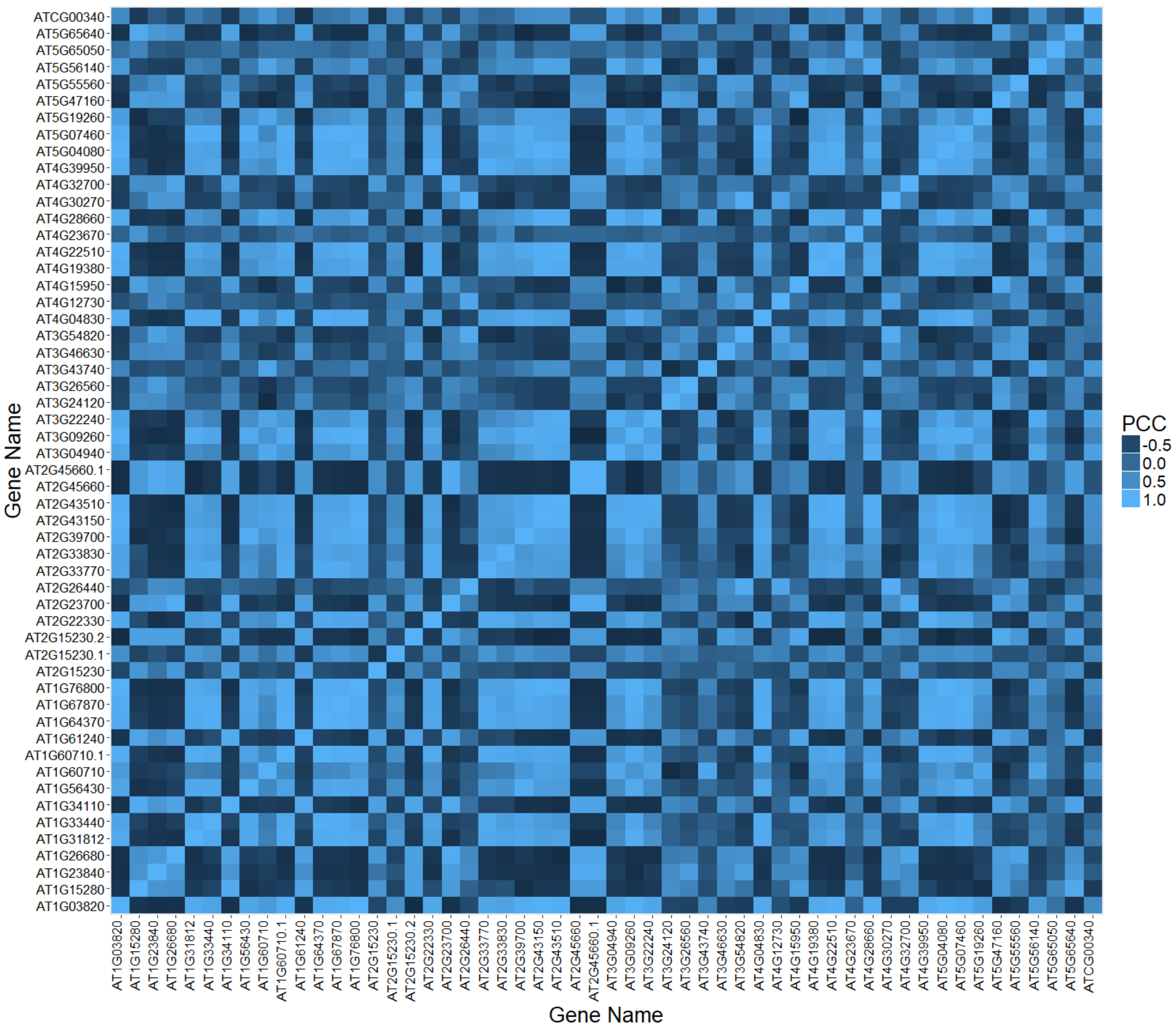
Figure 4.4: Expression profiles of flowering genes. The above figure illustrates relative expression profiles of genes involved in flowering, flower development, regulation of flower development and negative regulation of flower development. (a), (b), (c) and (d) shows relative expression of genes in “Against S7” sample pairs.

FPKM expression values of *FLC* and *LFY* genes from Day-1 to 10 were used to identify potential novel genes from the CGenes set by selecting those displaying the highest correlation with *FLC* and *LFY* expression profiles and having no ontology information for *A. thaliana*. Results of correlation and GO enrichment analysis showed that 69 and 7 genes which displayed the highest correlation (PCC≥0.9) in expression to *FLC* and *LFY* respectively did not get enriched in any biological or molecular function (Figure 4.5). 69 genes were found to be highly correlated by *FLC* out of which 14 genes were regulated and 55 genes were non-regulated. Similarly, for *LFY*, out of 7 genes 4 were regulated and 3 were non-regulated in the PPI network analysis. These genes were labeled as novel genes which can regulate the expression of other known floral regulators during the flowering transition phase. For identification of genes regulated by *FLC* or *LFY*, node connections were studied by filtering out genes connected with *FLC* or *LFY*.



a
RELATIVE EXPRESSION
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Sample-7 _FPKM
Sample-8 _FPKM
Sample-9 _FPKM
Sample-10 _FPKM
Sample-11 _FPKM
Sample-12 _FPKM
Sample-13 _FPKM
Sample-14 _FPKM
Sample-15 _FPKM
Sample-16 _FPKM
SAMPLE PAIRS
AT1G03820
AT1G60710
AT1G60710
AT1G67870
AT1G76800
AT2G22330
AT2G43150
AT2G43510
AT3G04940
AT3G09260
AT4G19380
AT4G22510
AT4G28660
AT4G39950
b
RELATIVE EXPRESSION
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Sample-7 _FPKM
Sample-8 _FPKM
Sample-9 _FPKM
Sample-10 _FPKM
Sample-11 _FPKM
Sample-12 _FPKM
Sample-13 _FPKM
Sample-14 _FPKM
Sample-15 _FPKM
Sample-16 _FPKM
SAMPLE PAIRS
AT1G33790
AT3G15170
AT4G36930
AT5G44620

c


ATCG00340
AT5G65640
AT5G65050
AT5G56140
AT5G55560
AT5G47160
AT5G19260
AT5G07460
AT5G04080
AT4G39950
AT4G32700
AT4G30270
AT4G28660
AT4G23670
AT4G22510
AT4G19380
AT4G15950
AT4G12730
AT4G04830
AT3G54820
AT3G46630
AT3G43740
AT3G26560
AT3G24120
AT3G22240
AT3G09260
AT3G04940
AT2G45660.1
AT2G45660
AT2G43510
AT2G43150
AT2G39700
AT2G33830
AT2G33770
AT2G26440
AT2G23700
AT2G22330
AT2G15230.2
AT2G15230.1
AT2G15230
AT1G76800
AT1G67870
AT1G64370
AT1G61240
AT1G60710.1
AT1G60710
AT1G56430
AT1G34110
AT1G33440
AT1G31812
AT1G26680
AT1G23840
AT1G15280
AT1G03820
Gene Name
AT1G03820
AT1G15280
AT1G23840
AT1G26680
AT1G31812
AT1G33440
AT1G34110
AT1G56430
AT1G60710
AT1G60710.1
AT1G61240
AT1G64370
AT1G67870
AT1G76800
AT2G15230
AT2G15230.1
AT2G15230.2
AT2G22330
AT2G23700
AT2G26440
AT2G33770
AT2G33830
AT2G39700
AT2G43150
AT2G43510
AT2G45660
AT2G45660.1
AT3G04940
AT3G09260
AT3G22240
AT3G24120
AT3G26560
AT3G43740
AT3G46630
AT3G54820
AT4G04830
AT4G12730
AT4G15950
AT4G19380
AT4G22510
AT4G23670
AT4G28660
AT4G30270
AT4G32700
AT4G39950
AT5G04080
AT5G07460
AT5G19260
AT5G47160
AT5G55560
AT5G56140
AT5G65050
AT5G65640
ATCG00340
Gene Name
PCC
-0.5
0.0
0.5
1.0

d

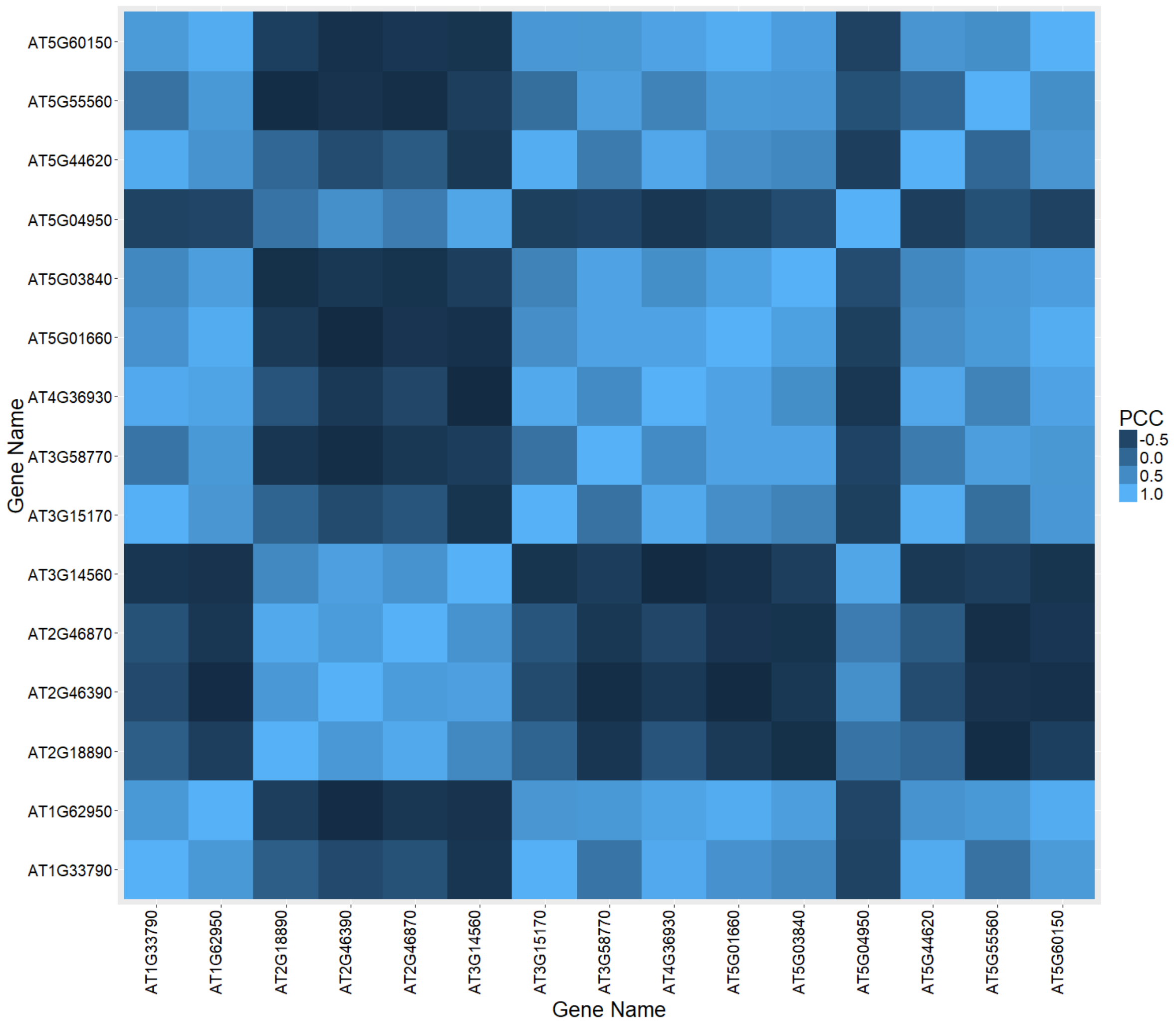


Figure 4.5: Expression profiles of common genes from Cuffdiff-DESeq-edgeR overlap showing correlation to *FLC* and *LFY* genes. (a) shows DEGs showing higher correlation to *FLC* and regulation by *FLC*, (b) shows DEGs showing higher correlation to *LFY* and regulation by *LFY,* (c) and (d) shows correlations of DEGs to *FLC* and *LFY* respectively.

4.2.5 Identification of lncRNA sequences

Using the TAIR10 annotation data in the *A. thaliana* reference dataset, 458 lncRNA sequences were identified. Using the iRF classifier, these lncRNA sequences were used as a test set to observe the prediction performance of the sequence features and classification accuracy. Using all 73 features, iRF successfully identified 283 lncRNA sequences with prediction accuracy of 79.18% with AUC of 91.96. Sensitivity of 98.94% shows that the percentage of correctly identified protein-coding sequences are much higher than the correctly identifying lncRNA sequences. The test generated specificity of 99.37% with no lncRNA sequence predicted as “protein-coding”. Approximately 60% of the lncRNA sequences were correctly identified as “lncRNA” and remaining 40% were identified as “protein-coding”. The classification produced an F1-score of 73.91 with PPV of 98.94 and NPV of 70.79. The test generated an MCC value of 0.638 which is comparatively closer to perfect prediction value of +1.

Feature selection using LiRF-FS was applied to obtain a list of optimal features producing similar prediction accuracy. With $\lambda_{lower} = 10^{-5}$, $\lambda_{upper} = 0.1$, $\lambda_{step-size} = 10^{-5}$ and $tolerance = 0.3$; 57 features were selected having prediction accuracy of 79.49% with sensitivity of 98.95%, specificity of 99.37%, F1-score of 74.41, PPV of 98.95, NPV of 71.10 and MCC of 0.643. The tolerance values parameter controls the selection of features. Using a given tolerance value, the accuracy of the features is compared to the feature set producing maximum prediction accuracy ($\mathrm{maxPredAcc}^{\mathrm{featureSet}}$). The feature set having prediction accuracy within the $\mathrm{differenceValue}$ is selected based on the following condition:

$$\mathrm{maxPredAcc}^{\mathrm{featureSet}} - \mathrm{PredAcc}^{\mathrm{featureSet}} = \mathrm{differenceValue}, \tag{4.1}$$

where $\mathrm{differenceValue} \leq \mathrm{tolerance}$.

The feature set having accuracy difference value below the tolerance value is selected to contain optimal feature set. From the analysis, 57 features were selected which identified 286 lncRNA sequences. These 57 features are *Hexamer Score*, *ORF Length*, *Mean ORF coverage*, *ORF coverage*, *Transcript Length*, *GC content*, *Fickett Score*, *Fop*, *CUB*, *RCB*, *EW*, *SCUO*, *TTT*^{RSCU,} *TTC*^{RSCU}, *TTA*^{RSCU}, *TTG*^{RSCU}, *CTT*^{RSCU}, *CTC*^{RSCU}, *CTG*^{RSCU}, *ATC*^{RSCU}, *ATA*^{RSCU}, *ATG*^{RSCU}, *GTC*^{RSCU}, *GTA*^{RSCU}, *GTG*^{RSCU}, *TCT*^{RSCU}, *TCC*^{RSCU}, *TCA*^{RSC}, *TCG*^{RSCU}, *CCG*^{RSCU}, *ACA*^{RSCU}, *ACG*^{RSCU}, *GCT*^{RSCU}, *GCC*^{RSCU}, *GCA*^{RSCU}, *GCG*^{RSCU}, *TAT*^{RSCU}, *TAC*^{RSCU}, *CAT*^{RSCU}, *CAC*^{RSCU}, *CAA*^{RSCU}, *AAC*^{RSCU}, *AAA*^{RSCU}, *AAG*^{RSCU}, *GAC*^{RSCU}, *GAG*^{RSCU}, *TGC*^{RSCU}, *TGG*^{RSCU}, *CGT*^{RSCU,} *CGA*^{RSCU}, *AGT*^{RSCU}, *AGC*^{RSCU}, *AGA*^{RSCU}, *AGG*^{RSCU}, *GGT*^{RSCU}, *GGC*^{RSCU} and *GGG*^{RSCU}.

The performance of the lncRNA identification using iRF, RF and SVM classifiers were compared and evaluation metrics were calculated for each. Using 73 features on 478 lncRNA sequences as the negative test set and 478 protein-coding sequences as the positive test set (Table 4.2); iRF identified predicted lncRNAs with 74.2% accuracy whereas RF predicted with 77.46% accuracy. SVM on the other hand, gave prediction accuracy of 70.07% with difference of 4.13% against iRF and 7.39% against RF. Sensitivity as well as specificity produced by RF was comparatively higher than iRF classifier. Sensitivity by iRF was comparatively lower with SENS value of 73.98. Compared to iRF, SVM obtained lower sensitivity of 69.48% with difference of 7.61 and 3.11 from RF sensitivity prediction. Additionally, higher F1, PPV, NPV and MCC values were detected with RF classification.

Performance comparison with 57F, 31F and 7F feature sets were compared with 73F using iRF (Table 4.2), RF and SVM classifiers. Both the 31F and 7F displayed higher accuracies of 75.12% and 75.87% respectively with iRF. Whereas 31F showed greater accuracy value of 77.02% using RF when compared against 57F and 7F. 7F showed lower accuracy of 75% with RF. SVM showed the lowest prediction performance of 69.83% for 57F, 70.11% for 31F and 66.89% for 7F feature sets. Performance of iRF and RF in 31F showed comparatively higher values in all the metrics. Both SENS and SPEC indicates that RF produced a greater chance of obtaining true negatives and true positives from a given dataset of FASTA sequences. Additionally, the results also display significantly higher performance against SVM in obtaining true positives and negatives. For the subsequent analysis and validation on test set sequences, 31F feature set was selected as an optimal feature set.

Table 4.2: Prediction performance of iRF, RF and SVM classifiers with 73F, 57F, 31F and 7F feature sets on *A. thaliana* apical shoot test set data.

Feature set	**Classifier**	**ACC**	**SENS**	**SPEC**	**F1**	**PPV**	**NPV**	**MCC**
73F	iRF	74.20	73.98	75.62	73.34	73.98	74.40	0.48
73F	RF	77.46	77.09	77.85	77.46	78.49	76.41	0.55
73F	SVM	70.07	69.48	70.68	70.07	71.31	68.83	0.40
57F	iRF	74.16	74.04	75.77	73.25	74/04	74.27	0.48
57F	RF	76.67	76.55	76.79	76.67	77.57	75.74	0.53
57F	SVM	69.83	69.95	69.70	69.83	70.77	68.85	0.39
31F	iRF	75.12	74.59	75.85	74.47	74.59	75.62	0.50
31F	RF	77.02	75.38	78.74	77.02	78.81	75.31	0.54
31F	SVM	70.11	71.81	68.32	70.10	70.39	69.80	0.40
7F	iRF	75.87	74.90	75.69	75.47	74.90	76.83	0.51
7F	RF	75	74.92	75.08	75	75.92	74.05	0.50
7F	SVM	66.89	75.15	58.22	66.63	65.36	69.08	0.34

4.2.6 Performance benchmarking results on TAIR10 lncRNA sequences against CPC Tools

Performance of the framework with 478 lncRNA sequences was benchmarked against state-of-the-art coding potential tools: PLEK (Li, Zhang and Zhou, 2014), lncScore (Zhao, Song and Wang, 2016), CPAT (Wang *et al.*, 2013), and CPC2 (Kang *et al.*, 2017). A 10-Fold Cross Validation (CV) accuracy benchmarking was performed on non-randomized (D1) and randomized (D2) datasets. Results from accuracy benchmarking on *A. thaliana* D1 data (Figure 4.6a) show that the framework achieved an average accuracy of 74.95%, whereas PLEK, CPAT, lncScore and CPC2 achieved a mean accuracy of 63.02%, 52.55%, 68.22% and 51.05%, respectively.

On fold-2, the framework achieves the highest accuracy of 79.27%. CPAT produced the lowest accuracy of 48.08% followed by CPC2 producing accuracy of 51.61% whereas PLEK produced 63.78% and lncScore produced 67.2%. The framework generated the lowest accuracy of 70.42% on fold-2 among the accuracies produced in the 10 folds. However, the accuracy generated by the framework is comparatively higher than PLEK, CPAT, lncScore and CPC2 by differences of 10.77%, 19.62%, 3.83% and 22.84% for PLEK, CPAT, lncScore and CPC2, respectively. In the rest of the folds from fold-3 to fold-10, the accuracy of the framework was between 74.09% to 75.75%.

While lncScore produces the highest accuracy of 71.28% in fold-3, the prediction accuracy decreases in subsequent folds, which fluctuates between 69.57% to 64.48%; with fold-10 being the lowest. PLEK shows a similar pattern as produced by lncScore, however the accuracy values are lower than lncScore. CPAT did not showed any variation in the prediction values and generated a similar accuracy value from fold-2 to fold-10. The prediction values produced by CPC2 ranged between 46.88% to 53.63% from folds-1 to 9. Comparatively higher prediction accuracy of 55.53% was detected in fold-10.

A second 10-Fold CV benchmarking was performed (Figure 4.6b) by randomizing and mixing the lncRNA and protein-coding sequences to perform a fair comparison of the performances of various tools. The framework produces a mean accuracy of 78.2% whereas PLEK, CPAT and lncScore produce mean accuracies of 63.6%, 55.5% and 68.98%, respectively. With the D2 data, the performance of the framework and the tools display non-variable consistent accuracies. Among all the tools, the framework generated the highest prediction accuracies with accuracies ranging between 76.66% and 80.22%. CPAT accuracies ranged between 55.05% and 56.46%, PLEK ranged between 61.3% and 65.37%, whereas lncScore ranged between 66.41% and 70.6%. CPC2 produced a mean accuracy of 49.75% from all the folds with accuracy fluctuating between 48.05% and 52.37%.

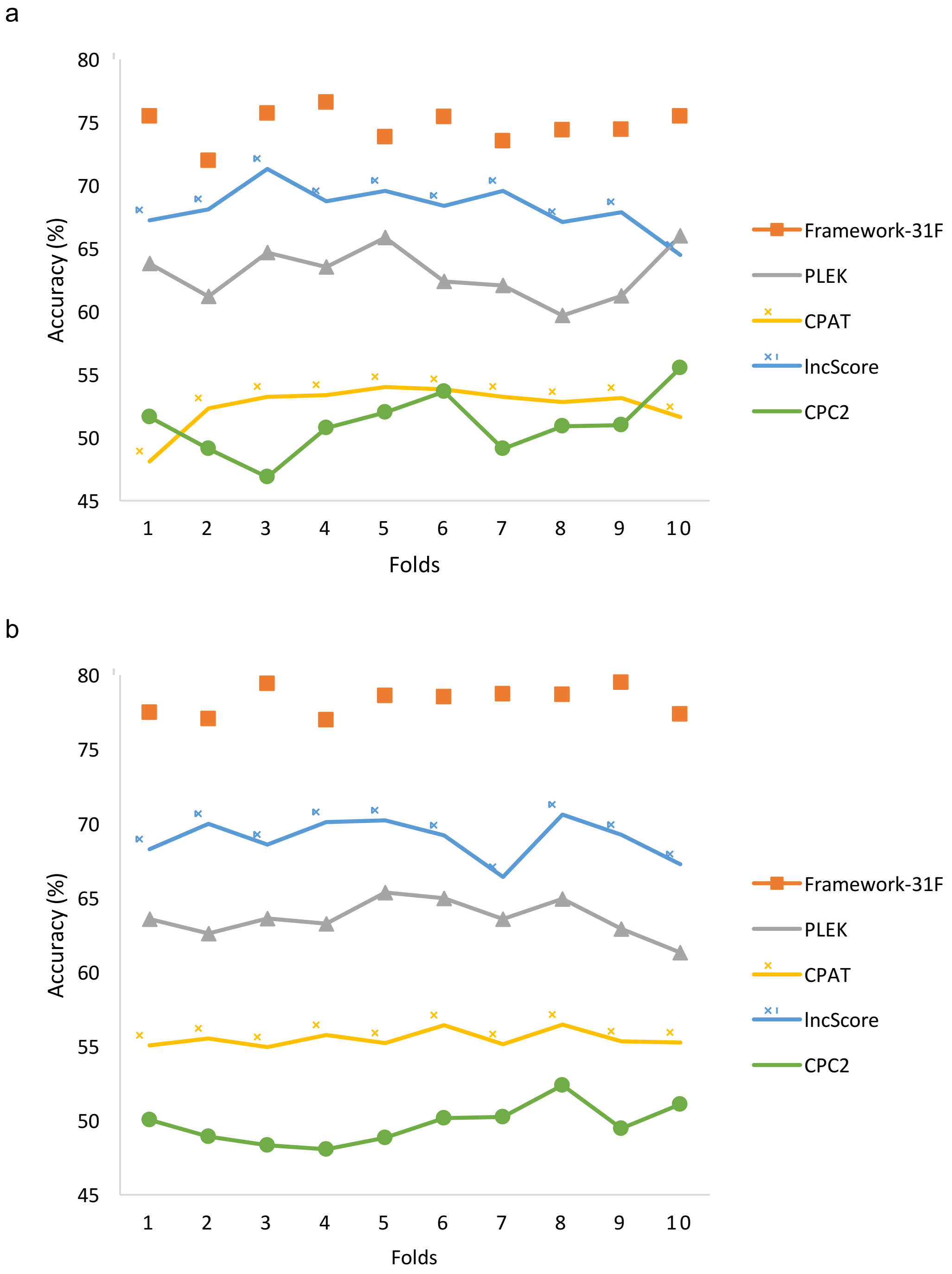


Figure 4.6: Performance benchmarking of the framework with known CPC tools on *A. thaliana* TAIR10 annotated (a) D1 non-shuffled dataset, and (b) D2 shuffled dataset.

The differences in the individual accuracies at each fold was computed in the D1 and the D2 datasets (Figure 4.7). Results from the 10-Fold CV benchmarking on D1 dataset (Figure 4.7a) clearly shows that the framework produces highest difference of 28.82% with the CPC2 on fold-3 followed by a mean difference of 23.6%. A significant difference of 27.37% was observed with CPAT on fold-1 with mean difference of 21.51% on all the folds. Difference with PLEK showed a mean of 11.62% whereas with lncScore showed difference of 6.42%. The difference with lncScore was comparatively higher on fold-1 and fold-10 with values of 8.25% and 10.97%, respectively.

Comparison of the results on the D2 dataset (Figure 4.7b) shows consistent difference in all the folds. The mean differences between the accuracies produced by the framework and other CPC tools were 14.6%, 22.7%, 9.2% and 28.47% for PLEK, CPAT, lncScore and CPC2, respectively.

Results from the 10-Fold CV shows that the framework performed comparatively better on both the non-randomized and the randomized datasets with superior performance in the prediction accuracies. Results from the other CPC tools fail to predict the lncRNA sequences generating lower prediction performance in the prediction performance tests. The 10-Fold CV test is performed to benchmark and test the robustness of a tool against a variable dataset. Results clearly show that the framework performed accurately on all the folds with consistently similar accuracies on both datasets which clearly validates the robustness of the framework.

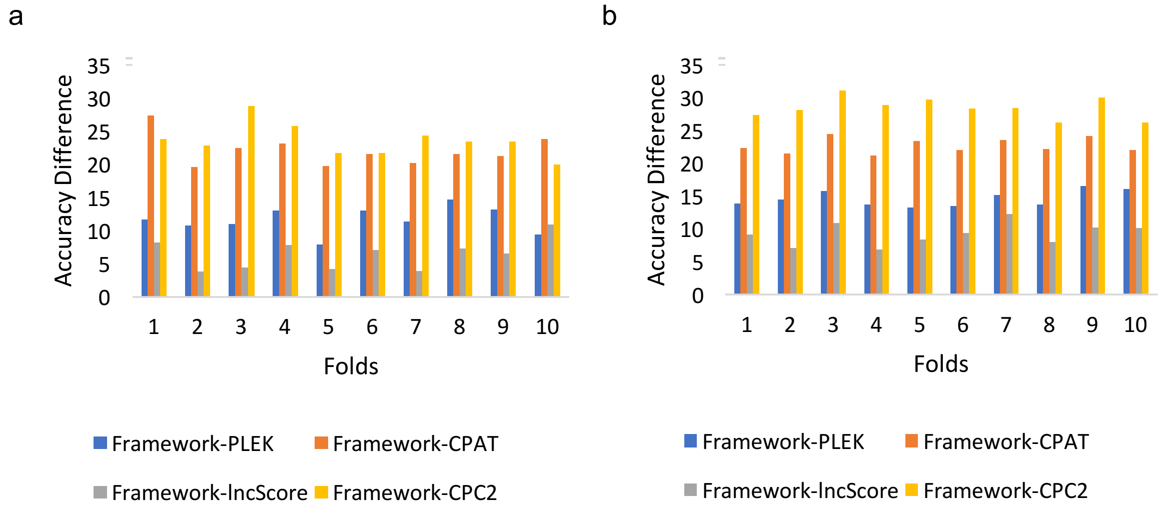


Figure 4.7: Bar chart showing accuracy differences of the framework against CPC tools with *A. thaliana* TAIR10 (a) D1, and (b) D2 datasets.

4.2.7 Performance benchmarking results on A. thaliana EST lncRNA sequences

To further evaluate the robustness of the framework in identifying the lncRNA sequences, optimal features obtained from feature selection from 6-plants dataset are examined. 31 features were obtained with $tolerance = 0.3$, $\lambda_{lower} = 10^{-5}$, $\lambda_{upper} = 0.1$ and $\lambda_{step-size} = 10^{-5}$. The performance of the model with 31 features was benchmarked against the sequences derived from PLncDB database (Jin *et al.*, 2013). Apart from the lncRNA sequences annotated in TAIR10 database, the PLncDB database also contains lncRNA sequences obtained from the Expressed Sequence Tags (EST) analysis. From the EST analysis, 4828 lncRNA sequences in *A. thaliana* have been identified. For performance benchmarking, a random set of 2000 lncRNA sequences were extracted and the accuracy was compared against the popular CPC tools as performed in the Section 6.2.6.

A density distribution comparison of transcript lengths in TAIR10 lncRNA sequences and EST-derived sequences (Figure 4.8) demonstrates that sequences derived from TAIR10 range from 204 bp to 7697 bp with represented by log(transcript length) where majority of the sequences having transcript length between 200-1000 bp. The sequences are clustered with log(transcript length) ranging between 5–9. Sequences from EST-derived data, however, show a different density distribution pattern with the majority of the sequences ranging between 200-10000 bp. The density distribution of sequences demonstrates large amount of sequences lying between 5 and 9.5 forming a long tail where sequences are clustered between 11–13. Sequences forming the long tail comprises large proportion of sequences with transcript lengths above 10000 bp with sequence length ranging between 1×10^5 to 7.8×10^5 derived from PLncDB database. However, the test set used for benchmarking contains 148 sequences having sequence lengths greater than 50000 bp. Such extremely long lncRNA sequences are generally mis-classified as protein-coding transcripts, due to which the overall prediction accuracy decreases. To evaluate and measure the performance of the framework, a comparative analysis was conducted using these extremely longer transcript sequences in the training and test set with the set of maximal optimal features obtained from LiRF-FS approach.

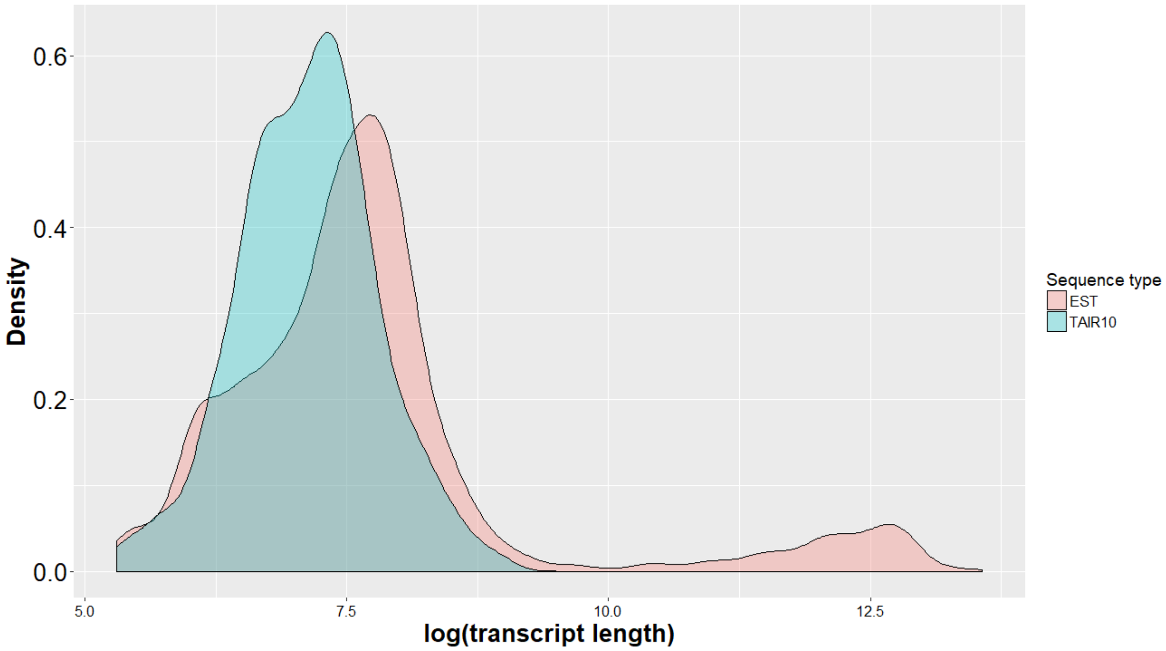


Figure 4.8: Density distribution of transcript lengths of lncRNA sequences in *A. thaliana* TAIR10-annotated and EST-predicted results. X-axis is log of transcript lengths and y-axis is density.

Figure 4.9 shows the 10-Fold CV performance of the framework against CPC2, PLEK, CPAT and lncScore tools on non-shuffled and shuffled datasets. 10080 sequences were used as the training set and 1120 sequences were used as the test set in each fold. Results from the non-shuffled dataset Figure 4.10a exhibits significant differences in the accuracies obtained from the tools. In each fold, the framework displays remarkable performance than other tools. Where the framework produces an accuracy of 72.59%, PLEK and lncScore generates similar accuracies of 63.75% and 63.66%, respectively. As the fold increases, accuracy of the framework increases to 75.44% on Fold-3 whereas accuracy for PLEK decreases to 62.41%. lncScore exhibits a slight increase of 0.62% with accuracy to 64.28% in Fold-3.

Accuracies of the framework in folds-6, 7 and 8 varies between 73.79% to 75.35% whereas PLEK generates a further decrease in the accuracies with stable values in the range of 60.94% ± 0.07%. lncScore also exhibits stable accuracy values however, a steady decrease can be observed where the accuracy decreases from 65.65% to 64.96%. Among all the tools, CPAT and CPC2 exhibited worst prediction performance with average accuracies of 50.76% and 50.57% respectively. A notable difference between the tools can be observed in fold-9 and 10. These folds consist of lncRNA sequences from EST analysis. When EST-derived lncRNA sequences are used as test set sequences, the framework identifies these sequences with 68.48% and 68.69% accuracy in fold-9 and fold-10, respectively. However, when the same sequences are tested against other CPC tools, the prediction performance decreases to 49.91% and 52.01% for lncScore; and, 52.15% and 54.33% for PLEK in fold-9 and fold-10, respectively. CPAT however, exhibits an increase in the prediction accuracy from 50% to 53.66% and 53.44% in the last two folds. CPC2 on the other hand, showed increase in the accuracy in fold-9 whereas a decrease from 52.14% to 49.33% was observed in fold-10 (Figure 4.9a).

The results from 10-fold CV performance benchmarking on the shuffled dataset can be observed in Figure 4.10b. In this analysis, the EST-derived lncRNA sequences have been shuffled to detect the performance of the tools in this scenario. Results indicate that the framework produces highest prediction accuracies among other tools an all the folds. By shuffling the lncRNA sequences, it can be clearly observed that the difference between the accuracies becomes much higher than the one observed in the D1 dataset. The graph demonstrates that, when lncRNA sequences are shuffled, the accuracies do not vary significantly. The framework produces an average accuracy of 76.03% whereas PLEK generates an average of 61.72%. lncScore on the other hand, shows an average of 62.75%, CPAT and CPC2 displayed an average of 53.57% and 49.45%.

Slight differences in the performance variation can be observed in folds-3, 4, 6 and 7 where lncScore and PLEK show contrasting accuracies when compared to previous folds. It is interesting to note that as the prediction accuracy for lncScore increases, the accuracy for PLEK decreases, and vice versa. These differences can be mainly attributed to the features extracted in the two CPC tools. The graph also shows that the framework produces pattern similar to lncScore in folds-5 to 10. Although the pattern is similar, the accuracy values produced by the framework are much higher than lncScore.

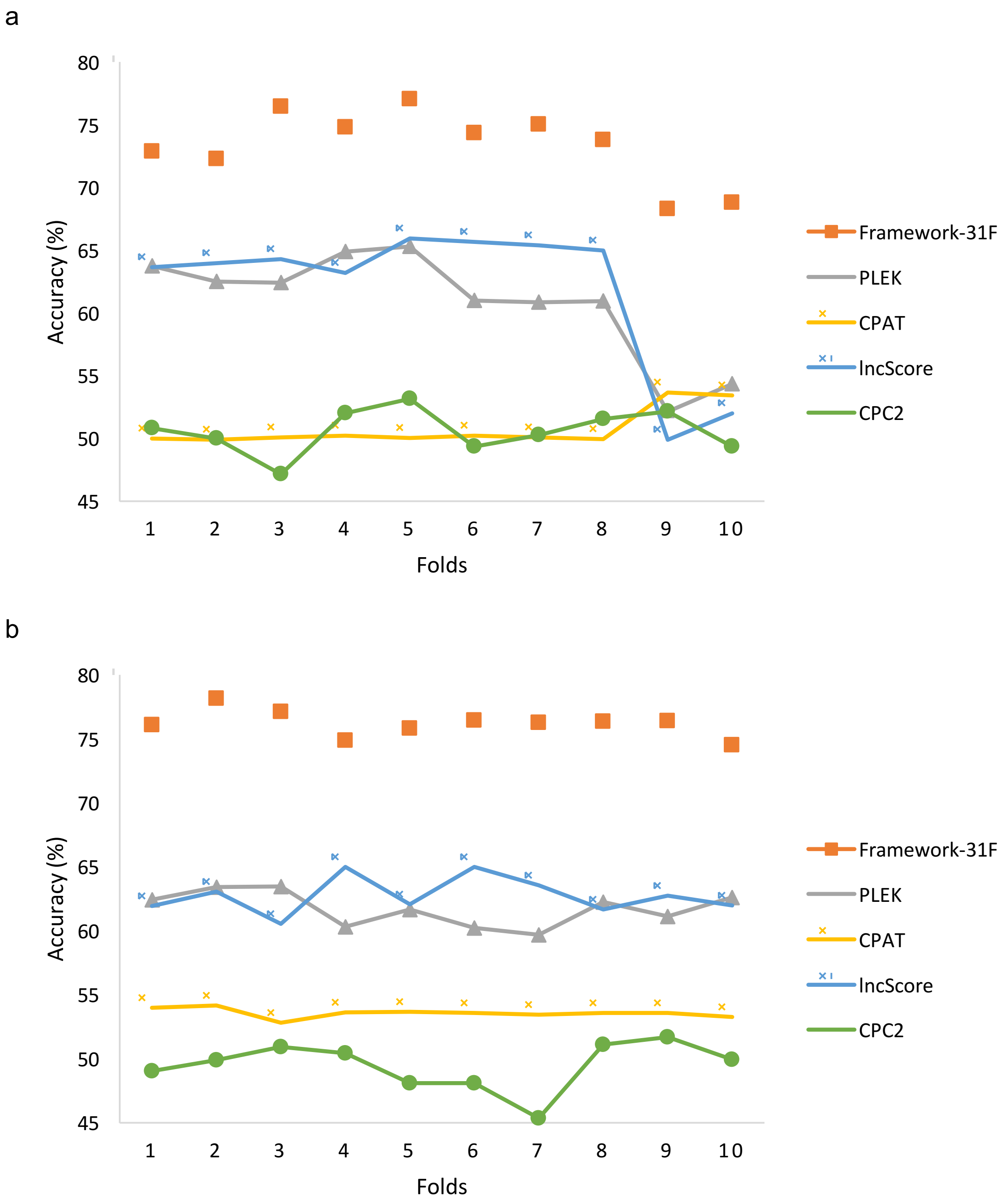


Figure 4.9: Performance benchmarking of the framework using 10-Fold CV with 31 features against CPC tools on *A. thaliana* EST annotated (a) D1 non-shuffled dataset, and (b) D2 shuffled dataset.

The differences between the prediction accuracies can be observed by plotting a histogram of the difference between the accuracies obtained by the framework against other CPC accuracy values in subsequent folds (Figure 4.10). In the D1 dataset (Figure 4.10a), the highest difference between the accuracies can be observed between the framework and CPC2 with average difference of 22.78% followed by CPAT with a mean difference of 22.59%. PLEK and lncScore generated differences of 12.54% and 11.46%, respectively. Whereas, in the D2 dataset (Figure 4.10b), the average difference between framework and CPAT remains similar when compared to D1 dataset producing 22.63%. Difference between the framework and CPC2 increased to 26.75% in D2 dataset. PLEK and lncScore, however, show an increase in the difference by 1.81% for PLEK and 1.86% for lncScore.

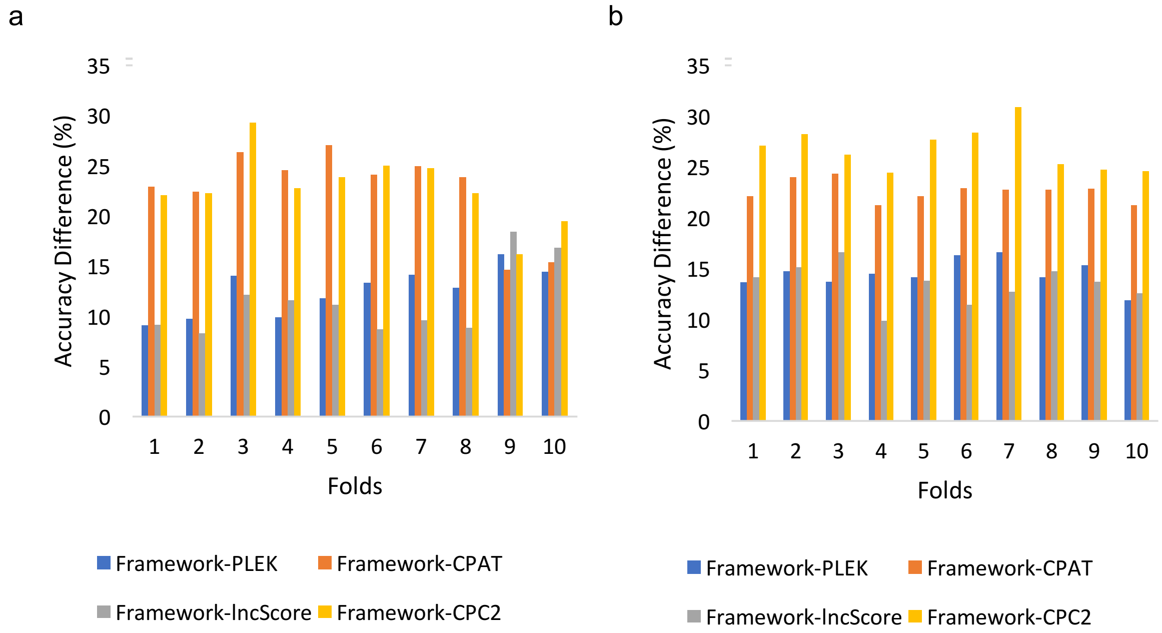
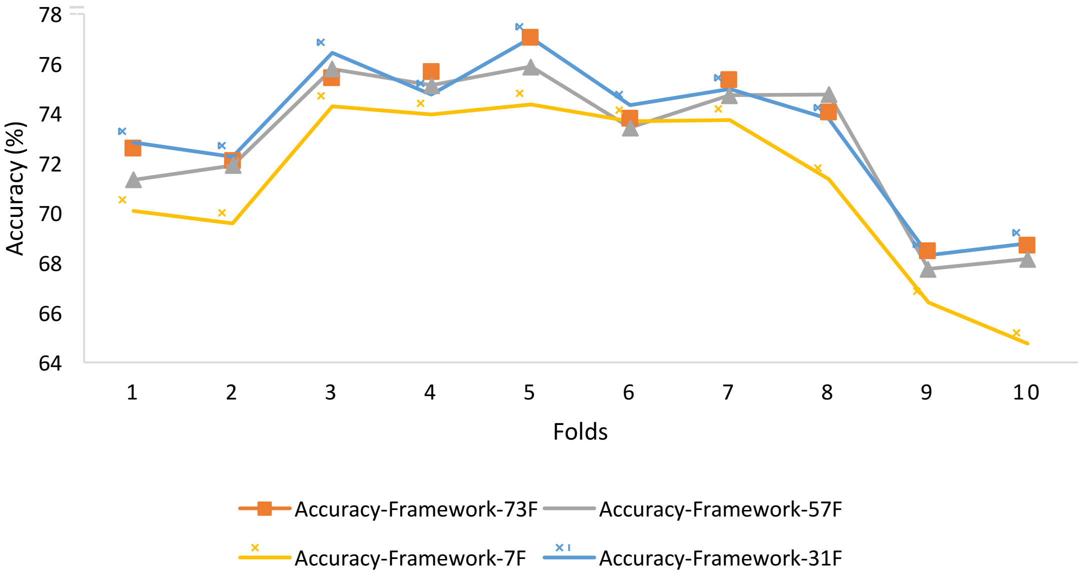


Figure 4.10: Bar chart showing accuracy differences of the framework against CPC tools with *A. thaliana* EST (a) D1, and (b) D2 datasets.

To evaluate the efficiency of the features produced by LiRF-FS approach, prediction accuracy benchmarking was performed between the feature sets selected from LiRF-FS approach. Evaluation of the prediction accuracy with 7F, 31F and 57F feature sets obtained from selection of minimal and maximal optimal features were compared against 73F feature set (Figure 4.11). Evaluation of the D1 dataset (Figure 4.11a) shows that the accuracies obtained from 73F, 57F and 31F do not display similar profiles. However, the accuracy profile generated by 7F exhibits significant differences. For each fold, the accuracy obtained from 7F has consistent difference of ~2% when compared to 73F except folds-6 and 10.

Accuracies obtained from the shuffled D2 dataset (Figure 4.11b) display a dissimilar pattern as observed in D1 dataset. Results show that accuracies from 31F display comparatively higher accuracies in folds-2, 3, 5, 8 and 10. In fold-1, 73F produces 77.14% whereas 31F produces 76.07%. In fold-2, the prediction accuracy from 31F increases to 78.17% whereas 73F generates 76.74% for the same fold having a difference of 1.43%. In fold-3, 31F shows a higher prediction performance among the rest of the selected feature sets displaying an accuracy of 77.14%. Next significant difference can be observed in fold-5 where 31F shows an accuracy of 75.8% whereas 73F generated accuracy of 74.73%. In fold-5, accuracy from 7F produced a difference of 1.88% with 73F and 2.95% with 31F. In the subsequent folds, 31F accuracy does not vary appreciably; however, it does exhibit increased accuracies in folds-8 and 10.

a



b

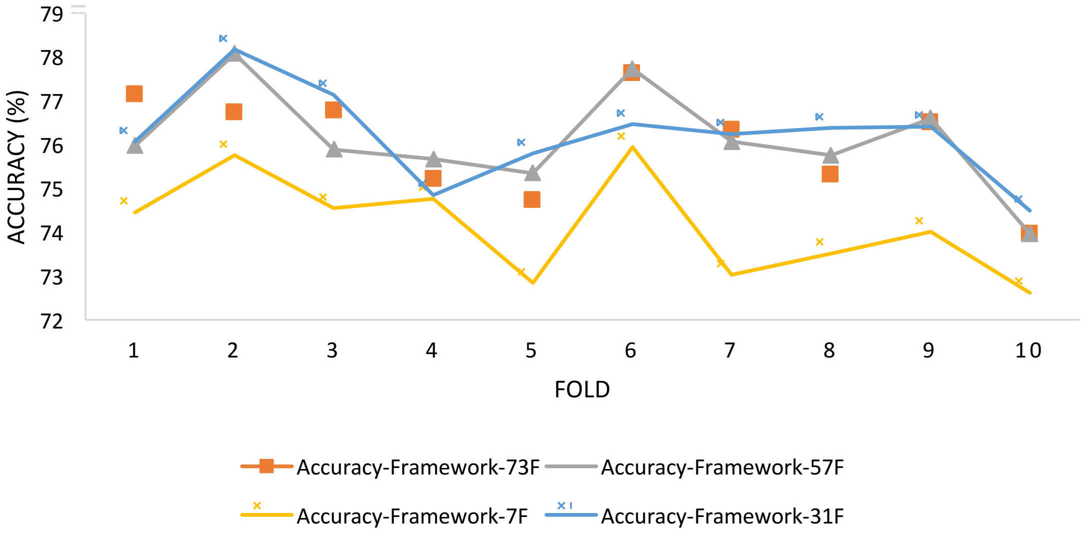


Figure 4.11: Performance comparison of selected features from LiRF-FS on *A. thaliana* (a) EST D1 non-randomized dataset, and (b) EST D1 randomized dataset.

Results from feature selection on ATH and 6-plants datasets resulted in three feature sets:

1. 7 optimal features obtained from LiRF-FS on the 6-plants dataset. (7F)
2. 31 optimal features obtained from LiRF-FS on the 6-plants dataset. (31F)
3. 57 optimal features obtained from LiRF-FS on the ATH apical-shoot TAIR10 dataset. (57F)

The 7F feature set was obtained by selecting the least number of optimal features having the $\mathrm{differenceValue}$ (Equation 2.23) less than 0.3 in the negative direction from the maximum accuracy λ value. The 31F feature set was obtained by selecting the maximum number of features having the $\mathrm{differenceValue}$ less than 0.3 in the positive direction. The negative direction implies that least number of features are obtained having a prediction accuracy within the threshold value, and the positive direction implies that maximum features are obtained having prediction accuracy within threshold value. As described in Table 2.7 (Algorithm 3), the minimal optimal features are obtained as having number of features less than the feature set producing highest prediction accuracy ($\mathrm{maxPredAcc}^{\mathrm{featureSet}}$), whereas the maximum optimal features are obtained having number of features greater than $\mathrm{maxPredAcc}^{\mathrm{featureSet}}$. 57 features were obtained from the feature selection performed on the TAIR10 dataset which was tested on EST-derived sequences.

Results indicate that the 31 features selected from LiRF-FS produces similar prediction values as observed by 73 features. Accuracies obtained from 31 features displayed a tradeoff between the accuracy values obtained from 73F and those obtained from 57F. Accuracy values from 31F exhibits higher accuracies than 73F or 57F. Accuracies from 7F display lower values when compared to 73F, 57F and 31F feature sets and therefore should not be considered, as these values lead to under-fitting of the model. Usage of 57F feature set has been derived from the feature selection of TAIR10 lncRNA data and therefore, using the selected features can lead to over-fitting of the model. The 31F feature set produces similar accuracies as those detected by 73F feature set. The accuracy values do not deviate significantly and also displays greater performance in some folds.

The 31F feature set consists of 7 sequence-based features: *Hexamer Score*, *ORF Length*, *Mean ORF Coverage*, *ORF Coverage*, *Transcript Length*, *GC content* and *Fickett Score*; 5 codon-bias features: *Fop*, *CUB*, *RCB*, *EW* and *SCUO*; and, 19 codon-bias RSCU features: ATT^{RSCU}, ATC^{RSCU}, ATA^{RSCU}, TCC^{RSCU}, CCA^{RSCU}, ACC^{RSCU}, ACA^{RSCU}, GCT^{RSCU}, GCA^{RSCU}, GCG^{RSCU}, CAT^{RSCU}, CAC^{RSCU}, AAC^{RSCU}, AAA^{RSCU}, AAG^{RSCU}, TGC^{RSCU}, CGT^{RSCU}, AGG^{RSCU} and GGA^{RSCU}. Since the majority of the selected features are RSCU features, this indicates the significance of the synonymous codon usage in distinguishing the lncRNA and protein-coding sequences.

4.2.8 Repeated K-Fold Cross-Validation analysis of A. thaliana lncRNA transcripts

In order to evaluate the model and avoid overfitting of the data, a 10-Fold repeated CV was performed to evaluate the robustness of the framework and the optimal features selected from LiRF-FS analysis. A repeated CV is performed in order to avoid overfitting during model training. A 10-Fold CV was performed 50 times with repeated shuffling of the sequences on each repetition. The performance of the prediction accuracy, precision and recall of the 31F optimal feature set was evaluated by repeated K-Fold CV on TAIR10 and EST-annotated datasets. The accuracy values vary between 77.1% and 78% on TAIR10 dataset (Figure 4.12), whereas precision and recall values displayed identical profiles varying between 78% and 78.7%.

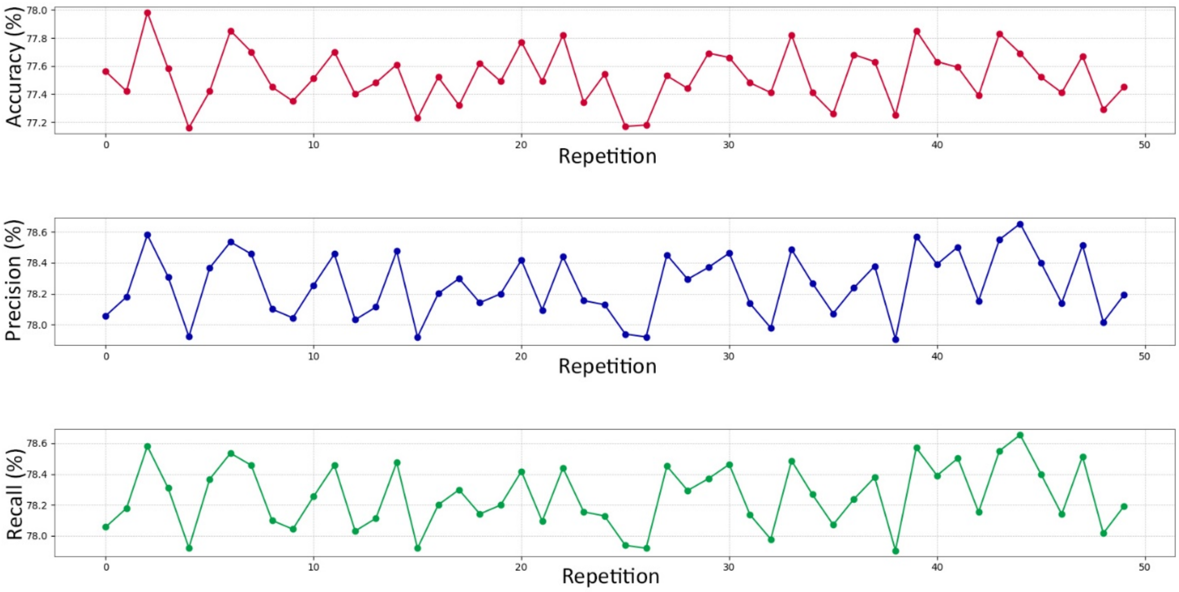


Figure 4.12: Repeated 10-Fold CV plots of Accuracy, Precision and Recall on *A. thaliana* TAIR10 dataset.

The prediction accuracy, precision and recall values on *A. thaliana* EST-annotated dataset (Figure 4.13) exhibited similar profiles with values fluctuating between 74.7% and 75.5%.

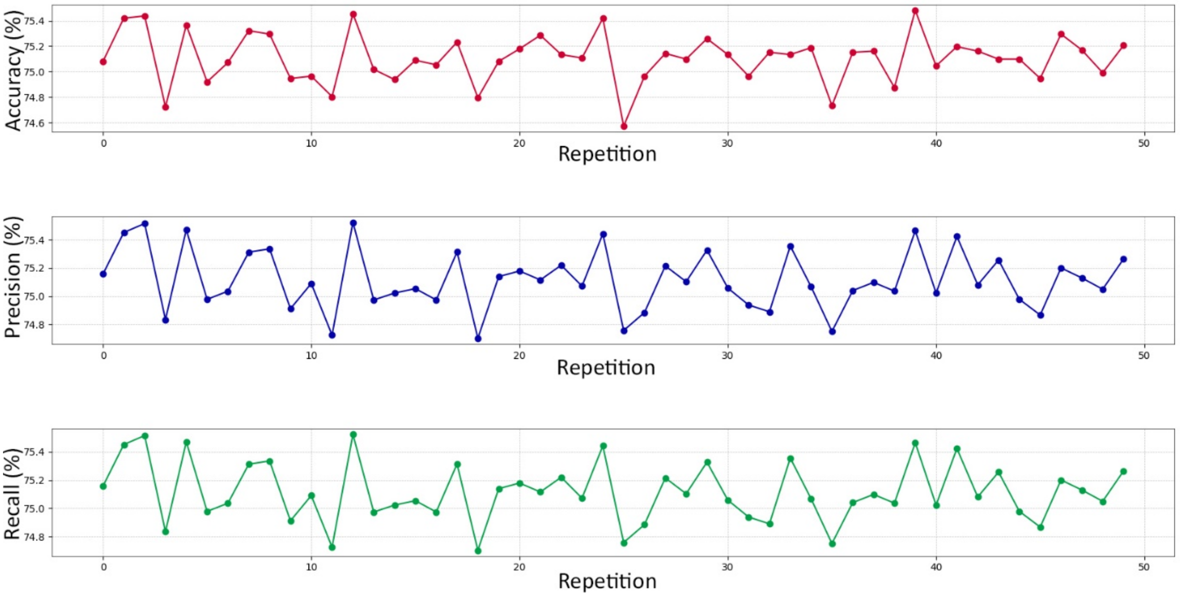
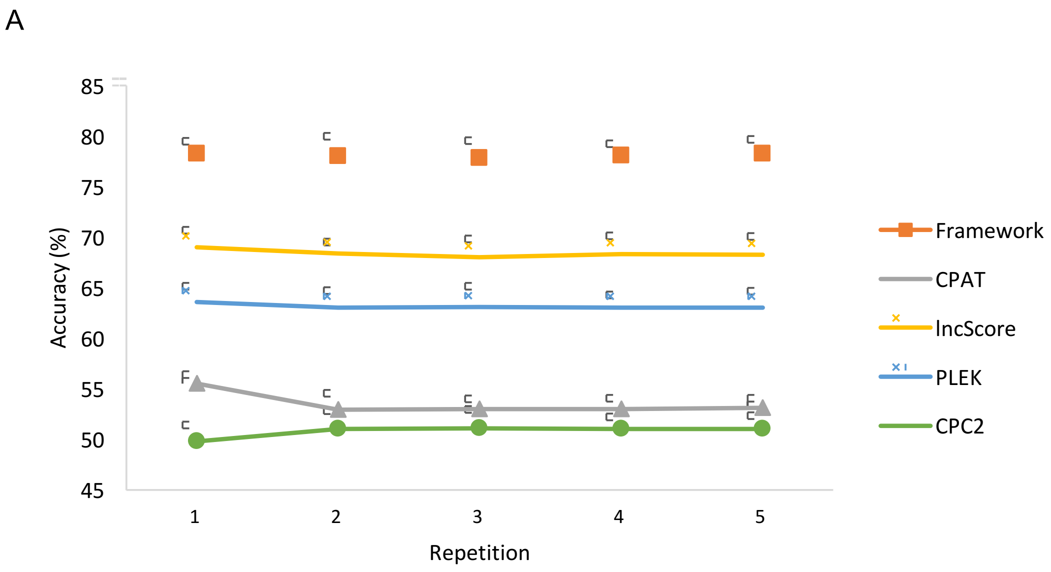


Figure 4.13: Repeated 10-Fold CV plots of Accuracy, Precision and Recall on *A. thaliana* EST dataset.

The results show that the accuracy values fluctuated between 74% and 79% for TAIR10 and EST-annotated datasets which exhibits consistent prediction accuracies. This shows that the framework predicts the lncRNAs in the *A. thaliana* dataset and does not display overfitting of the data.



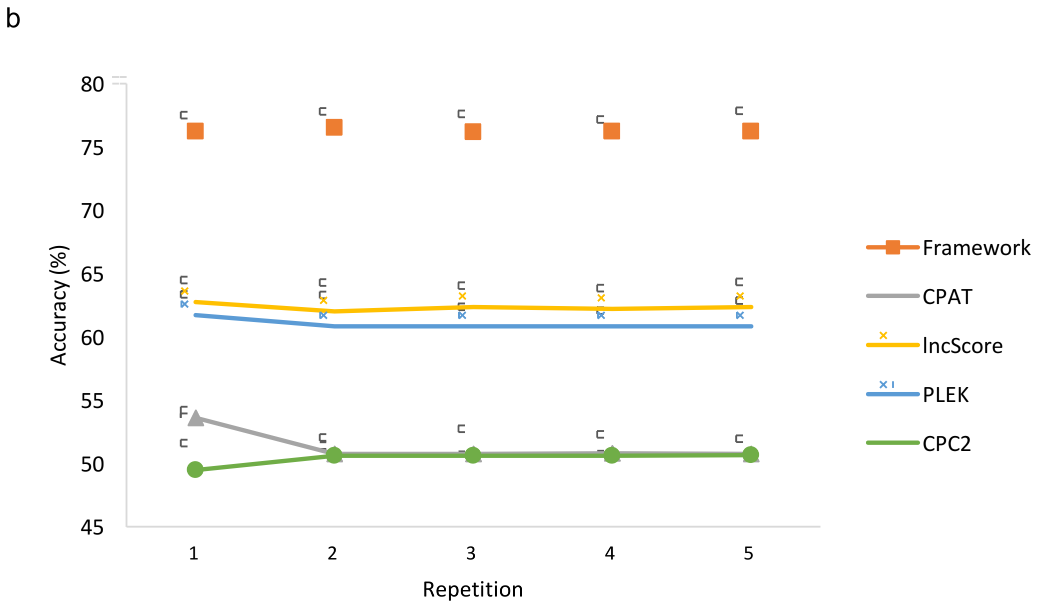


Figure 4.14: Performance benchmarking of the framework using repeated k-fold CV against CPC tools on (a) *A. thaliana* TAIR10, and (b) *A. thaliana* EST annotated datasets. Mean accuracy values are plotted with error bars representing standard deviation.

The performance of the framework was further benchmarked against CPAT, PLEK, lncScore and CPC2 tools using repeated 10-fold CV with data shuffling (Figure 4.14). Five repetitions were performed to evaluate the robustness of the framework in accurately identifying the lncRNA sequences. Results from the analysis on *A. thaliana* TAIR10 (Figure 4.14a) annotated lncRNA sequences shows that the framework achieves a mean accuracy value of 78% in all the repetitions with with standard deviations (SD) values of 78.07±1.67%. Both PLEK and lncScore displays consistent accuracies displaying average accuracies of 63% and 68% respectively. Comparatively lower prediction accuracies values were observed for CPAT and CPC2 with values in the range of 52.9% – 55.5% for CPAT and 49.7% – 51.06% for CPC2.

Prediction accuracies from *A. thaliana* EST-annotated dataset (Figure 4.14b) demonstrate consistently superior performance of the framework in all the repetitions having average accuracy of 76.26%. CPAT and CPC2 exhibited similar profiles where lower accuracies were detected with values ranging between 49.4% – 53.5%. CPAT generated average accuracy of 51.31% whereas CPC2 displayed a mean value of 50.36%. Furthermore, PLEK and lncScore also generated similar profiles with average accuracy values of 61% for PLEK and 62.3% for lncScore. Slight variations in SD values were detected in the folds with SDs of ±2.2 and ±2.03 for PLEK and lncScore respectively. Comparison of repeated 10-fold CV analysis (Figure 4.12 and 4.13) with performance benchmarking results (Figure 4.14) demonstrates insignificant deviation of prediction accuracy values of framework and other CPC tools as represented by SD values around the mean. The framework identifies the lncRNA sequences of varying lengths with reasonable accuracy where the prediction accuracy as well as sensitivity and specificity does not drop beyond 75%. Furthermore, comparison of accuracy and SD values (Figure 4.14) produces an approximate straight line across five repetitions. The small deviation in SD indicates that the accuracy generated by the tools are consistent across varying data and hence exhibits superiority in sequence identification of the framework over other tools.

4.2.9 Sub-classification analysis of A. thaliana lncRNA transcripts

Using the PBC algorithm as mentioned in Section 2.12, 478 lncRNA transcript sequences derived from the TAIR10 dataset were classified into five different types: Sense-overlap, Antisense-overlap, Antisense RNA (ANT), Intergenic (INT) and Bidirectional promoter (BDP). The Sense-overlap and the Antisense-overlap classes were further sub-classified into Sense-Overlap Intronic (SOI), Sense-Overlap Exonic (SOE), Antisense-Overlap Intronic (AOI) and Antisense-Overlap Exonic (AOE). Overall, seven transcript sequences have been classified into seven different classes.

The classification was performed based on the two rules mentioned in Chapter-3 Section 3.7. Table 4.3 shows the number of transcript sequences classified into one of the seven classes. Results indicate that only 2 and 3 sequences have been classified as SOE and SOI using Rules-1 and 2 (as mentioned in Section 3.7), respectively. Whereas the number of sequences classified into AOE and AOI are much higher than the Sense-overlap class. Sequences were also classified into the ANT class. Since this classification is purely based on the strand information, lncRNA sequences located on the antisense strand are classified into the ANT class. This classification is independent of the rules applied on Sense-overlap and the Antisense-overlap categories.

Classification of transcript sequences as AOE and AOI classes using Rule-2 shows much higher proportion when compared to those observed using Rule-1. 122 and 121 sequences were classified into AOE and AOI categories, respectively, which show an increase of additional 50 sequences. Rule-2 classification considers the location of exonic (E) and intronic (I) lncRNA sequences on the sense strand, whereas Rule-1 classification is restricted to searching the E and I sequences on the antisense strand.

The number of transcript sequences classified into the INT class is much lower with only 5 sequences in this category. Whereas 306 sequences have been classified into the BDP class. As illustrated from the statistics, the INT and the BDP classes are independent of the rules applied on the Sense and Antisense overlap classes.

Table 4.3: Sub-classification statistics of the *A. thaliana* apical-shoot lncRNA transcript sequences based on Rule-1 and Rule-2. (SOE: Sense-Overlap Exonic, SOI: Sense-Overlap Intronic, AOE: Antisense-Overlap Exonic, AOI: Antisense-Overlap Intronic, ANT: Antisense RNA, BDP: Bidirectional Promoter).

Rules	SOE	SOI	AOE	AOI	ANT	INT	BDP
Rule-1	2	2	70	69	252	5	306
Rule-2	3	3	122	121	252	5	306

The lncRNA transcript sequences annotated in the TAIR10 database have been classified into two classes: Intergenic (LincRNA; Long intergenic non-coding RNA) and Antisense RNA (NATs; Natural Antisense Transcripts). The annotation set consists of 36 LincRNA sequences and 225 NAT sequences. These sequences were intersected with the results obtained from PBC analysis. By intersection of the PBC results with TAIR10 annotation set using Rules-1 and 2, two LincRNA sequences were found as common sequences. Intersection of LincRNA sequences with Rules-1 and 2 produced two matches out of 36 sequences. However, intersection of NAT sequences produced 151 matches out of 226 sequences using Rule-1, and 194 matches out of 226 sequences using Rule-2. The results of ANT overlap sequences indicate an accuracy of 66.81% using Rule-1 and 85.84% using Rule-2.

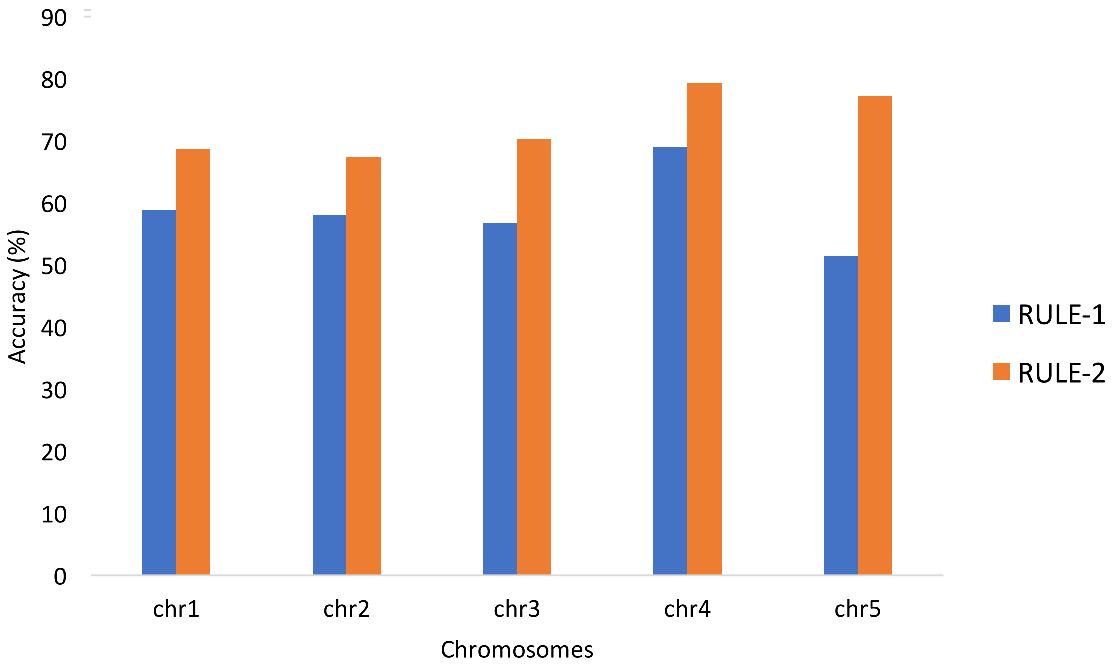


Figure 4.15: Accuracy comparison of Rule-1 and Rule-2 sub-classification using PBC approach on *A. thaliana* dataset for identification of NATs on individual chromosomes.

Results from the sub-classification analysis exhibits that a higher matching percentage can be obtained for the ANT class, whereas a lower proportion of matching INT sequences were obtained. Since the PBC approach relies on the genomic coordinates i.e. start position, end position and strand information, the sequences are purely classified based on the overlapping of the E and I sequences which are derived from the ORF for each sequence. These coordinates are derived from the PLncDB and TAIR databases which confirms the location of each transcript sequence.

Results from the PBC analysis were compared against the annotated lncRNA sequences from the TAIR10 database. Since the annotation in TAIR10 consists mainly consisted of NATs, the NAT annotation results from the PBC Rule-1 and Rule-2 analysis were compared (Figure 4.15). Overall analysis demonstrates that annotation results obtained from Rule-2 indicate a higher prediction accuracy as compared to Rule-1. For the first two chromosomes, the difference between the accuracies is 9.8% and 9.3% whereas for chromosomes 3, 4 and 5, the difference sharply increases to 13.52%, 10.35% and 25.72%. The accuracy for chromosomes 3, 4 and 5 increases steeply, which shows that the PBC algorithm can efficiently identify the various genomic sub-classes in *A. thaliana* species.

Apart from the matching lncRNA sequences, the algorithm also identifies various other sub-classes such as SOE, SOI, AOE, AOI and BDP which have not been annotated and reported in the *A. thaliana* lncRNA sequence data.

4.2.10 Function determination of lncRNA genes based on co-expression data

Using the BMRF approach, function prediction of 478 lncRNA sequences was performed. 14776 protein-coding sequences were used for computing correlations of lncRNA Relative Gene Expression (RGE) values with protein-coding RGE values.

Using a threshold of PCC ≥ 0.8 and PCC ≤ -0.8, 156735 correlations were obtained. These consisted of 118672 positive correlations and 38063 negative correlations. Results from Cuffdiff analysis shows that many lncRNA sequence FPKM values consisted of “NULL” values. Therefore, these lncRNA sequences were removed from the matrix by applying a cutoff of 70%. This means that those lncRNA sequences were removed from the analysis having 70% of the RGE values equal to NULL. This resulted in 402 lncRNA sequences having 156735 correlations with 9674 protein-coding genes forming the LPCS matrix.

The PPI matrix was constructed based on the protein-coding identifiers extracted from the LncRNA-Protein Co-expression Similarity (LPCS) matrix. The matrix was constructed between FPKM values of each pair of lncRNA and mRNA gene and retaining only those lncRNA-protein pairs whose PCC ≥ 0.9 and ≤ -0.9 This resulted in 998566 protein-protein interactions having the interaction strength ≥ 800. The LPCS and the PPI matrices were combined to generate a LPCS-PPI matrix. Filtered protein-coding geneset from the LPCS-PPI matrix was extracted and applied on the 251297 protein-GOTerm association data. This generated a total of 63326 protein-GOTerm association values. BMRF method was applied on the LPCS-PPI matrix and the filtered protein-GOTerm association data which produced 1076958 probabilistic GOTerm associations for the lncRNA sequences.

Results from the BMRF analysis were filtered based on the probability values. 203295 lncRNA-GOTerm connections were obtained having association probability ≥ 0.8. Whereas 814938 connections were found to have probability < 0.1. With probability > 0.1 and probability < 0.8, the proportion decreases to 58725. This shows that most of the GOTerm connections were associated with lower probability. These were removed from the analysis and only the connections having the probability ≥ 0.8 were retained for downstream analysis. This resulted in 111783 GOTerms-function association for 401 lncRNAs.

Table 4.4: Number of lncRNA sequences predicted to have association with function type.

Function Type	**Associated lncRNAs**
DNA or RNA metabolism	159
DNA gap filling	157
DNA or RNA binding	158
Gap-filling	157
Nucleic acid binding	143
Nucleotide binding	3
Other binding	266
Protein binding	270
Receptor binding or activity	238
Transcription	137
5-bisphosphate binding	53
Exonucleolytic	86
Flowering	84
Nucleus	351
Other molecular functions	371
Signal sequence recognition	128
Extracellular	79

Results from the function prediction and GOTerm annotation provide resulted in association of several nuclear and cytoplasmic functions with lncRNA sequences (Table 4.4). Results demonstrate that a greater number of lncRNA sequences (371 and 351 associations) have been associated with "other molecular functions" and "Nucleus". Molecular functional association includes association with "transcription co-factor activity", "transcription coactivator activity", "serine-type endopeptidase inhibitor activity", "cysteine-type endopeptidase inhibitor activity", "protein kinase activator activity" and "DNA polymerase processivity factor activity".

4.2.11 Filtering functions based on plant experimental data

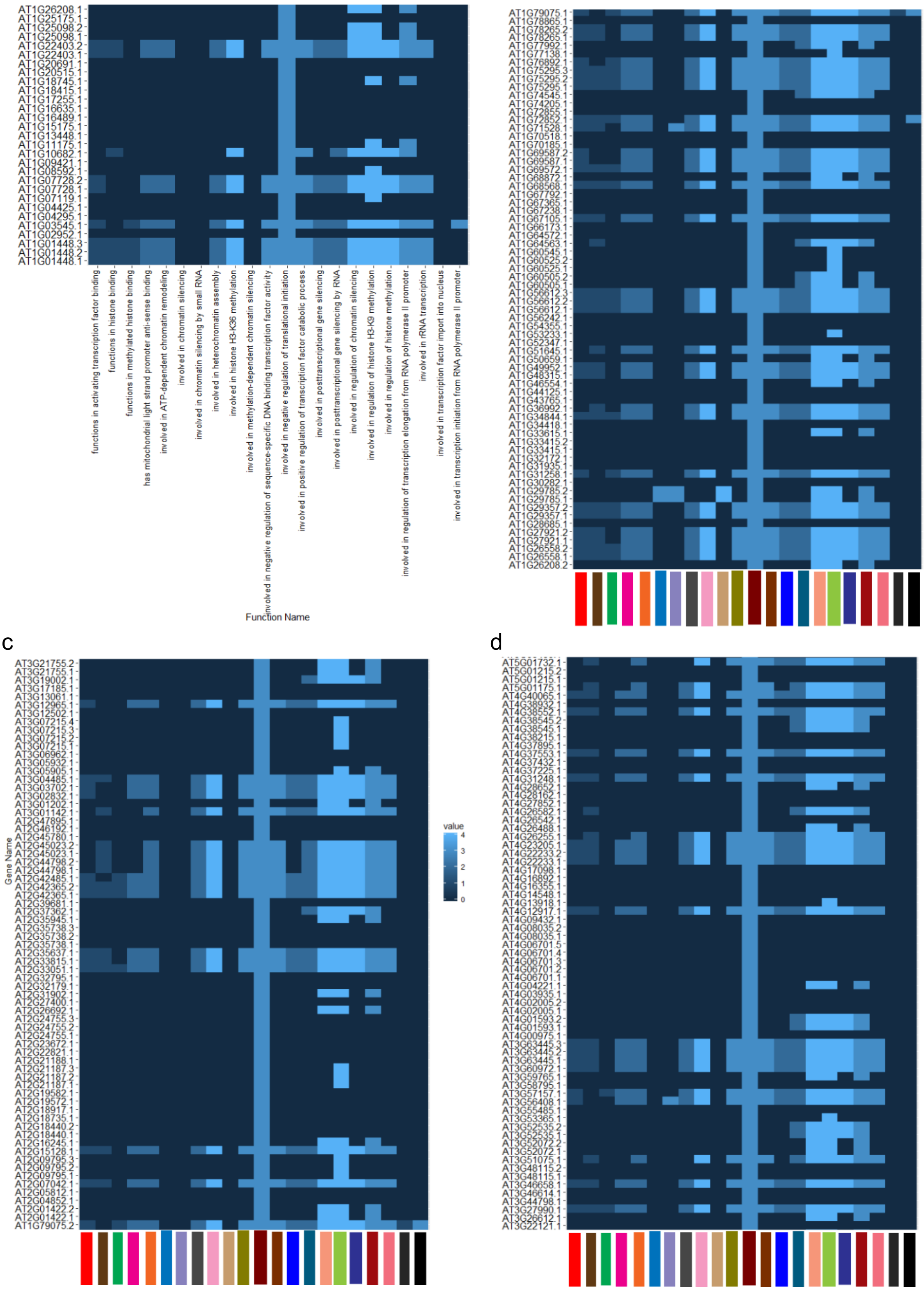
Results from the BMRF analysis predicted molecular functions similar to the protein-coding genes. However, from published experimental studies, it is now known that lncRNA sequences are mainly involved in the regulatory mechanisms. To determine the lncRNA-function association based on the experimental data obtained from the model plant species, the keyword-filtering algorithm was applied on the *A. thaliana* apical-shoot data. The algorithm was applied on protein GOTerms-function association data.

Based on the function annotation obtained from BMRF analysis, the predicted functions were stored in the keyword list. This list was then applied on the protein-coding annotated gene-GOTerm function association data for extracting gene-function pair containing the keyword.

Results from the experimental studies on plant species suggests several regulatory mechanisms such as “promoter methylation”, “translational enhancer”, “antisense transcription” and “alternative splicing regulators” which are specifically associated with nuclear processes as mentioned in Table 2.9. Based on the experimentally-derived functions of plants, keyword-filtering algorithm was applied to construct list of keywords from the function list consisting of “Histone modification”, “Promoter interference”, “Promoter methylation”, “Chromatin association”, “Target mimicry”, “Translational enhancer”, “Antisense transcription”, “Alternative splicing regulators”, “Chromatin loop dynamics” and “nucleus”.

Based on the function prediction results from co-expression analysis, the filtering algorithm was applied on 111783 GOTerms-function associations for 401 lncRNAs. This resulted in 283 lncRNA transcript association with 22 GOTerms. A heatmap analysis of the lncRNA-function association revealed that majority of the lncRNAs were predominantly associated with negative regulation of translational initiation (Figure 4.16). 42 lncRNA genes are AT1G01448, AT1G26558, AT1G22403, AT1G27921, AT1G34844, AT1G49952, AT1G48315, AT1G67105, AT1G68568, AT1G69572, AT1G72852, AT1G75295, AT1G78265, AT1G79075, AT2G07042, AT2G15128, AT2G35637, AT2G33815, AT2G33051, AT2G42485, AT2G42365, AT3G04485, AT3G27990, AT3G46658, AT3G56408, AT3G57157, AT3G60972, AT3G63445, AT4F12917, AT4G22233, AT4G23205, AT4G31248, AT4G37553, AT4G38552, AT4G40065, AT5G07152, AT5G24205, AT5G24735, AT5G28262, AT5G34871, AT5G36002 and AT5G54569 have been found to be associated with transcription factor binding, histone binding, promoter anti-sense binding, chromatin re-modeling, heterochromatin assembly, chromatin silencing, DNA binding transcription factor activity, regulation of transcription factor catabolic process, posttranscriptional gene silencing, regulation of chromatin silencing, regulation of histone H3-K9 methylation, regulation of histone methylation, transcriptional elongation of RNA polymerase II promoter and rRNA transcription.

a b


AT1G26208.1
AT1G25175.1
AT1G25098.2
AT1G25098.1
AT1G22403.2
AT1G22403.1
AT1G20691.1
AT1G20515.1
AT1G18745.1
AT1G18415.1
AT1G17255.1
AT1G16635.1
AT1G16489.1
AT1G15175.1
AT1G13448.1
AT1G11175.1
AT1G10682.1
AT1G09421.1
AT1G08592.1
AT1G07728.2
AT1G07728.1
AT1G07119.1
AT1G04425.1
AT1G04295.1
AT1G03545.1
AT1G02952.1
AT1G01448.3
AT1G01448.2
AT1G01448.1
functions in activating transcription factor binding
functions in histone binding
functions in methylated histone binding
has mitochondrial light strand promoter anti-sense binding
involved in ATP-dependent chromatin remodeling
involved in chromatin silencing
involved in chromatin silencing by small RNA
involved in heterochromatin assembly
involved in histone H3-K36 methylation
involved in methylation-dependent chromatin silencing
involved in negative regulation of sequence-specific DNA binding transcription factor activity
involved in negative regulation of translational initiation
involved in positive regulation of transcription factor catabolic process
involved in posttranscriptional gene silencing
involved in posttranscriptional gene silencing by RNA
involved in regulation of chromatin silencing
involved in regulation of histone H3-K9 methylation
involved in regulation of histone methylation
involved in regulation of transcription elongation from RNA polymerase II promoter
involved in rRNA transcription
involved in transcription factor import into nucleus
involved in transcription initiation from RNA polymerase II promoter
Function Name
AT1G79075.1
AT1G78865.1
AT1G78265.2
AT1G78265.1
AT1G77992.1
AT1G77138.1
AT1G76892.1
AT1G75295.3
AT1G75295.2
AT1G75295.1
AT1G74545.1
AT1G74205.1
AT1G72855.1
AT1G72852.1
AT1G71528.1
AT1G70518.1
AT1G70185.1
AT1G69587.2
AT1G69587.1
AT1G69572.1
AT1G68872.1
AT1G68568.1
AT1G67792.1
AT1G67365.1
AT1G67238.1
AT1G67105.1
AT1G66173.1
AT1G64572.1
AT1G64563.1
AT1G60545.1
AT1G60525.2
AT1G60525.1
AT1G60505.2
AT1G60505.1
AT1G56612.3
AT1G56612.2
AT1G56612.1
AT1G56242.1
AT1G54355.1
AT1G53233.1
AT1G52347.1
AT1G51645.1
AT1G50659.1
AT1G49952.1
AT1G48315.1
AT1G46554.1
AT1G44125.1
AT1G43765.1
AT1G36992.1
AT1G34844.1
AT1G34418.1
AT1G33615.1
AT1G33415.2
AT1G33415.1
AT1G32172.1
AT1G31935.1
AT1G31258.1
AT1G30282.1
AT1G29785.2
AT1G29785.1
AT1G29357.2
AT1G29357.1
AT1G28685.1
AT1G27921.2
AT1G27921.1
AT1G26558.2
AT1G26558.1
AT1G26208.2
c
Gene Name
AT3G21755.2
AT3G21755.1
AT3G19002.1
AT3G17185.1
AT3G13061.1
AT3G12965.1
AT3G12502.1
AT3G07215.4
AT3G07215.3
AT3G07215.2
AT3G07215.1
AT3G06962.1
AT3G05932.1
AT3G05905.1
AT3G04485.1
AT3G03702.1
AT3G02832.1
AT3G01202.1
AT3G01142.1
AT2G47895.1
AT2G46192.1
AT2G45780.1
AT2G45023.2
AT2G45023.1
AT2G44798.2
AT2G44798.1
AT2G42485.1
AT2G42365.2
AT2G42365.1
AT2G39681.1
AT2G37362.1
AT2G35945.1
AT2G35738.3
AT2G35738.2
AT2G35738.1
AT2G35637.1
AT2G33815.1
AT2G33051.1
AT2G32795.1
AT2G32179.1
AT2G31902.1
AT2G27400.1
AT2G26692.1
AT2G24755.3
AT2G24755.2
AT2G24755.1
AT2G23672.1
AT2G22821.1
AT2G21188.1
AT2G21187.3
AT2G21187.2
AT2G21187.1
AT2G19582.1
AT2G19572.1
AT2G18917.1
AT2G18735.1
AT2G18440.2
AT2G18440.1
AT2G16245.1
AT2G15128.1
AT2G09795.3
AT2G09795.2
AT2G09795.1
AT2G07042.1
AT2G05812.1
AT2G04852.1
AT2G01422.2
AT2G01422.1
AT1G79075.2
value
4
3
2
1
0
d
AT5G01732.1
AT5G01215.2
AT5G01215.1
AT5G01175.1
AT4G40065.1
AT4G38932.1
AT4G38552.1
AT4G38545.2
AT4G38545.1
AT4G38215.1
AT4G37895.1
AT4G37553.1
AT4G37432.1
AT4G37225.1
AT4G31248.1
AT4G28652.1
AT4G28162.1
AT4G27852.1
AT4G26582.1
AT4G26542.1
AT4G26488.1
AT4G26255.1
AT4G23205.1
AT4G22233.2
AT4G22233.1
AT4G17098.1
AT4G16892.1
AT4G16355.1
AT4G14548.1
AT4G13918.1
AT4G12917.1
AT4G09432.1
AT4G08035.2
AT4G08035.1
AT4G06701.5
AT4G06701.4
AT4G06701.3
AT4G06701.2
AT4G06701.1
AT4G04221.1
AT4G03935.1
AT4G02005.2
AT4G02005.1
AT4G01593.2
AT4G01593.1
AT4G00975.1
AT3G63445.3
AT3G63445.2
AT3G63445.1
AT3G60972.1
AT3G59765.1
AT3G58795.1
AT3G57157.1
AT3G56408.1
AT3G55485.1
AT3G53365.1
AT3G52535.2
AT3G52535.1
AT3G52072.2
AT3G52072.1
AT3G51075.1
AT3G48115.2
AT3G48115.1
AT3G46658.1
AT3G46614.1
AT3G44798.1
AT3G27990.1
AT3G26612.1
AT3G22121.1

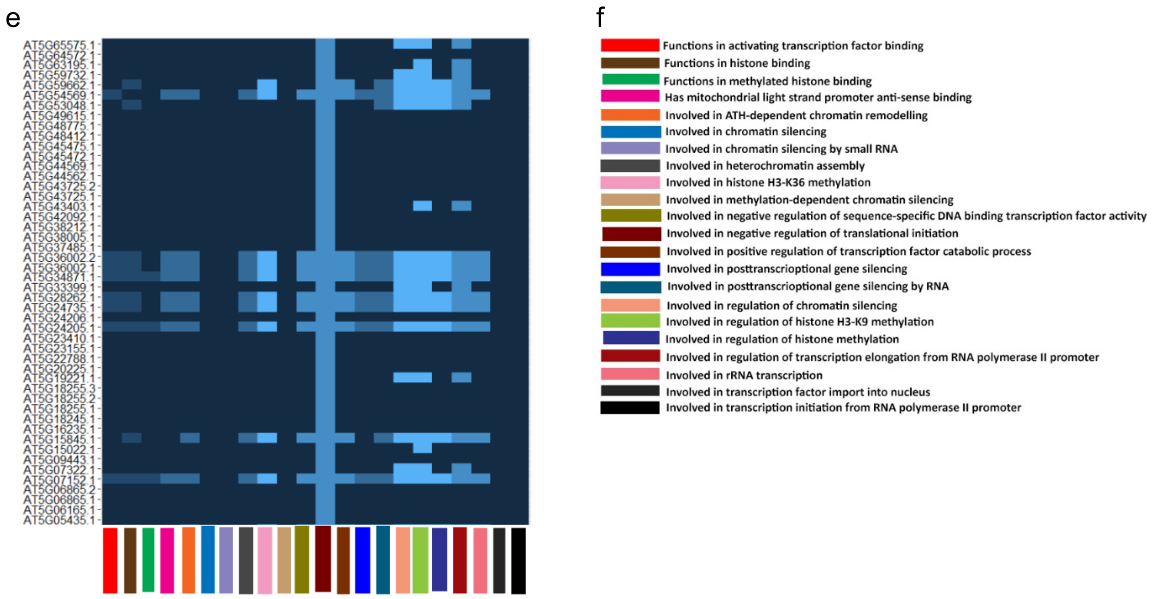


Figure 4.16: Heatmap of lncRNA sequences associated with experimentally determined functions in *A. thaliana* apical-shoot dataset. Function association is represented by dark blue colour. The complete heatmap is broken down to five maps (a – e). The genes are associated with molecular functions. Each molecular function is represented by a specific colour. The legend (f) provides description and association of each colour with its molecular function.

The function association of these 48 lncRNA genes shows that these are broadly classified into transcriptional regulation and histone/chromatin modification. 38 lncRNA genes, namely, AT1G11175, AT1G18745, AT1G25098, AT1G26208, AT1G33615, AT1G46554, AT1G60505, AT1G64563, AT1G74545, AT1G77992, AT2G01422, At2G16245, AT2G31902, AT2G26692, AT2G35945, AT2G37362, AT3G19002, AT3G21755, AT3G26612, AT3G52072, AT3G52535, AT3G59765, AT4G01593, AT4G04221, AT4G13918, AT4G26488, AT4G26582, AT4G28652, AT4G38545, AT5G07322, AT5G19221, AT5G43403, AT5G65575, AT5G59732, AT5G63195, AT5G59662, AT5G54569 and AT5G53048 were found to be associated with post-transcriptional gene silencing, regulation of chromatin silencing, H3-K9 methylation, regulation of histone methylation and transcriptional elongation of RNA polymerase II promoter. As discussed above, the functional association was associated with either gene silencing or chromatin silencing or transcriptional regulation.

From the lncRNA heatmap cluster, 12 lncRNA genes were found to be annotated with the regulation of histone H3-K9 methylation. These are AT1G07119, AT1G08592, AT1G53233, AT1G60525, AT1G60545, AT1G77138, AT2G09795, AT2G21187, AT3G07215, AT3G53365, AT4G13918 and AT5G15022.

Function association by heatmap analysis also reveals that genes AT1G10682, AT1G64563 and AT5G59662 were annotated with histone binding, regulation of chromatin silencing, regulation of histone H3-K9 methylation and regulation of histone methylation. All the function association primarily represents an identical regulatory mechanism. Therefore, the lncRNA genes are associated with histone/chromatin regulation.

4.2.12 Function prediction of DE sequences based on co-expression data

For identification of functions for DE lncRNA sequences, DGE results from the Cuffdiff (Trapnell *et al.*, 2012) analysis was performed. Results from the analysis were filtered based on the q-value metric. Sequences were filtered having cutoff of q-value ≤ 0.05 in ≥ 4 sample pairs. This resulted in 1532 protein-coding genes and 18 lncRNA sequences having significant gene expression values in ≥ 4 sample pairs. 5923 correlations were obtained containing positive and negative lncRNA-protein co-expression connections. 4193 protein-protein interactions were obtained having interaction strength ≥ 0.8. The LPCS matrix and the PPI matrix were concatenated to generate LPCS-PPI matrix consisting of 10116 correlation values.

From the BMRF analysis, 5502 lncRNA-GOTerm associations were obtained using the probability cutoff of 0.8. Function annotation was performed on the GOTerms which resulted in 10116 lncRNA-GOTerm-Function associations. By filtering the unique functions, 574 functions and 40 function types have been found to be associated with 18 lncRNA sequences.

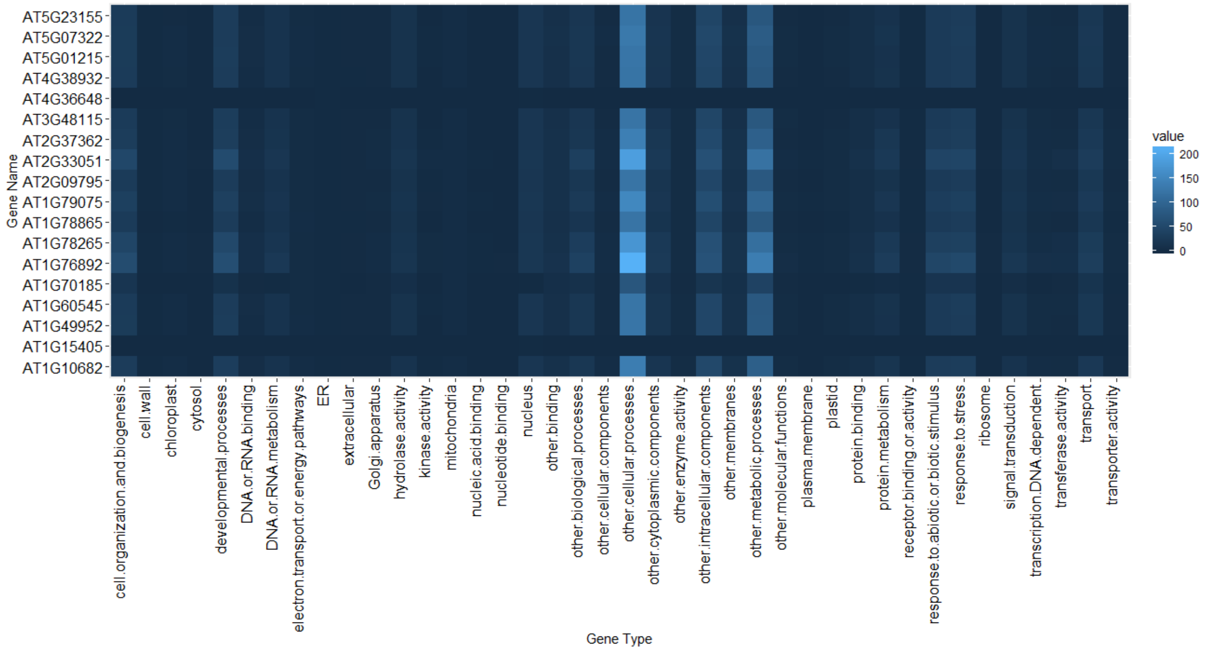


Figure 4.17: Heatmap of lncRNA sequences associated with function type in *A. thaliana* dataset. Number of lncRNA sequences associated with the function type is represented by "value". Lighter colours represent larger lncRNA association whereas darker colours represent lesser lncRNA sequences associated with a function type.

A heatmap of the lncRNA-gene type association was constructed to observe the number of lncRNA sequences associated with each gene type (Figure 4.17). Results from the heatmap analysis show that the majority of the lncRNA genes are associated with "other cellular processes". Particularly, 15 out of 18 genes showed higher association with various functions of "other cellular processes" having frequencies ranging between 150 and 200. The second higher association can be observed in "other metabolic processes" and "other intracellular components" where frequencies ranging between 50 and 100. Other function types such as "developmental processes", "cell organisation and biogenesis", "DNA or RNA metabolism", "Hydrolase activity", "nucleus", "other biological processes", "response to abiotic or biotic stimulus", "response to stress", "signal transduction" and "transport" showed moderate association frequencies.

Function annotation results from the BMRF analysis of DE lncRNA dataset were filtered based on dictionary of keywords extracted from the experimentally-derived lncRNA regulatory functions. The keyword-filtering algorithm was applied for filtering the Gene-Function associations. From the analysis, 16 lncRNA genes were found to have association with 34 regulatory functions, some of which included histone modification, regulation of transcription from RNA polymerase II promoter, DNA-templated transcription initiation, single-stranded DNA binding and alternative RNA splicing. This approach was implemented to identify genes with similar functions.

Heatmap analysis (Figure 4.18) of the lncRNA sequences demonstrated the degree of association/non-association of lncRNA genes to regulatory mechanisms. 16 lncRNA sequences were primarily associated with 6 functions, namely, translational elongation, regulation of vesicle targeting, posttranscriptional gene silencing, heterochromatin assembly, transcription coactivator activity and chromatin binding. 15 lncRNAs were associated with 8 regulatory functions, namely, rRNA transcription, RNA splicing, regulation of transcription elongation, histone phosphorylation, histone H3-K36 methylation, histone acetylation, single stranded DNA endodeoxyribonuclease activity and single-stranded DNA binding.

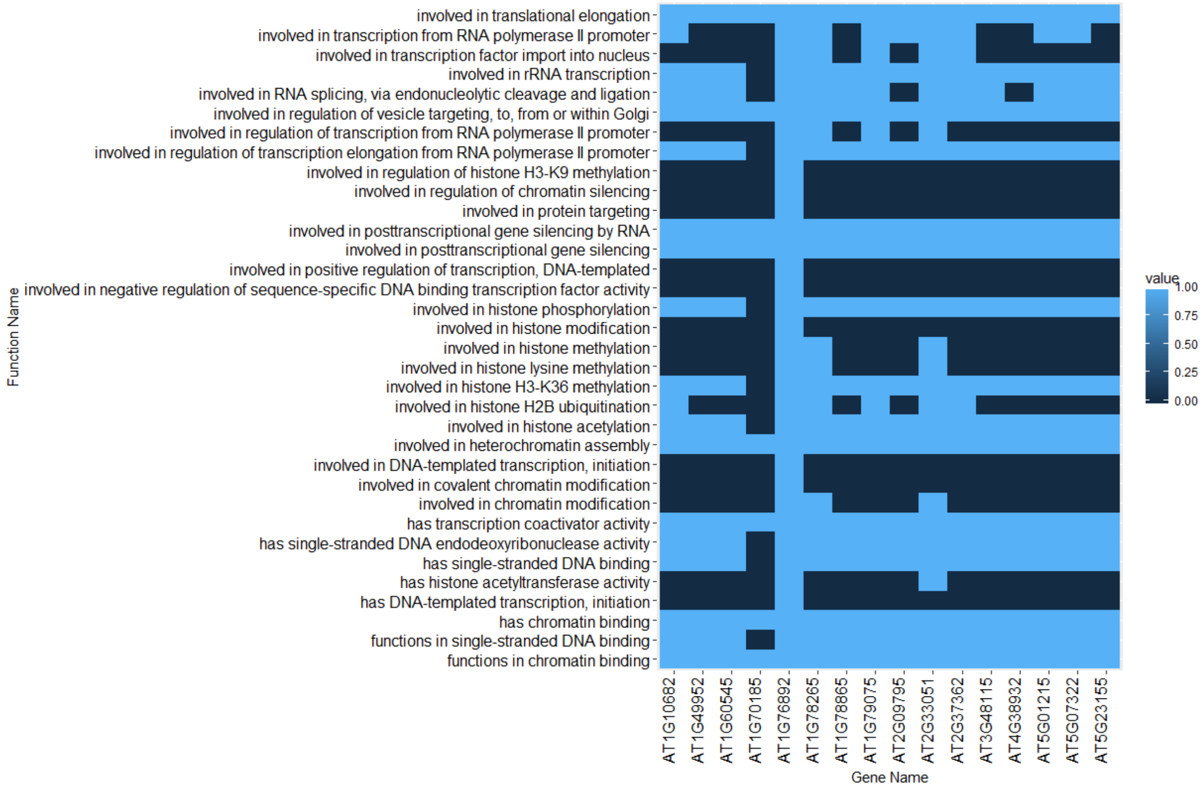


Figure 4.18: Heatmap of lncRNA-function association based on keyword filtering approach in *A. thaliana* dataset. X-axis shows the lncRNA genes and the y-axis shows the function name. Value represents number of lncRNA genes associated with the function. “Light blue” colour represents association and “dark blue” colour represents non-association.

None of the lncRNA sequences except AT1G76892 were found to play a role in 9 regulatory functions. AT1G76892 was found to play a significant role in the regulation of histone H3-K9 methylation, regulation of chromatin silencing, protein targeting, positive regulation of transcription, negative regulation of sequence-specific DNA binding transcription factor activity, histone modification, DNA-templated transcription, covalent chromatin modification and histone acetyltransferase activity. To summarize, AT1G76892 was primarily involved in the regulation of transcription activity or chromatin modification or histone modification. A summary of the number of lncRNA sequences associated with intra-nuclear molecular function has been provided in Table 4.5.

Table 4.5: Number of lncRNA sequences predicted to have association with experimentally determined regulatory functions in *A. thaliana* dataset.

Molecular function	**Number of associated lncRNAs**
has single-stranded DNA endodeoxyribonuclease activity	15
involved in RNA splicing, via endonucleolytic cleavage and ligation	13
functions in chromatin binding	16
has chromatin binding	16
functions in single-stranded DNA binding	15
has single-stranded DNA binding	15
has transcription coactivator activity	16
has histone acetyltransferase activity	2
involved in DNA-templated transcription, initiation	1
has DNA-templated transcription, initiation	1
involved in regulation of transcription from RNA polymerase II promoter	4
involved in transcription from RNA polymerase II promoter	9
involved in translational elongation	16
involved in protein targeting	1
involved in rRNA transcription	15
involved in histone H3-K36 methylation	15
involved in posttranscriptional gene silencing	16
involved in chromatin modification	3
involved in covalent chromatin modification	1
involved in histone modification	1
involved in histone methylation	3
involved in histone phosphorylation	15
involved in histone acetylation	15
involved in heterochromatin assembly	16
involved in regulation of chromatin silencing	1
involved in histone H2B ubiquitination	6
involved in regulation of transcription elongation from RNA polymerase II promoter	15
involved in histone lysine methylation	3
involved in posttranscriptional gene silencing by RNA	16
involved in transcription factor import into nucleus	5
involved in negative regulation of sequence-specific DNA binding transcription factor activity	1
involved in positive regulation of transcription, DNA-templated	1
involved in regulation of vesicle targeting, to, from or within Golgi	16
involved in regulation of histone H3-K9 methylation	15

Table 4.6 indicates that the lncRNA-function association is broadly divided into three clusters. The first cluster consists of the lncRNA sequences involved in DNA or chromatin binding or transcriptional silencing activities. The second cluster contains sequences between 5 and 14

which have been associated with transcription, ubiquitination or in the import of transcription factor into nucleus. The third cluster contains less than 5 sequences which are associated with transcriptional regulation, protein targeting, histone modification and in the regulation of sequence-specific DNA binding. Based on this broad classification, a generalized functional association can be performed.

4.2.13 Experimental validation of lncRNA functions from BMRF analysis

Results of BMRF analysis in *A. thaliana* apical-shoot data were verified from the experimentally reported lncRNA function association data of *A. thaliana*. A summary of experimentally reported lncRNA-function association data presented by Liu et al. (2015) were used for validation of the results. Results from the experimental studies show that some of the lncRNA sequences such as, COLD ASSISTED LONG ANTISENSE INTRAGENIC RNAs (COOLAIR) and COLD ASSISTED INTRONIC NONCODING RNA (COLDAIR) has been found to be primarily involved in histone modifications via epigenetic regulation and promoter interference (Csorba *et al.*, 2014; Kim, Xi and Sung, 2017).

Experimental data also shows that certain lncRNAs found in *A. thaliana*, *O. sativa* and *S. lycopersicum* such as IPS1, *Cis-NAT*$_{PHO1;2}$*, OsPI1* and *TPS11* have been found to be involved in phosphate homeostasis as translational enhancer (Liu, Muchhal and Raghothama, 1997; Wasaki *et al.*, 2003; Franco-Zorrilla *et al.*, 2007; Jabnoune *et al.*, 2013). *ASCO-lncRNA* found in *A. thaliana* was found to act as alternative splicing regulator in lateral root development (Bardou *et al.*, 2014) whereas *APOLO* lncRNA has been involved in chromatin loop dynamics in auxin controlled development (Ariel *et al.*, 2014). Furthermore, *asHSFB2a* lncRNA in *A. thaliana* has been found to be involved in vegetative and gametophytic development (Wunderlich, Groß-Hardt and Schöffl, 2014).

Table 4.6: List of lncRNA sequences associated with experimentally verified molecular functions.

Molecular Function	**GO Term**	**Number of associated lncRNAs**	**Average probability of association**
Cellular phosphate ion homeostasis	GO:0030643	89	0.943
Post-embryonic root development	GO:0048528	30	0.803
Chromatin organisation	GO:0006325	402	0.002
Chromatin silencing	GO:0006342	1	0.963
Developmental vegetative growth	GO:0080186	156	0.967

The lncRNA-function association results from the BMRF analysis were verified by matching the GOTerms corresponding to the functions mentioned above. Results from the experimental verification analysis (Table 4.6) of the lncRNAs and molecular functions demonstrate that there is a reasonable number of matching lncRNA sequences with experimentally verified molecular functions. 89 lncRNA sequences were predicted to be involved in cellular phosphate ion homeostasis with an average probability of 0.943. Whereas only 30 lncRNA transcripts were found to be involved in post-embryonic root development. Dicot plants such as *A. thaliana*, *Z. mays*, *H. vulgare* and *O. sativa* consists of shoot-borne crown roots that branches sequentially and form a herringbone-like structure (Orman-Ligeza *et al.*, 2013). The crown and lateral root formation in maize and barley includes post-embryonic developmental processes through which root nodes arise.

Post-embryonic root development function association results show that 30 lncRNA sequences were found to be associated with probability of 0.803. However, none of the lncRNA sequences were predicted to have chromatin organisation function association. 402 lncRNAs were predicted with probability of 0.002. In contrast to GO:0006325 association, only one lncRNA (AT1G29785) was predicted to be involved in chromatin silencing with probability of 0.963.

LncRNA sequences predicted to be associated with developmental vegetative growth showed a much higher proportion of 156 with average probability of 0.967. These results confirm that lncRNAs which are co-expressed along with protein-coding genes share a similar molecular function.

Heatmap analysis of the molecular function association (Figure 4.19) shows that a large cluster of the genes are primarily associated with developmental vegetative growth (GO:0080186). The second largest cluster of genes is associated with cellular phosphate ion homeostasis (GO:0030643). These two clusters primarily have genes with higher probability values. Whereas the number of genes in the third cluster is much smaller with probability values ranging from 0.5 to 0.8. It can be observed that most of the genes share the same functions as the above-mentioned two GOTerms but some do not.

To broadly specify a molecular function to the lncRNA genes, the molecular function having lowest associated lncRNA genes are given the higher preference. The second preference is given to the function having third highest number of genes. The number of genes associated in this cluster is subtracted from the genes present in the fourth cluster. For calculating gene association in the second cluster, genes are subtracted from the genes present in the third and fourth clusters. Similarly, gene subtraction in cluster-1 is performed by subtracting genes from previous clusters.

GO:0006342 consists of only one lncRNA gene (i.e. AT1G29785), therefore there is a higher probability of association. GO:0048528 consists of 30 genes and does not have any overlap with GO:0006342. Thus, 30 genes are associated with post-embryonic root development function. Genes belonging to GO:0030643 have 89 genes. Many genes have overlaps with GO:0048528, consequently an intersection of these genes is performed, and non-matching genes are extracted. Gene-function filter algorithm was implemented for separating the overlapping and non-overlapping clusters. From the analysis, genes associated with post-embryonic root development and cellular phosphate ion homeostasis were compared. Since the algorithm finds the lowest number of gene-function association cluster, 30 genes were associated with post-embryonic root development. Based on the size of the gene-function association cluster, the algorithm selects the second smallest cluster and removes any overlapping genes matching with the cluster having smallest gene-function association (i.e. genes associated with embryonic root development). This resulted in 71 non-overlapping genes having role in cellular phosphate ion homeostasis. In the next iteration, the algorithm selected the genes associated with developmental vegetative growth. Removal of genes overlapping with “embryonic root development” and “cellular phosphate ion homeostasis” from the 156 gene-function cluster produced 128 genes functionally associated with developmental vegetative growth function.

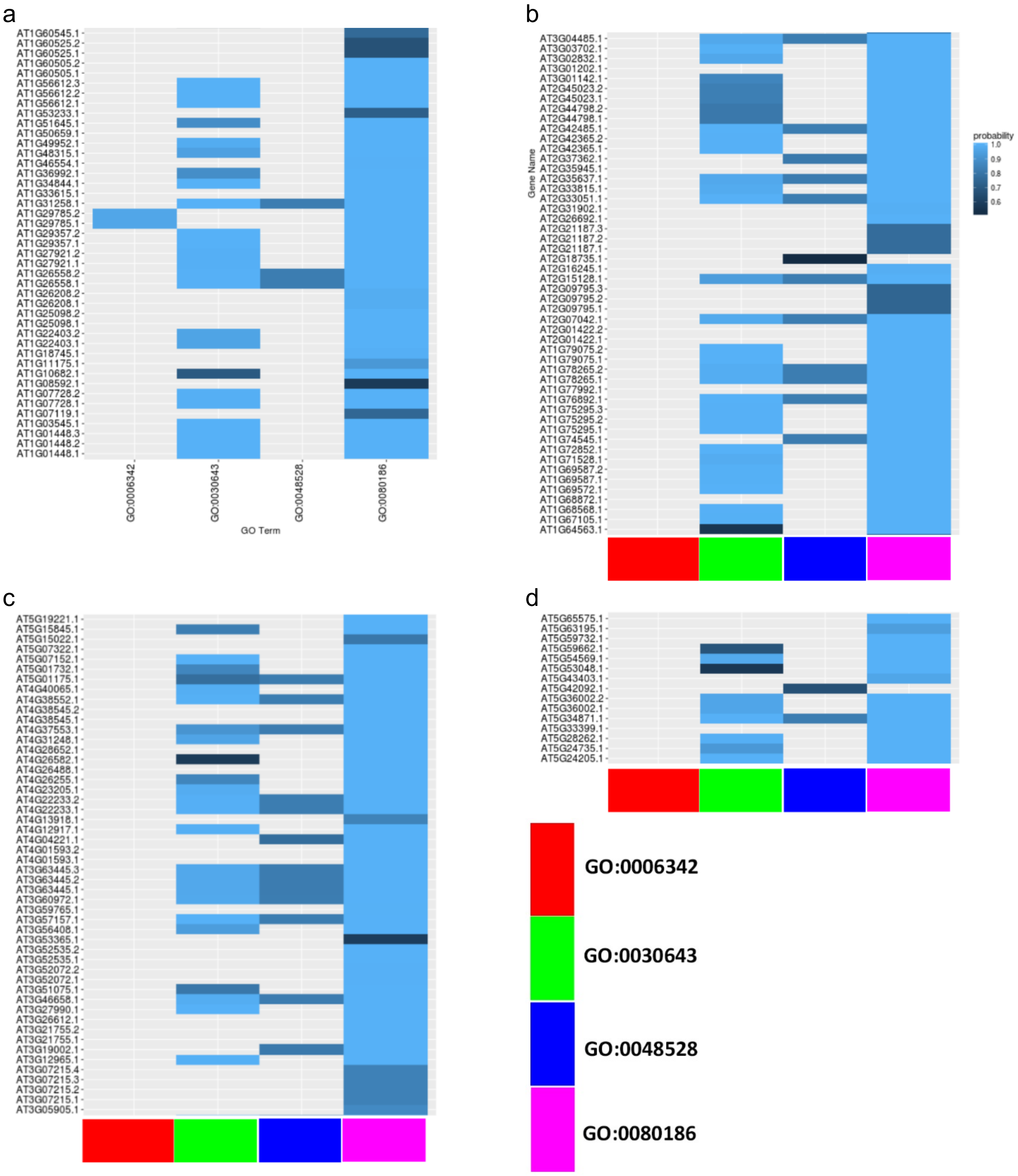


Figure 4.19: Heatmap of lncRNA sequences associated with experimentally verified molecular function in *A. thaliana* dataset. The x-axis shows the GO terms and the y-axis shows the gene names. Function association is represented by dark blue colour. The complete heatmap is broken down to four maps (a – d). The genes are associated with molecular functions. Each molecular function is coloured represented by a specific colour. The legend in (d) provides description and association of each colour with the GOTerm.

Results from function prediction based on co-expression analysis are filtered based on higher correlation displaying association of lncRNA genes with several molecular and regulatory functions as well as demonstrating similarity with experimentally-published results. The analysis was undertaken for investigating function association based on Pearson correlation analysis. Since, determination of lncRNA and protein interactions are crucial for governing accurate biological functions, NRLMF analysis was conducted for deriving the LPI pairs. Results from NRLMF were compared against co-expression data, thereby providing degree of correlation between the results.

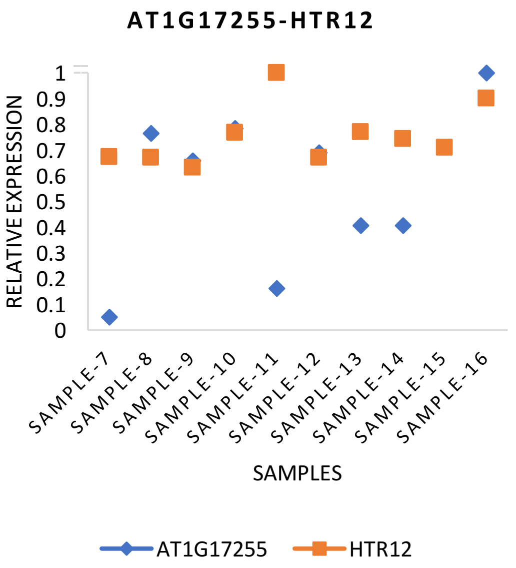
4.2.14 Function prediction of the lncRNA sequences based on NRLMF and BMRF analysis

For predicting the functions of the lncRNA sequences, a subset of 50 lncRNA transcripts and 402 protein-coding sequences were selected from a pool of 478 known lncRNAs and 35343 protein-coding sequences. For predicting the functions of lncRNAs, NRLMF analysis was conducted for obtaining physical interactions between lncRNA and protein-coding genes. A sequence similarity matrix was constructed between *A. thaliana* and *H. sapiens* sequences generating lncRNA-lncRNA and protein-protein similarity matrices as mentioned in Section 2.13.1. From *H. sapiens*, the 50 lncRNAs and 402 protein-coding sequences were selected. An adjacency matrix was formed between the lncRNA and protein-coding genes (i.e. a 100 × 804 matrix).

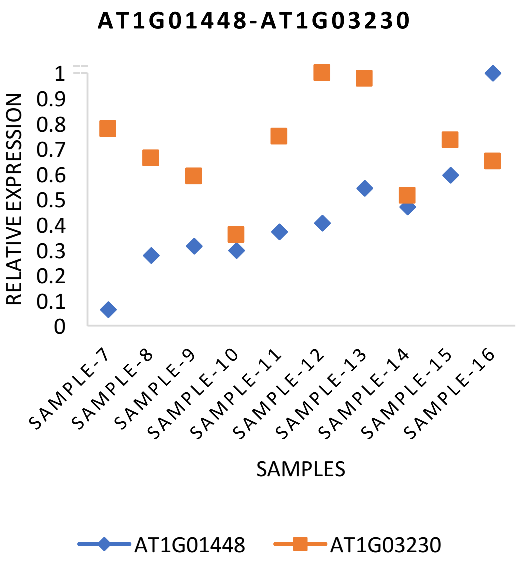
Results from the NRLMF analysis generated 3192 interactions having scores between 0.8865 and 0.1801. These were filtered to obtain the interactions between lncRNAs and protein-coding sequences in *A. thaliana* species. Based on a threshold value of 0.7, 184 novel interactions were obtained between 50 lncRNAs and 6 protein-coding genes having scores ranging between 0.7 and 0.8865.

Correlation analysis of lncRNA and protein-coding co-expression data shows correlations ≥ -0.5 for lncRNA sequences (Figure 4.20). Figure 4.20 illustrates relative expression of the NRLMF-derived lncRNA and protein-coding genes predicted to have interactions with scores > 0.8. The relative expression of genes suggests that most of the protein-coding genes were highly expressed during the transition phase (i.e. S9 to S13). LncRNA sequences on the other hand, displayed an increase in expression during the transition phase having peak value at S7-S13 (Figure 4.20c–f). AT1G17255 and HTR12 (Figure 4.20a) exhibited unique expressional profiles producing opposite values at S7-S11, S7-S13 and S7-S14 samples. However, AT1G17255 exhibits similar expression values for S7-S8, S7-S9, S7-S10, S7-S12, S7-S15 and S7-S16 which demonstrates co-expression during the majority of the floral transition period.

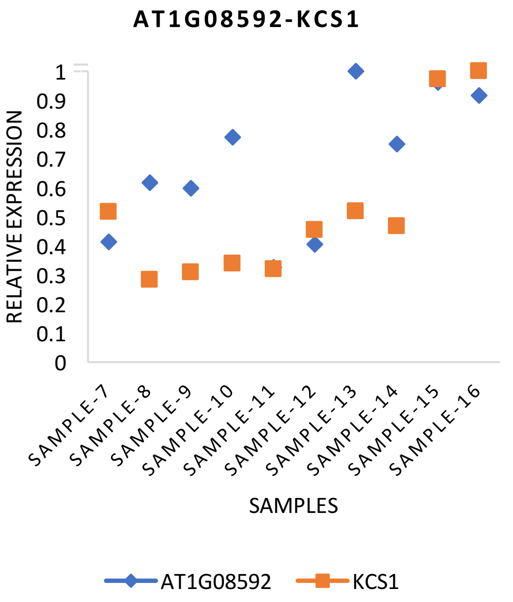
a



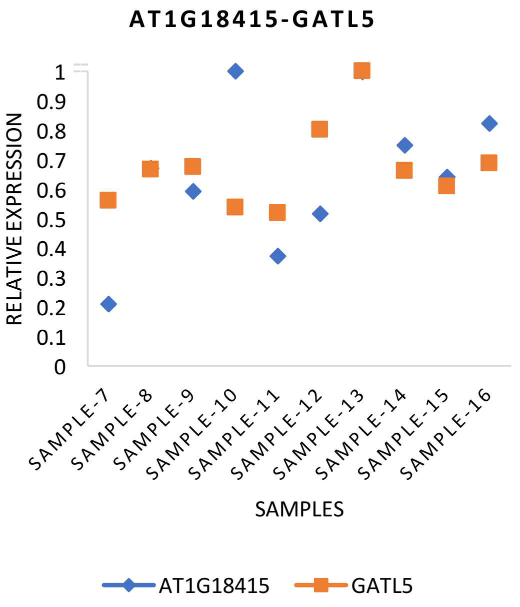
b



c



d



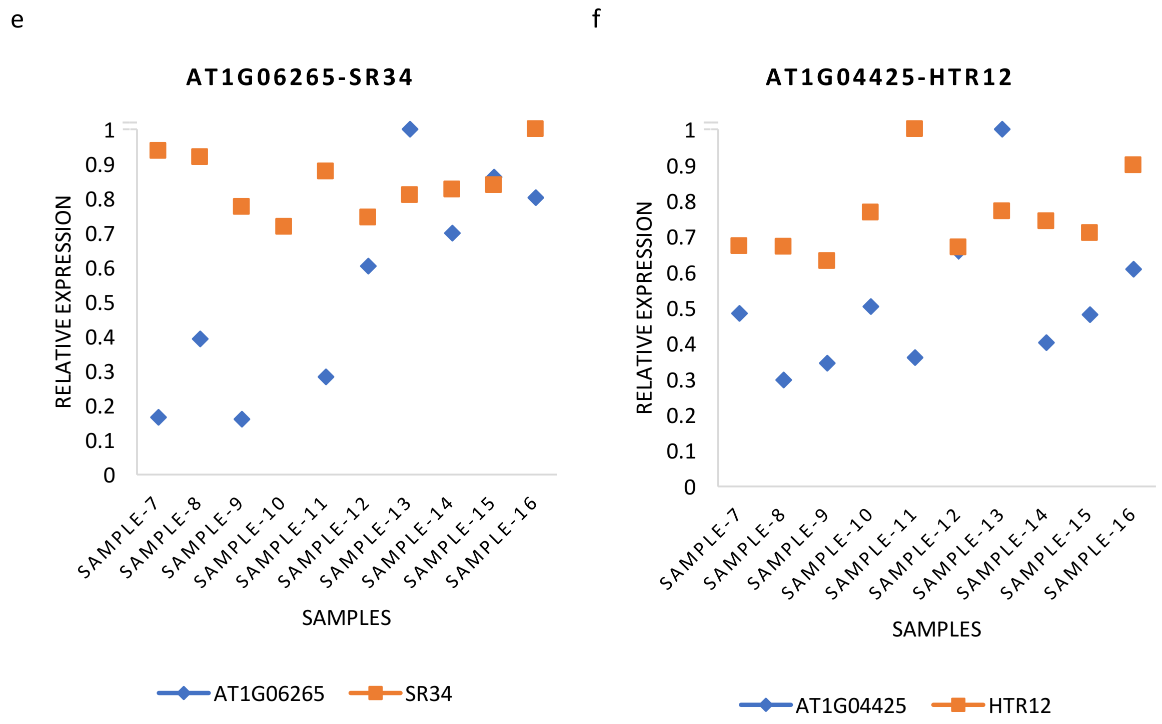


Figure 4.20: Relative gene expression of lncRNA and protein-coding genes predicted to have interactions in *A. thaliana* dataset.

An analysis of the sequence similarities of lncRNA and protein-coding sequences in *A. thaliana* shows that highest similarities range between 51–56% for *A. thaliana* (lncRNA) and *H. sapiens* (lncRNA) (Figure 4.21a), and 50–59% for *A. thaliana* (proteins) and *H. sapiens* (proteins) (Figure 4.21b). The analysis was performed by matching single *A. thaliana* sequence against n HS sequences (where n=402). The *H. sapiens* sequence producing the highest similarity was selected and plotted on the scatter plot. The computation of LPI scores fundamentally depend on the highest matching similarity of lncRNA and protein sequences. Since the analysis has been performed on a subset of lncRNA and protein-coding sequences, a higher sequence similarity is likely to be observed on a much larger set of sequences.

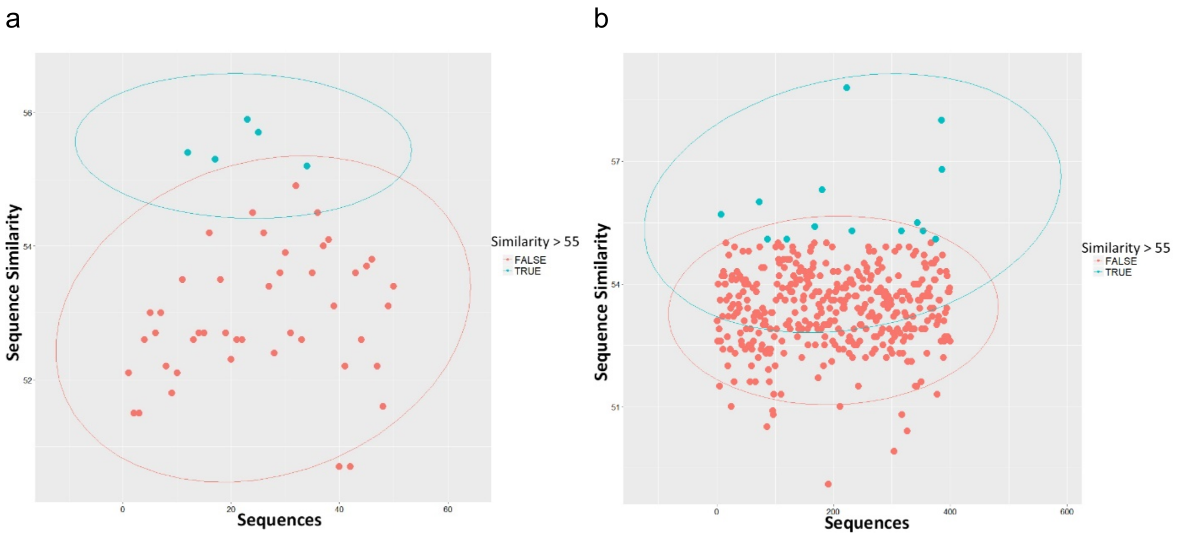


Figure 4.21: Scatter plots of sequence similarities in *A. thaliana* dataset. (a) lncRNA-lncRNA SSM plot, and (b) protein-protein SSM plot.

Using the BMRF approach, function prediction for 50 lncRNA sequences was performed. A PPI matrix was constructed based on the protein-coding identifiers extracted from the LPI matrix which resulted in 1035 protein-protein interactions. The LPI and the PPI matrix were combined to generate a LPI-PPI matrix as mentioned in Section 2.12.3. Filtered protein-coding geneset from the LPI-PPI matrix was extracted and applied on the 251297 protein-GOTerm association data. This generated a total of 745 protein-GOTerm association values. The BMRF method was applied on the LPI-PPI matrix and the filtered protein-GOTerm association data which produced 5520 probabilistic GOTerm associations for the lncRNA sequences.

Results from the BMRF analysis were filtered based on the probability values. 184 lncRNA-GOTerm connections were obtained having association probability ≥ 0.8. The lncRNA sequences were found to be associated with two functions: (1) located in cytosol, and (2) located in nucleus.

Results from the LPI analysis using NRLMF produced 184 interactions. Functions of the protein-coding genes shows that the majority of the genes are annotated with "located in cytosol ribosome", "located in chromatin", "located in nucleus", "located in golgi appratus", "located in plant type cell wall", "located in response to salt stress" and "located in extracellular region" functions. Results from BMRF analysis shows that similar functions have been assigned based on protein-GO association data.

Another analysis for determination of LPIs and functions using BMRF was conducted with a different set of protein-coding genes for identifying intra-nuclear regulatory functions. 2434 novel LPIs were obtained having score ≥ 0.7 where 50 lncRNA genes were found to interact with 75 proteins. Using protein-coding genes as IDs, 2481 PPIs were retained having interaction score ≥ 0.5. Altogether, 4914 relationships were produced containing lncRNA-protein and protein-protein interactions.

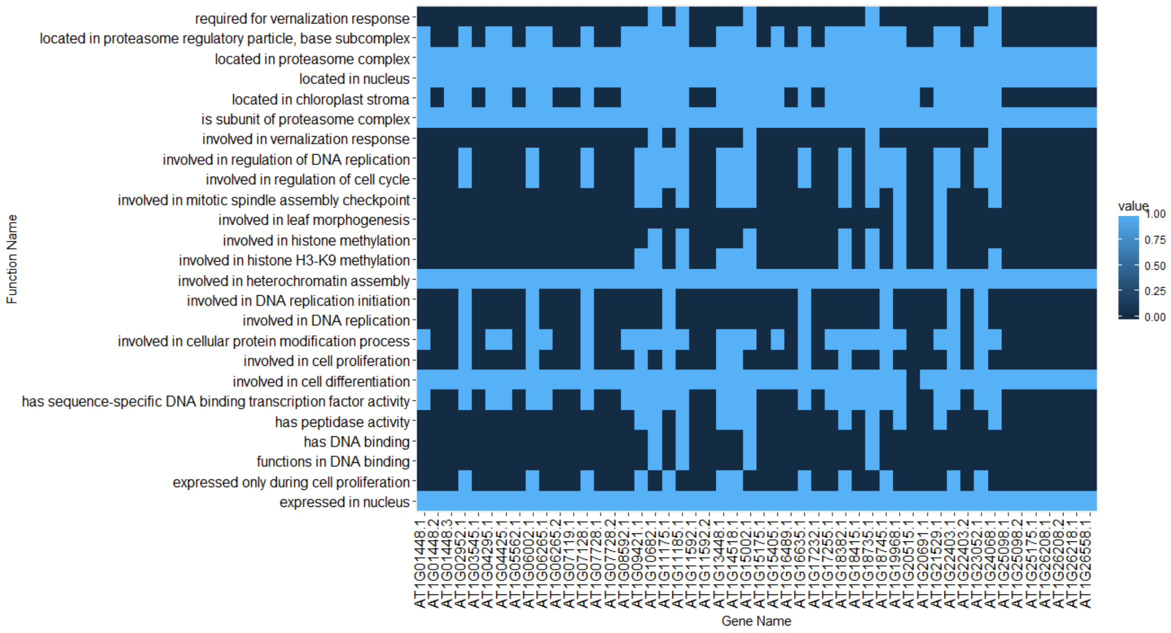


Figure 4.22: Heatmap of lncRNA sequences associated with functions in *A. thaliana* with probability above 0.7. Number of lncRNA sequences associated with the function type is represented by "value". Lighter colours represent larger lncRNA association whereas darker colours represent lesser lncRNA sequences associated with a function type.

From the BMRF function prediction analysis, 427 gene-function-probability associations were obtained. The gene-function association can be visualized through a heatmap of gene names and functions (Figure 4.22). Results demonstrate that all lncRNA genes were found to be located in nucleus, involved in heterochromatin assembly, involved in cell differentiation, expressed in nucleus and located in proteasomal complex with probability values ranging between 0.7 and 1.0 (Appendix A, Table A.1). Twelve genes were found were associated with "regulation of DNA replication", "expressed only during cell proliferation", "has sequence-specific DNA binding transcription factor activity", "cellular protein modification process", and located in "proteasome regulatory particle". Five lncRNA genes were found to be particularly associated with "vernalization response", "histone methylation", and "peptidase activity". From the BMRF analysis, function association can be derived which provides more insights into specific roles of lncRNAs in the genome. Additionally, AT1G15405.1 and AT1G10682.1 were found to be DE expressed during the floral transition phase. AT1G15405.1 was found to be associated with the following functions: (1) expressed in nucleus, (2) involved in cell differentiation, (3) involved in heterochromatin assembly, (4) subunit of proteasome complex, (5) located in chloroplast stroma, (6) located in nucleus, and (7) located in proteasome complex. Whereas AT1G10682.1 has been found to associated with the following functions: (1) expressed in nucleus, (2) functions in DNA binding, (3) has peptidase activity, (4) has sequence-specific DNA-binding transcription factor activity, (5) involved in cell differentiation, (6) involved in cellular protein modification process, (7) involved in histone H3-K9 methylation, (8) involved in mitotic spindle assembly checkpoint, (9) involved in regulation of cell cycle, (10) involved in DNA replication, (11) involved in vernalization response, and (12) located in proteasome complex. These function annotation results of DE lncRNA genes clearly demonstrates primary roles in transcription factor regulation and histone methylation of lncRNA AT1G10682.1. LncRNA AT1G15405.1 has been predicted to be located in proteasome complex which help in degradation of intracellular proteins.

Results from the NRLMF-based LPI prediction demonstrates lower to moderate correlation in the co-expression of the genes. The results exhibit mismatch between the LPI pairs derived from the co-expression-based analysis and those derived from NRLMF analysis. This suggests that prediction of functions should not be made exclusively on the basis of co-expression-based analysis. To strengthen the function prediction approach, computing the LPI pairs is essential for increasing true positive LPI pairs and reducing false positive LPI pairs.

4.2.15 LPI-PPI network analysis of A. thaliana apical-shoot dataset

Regulatory network constructed for function prediction of lncRNA and protein-coding sequences were analysed using Cytoscape. Analysis of the 50 lncRNA and 402 protein-coding sequences was performed with Cytoscape to observe the distribution of nodes and connectivity of the edges in the regulatory network. To evaluate the node degrees in the network, Betweenness Centrality (BC), Closeness Centrality (CC) and node degree distribution were analysed. BC is a measure of centrality of nodes in the network (Prountzos and Pingali, 2013). It is equal to number of shortest paths that emerge from the nodes to other nodes in the network. BC reflects the amount of control a node exerts over the interaction of other nodes in the network. CC is a measure of centrality and is computed by the sum of shortest path length between the nodes in the network (Sabidussi, 1966). The centrality measures were computed for each node in the network. The BC and CC values for each node were calculated for measuring the degree of closeness of each node and its interaction with other nodes in the network.

Analysis of the network topology was compared to scale-free topology by fitting a power-law distribution. An examination of the node degrees in the regulatory network revealed a power-law distribution with a slope of -0.925 and $R^2 = 0.674$ (Figure 4.23). Correlation analysis of the number of nodes in the network exhibited a PCC of 0.957 (Figure 4.23a). This advocates that the nodes in the network follow the slope of the fitted power-law distribution with inverse proportionality i.e. the number of nodes decreases with increase in the degree. The node degree represents the number of connections of a node with other nodes in the network. This is represented by neighbourhood connectivity. It can be observed that greater proportion of nodes possess single connectivity to other nodes. Smaller proportion of nodes with degree > 10 and < 100 have ~50-75 connections. These are called hub-nodes which regulate the expression of multiple genes and hence are crucial in the gene-regulatory network (Figure 4.23a). Most connections have degrees between 1 and 10. This suggests that large number of nodes in the network have 3-4 interacting partners. Regulatory network of the lncRNA genes advocates that a single lncRNA gene interacts with protein. The protein regulates another protein with protein through binding or catalysis events. Thus single gene possess at least two to three interacting partners which can be observed from the node degree distribution analysis.

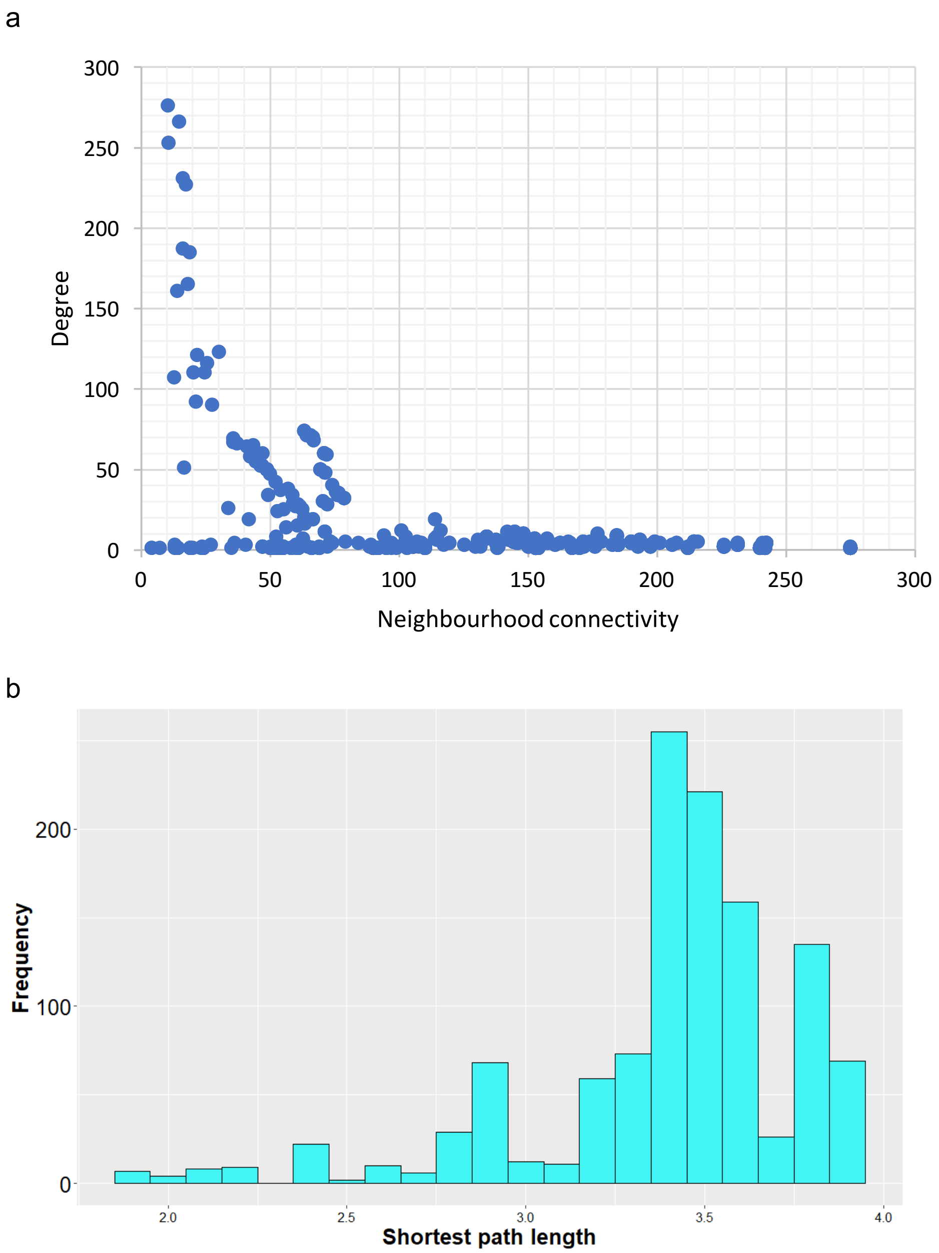
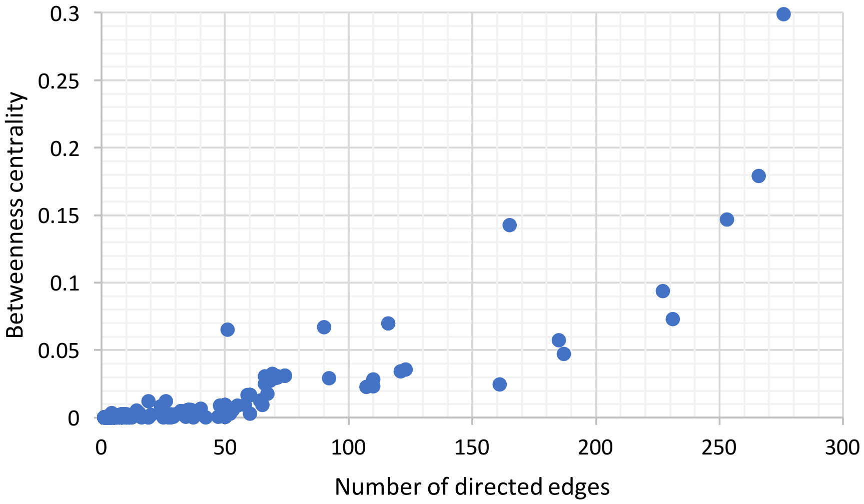


Figure 4.23: Illustrations of the regulatory network characteristics in *A. thaliana* dataset. The (a) node degree distribution and (b) shortest path length distribution of the regulatory network shown as independent plots.

An analysis of shortest path length was conducted for measuring the average number of steps along the shortest paths in the network. The shortest path length data demonstrated that ~230 edges have path length equal to 3.5, whereas relatively smaller number of edges displayed path lengths of 2-2.5. The network also produced edges possessing path lengths of 3 with frequency > 50 and < 100. These results indicate that approximately 2 – 4 steps are required for traversal between the nodes. BC and CC analysis of the nodes in the LPI-PPI dataset can be observed from the Figure 6.1 The network produced BC values ranging between 0.0 and 0.95 (Figure 4.24a) having an average BC of 0.475. It can also be seen from the graph that 2 nodes displayed BC above 0.25 with BC of 0.28 and 0.94 exhibiting their central position in the network as compared to other nodes. CC analysis of the nodes displayed large cluster of nodes having CC between 0.25 and 0.55 (Figure 4.24b). As compared to CC, the BC values were much lower showing lower centrality. This demonstrates that the nodes possess less control over the other nodes and is comparatively have smaller number of inter-connections. The clustering of nodes is directly correlated with the correlation values obtained from the co-expression data. The correlation values indicate greater percentage of protein-coding genes possess highly similar expression values, the CC values directly reflects the correlations. The smaller the CC value, the higher is the correlation, following an inverse proportionality.

a



b

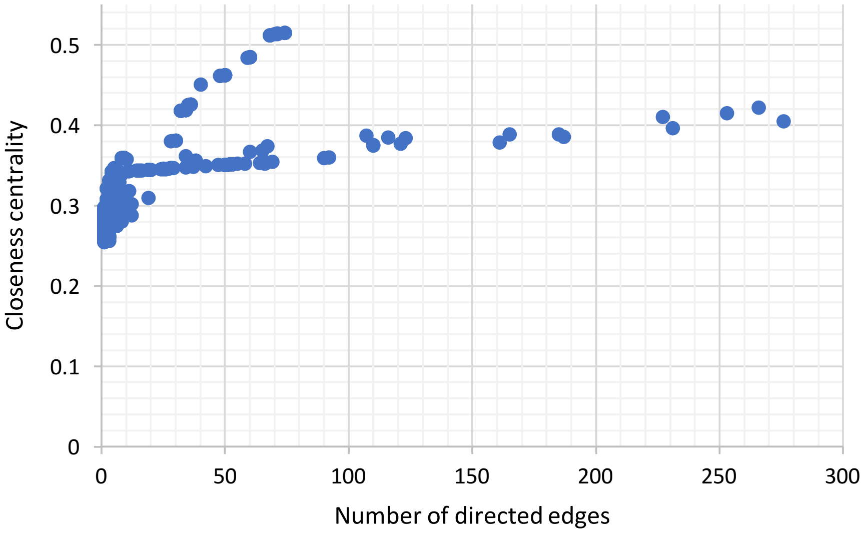


Figure 4.24: Illustration of (a) betweenness centrality and (b) closeness centrality in ATH dataset.

These results indicate that the co-expression regulatory network is similar to many biological networks which is well characterized by co-expression regulatory principles which distinguishes it from randomly generated networks (Nacher and Akutsu, 2007).

4.3 Case study 2: *Z. mays* inbred line B73 RNA-seq dataset

4.3.1 Identification of lncRNA sequences

Prediction performance of the features extracted from the *Z. mays* B73 RNA-seq dataset was tested on the test set sequences. The test set contains 5022 transcript sequences. Out of 5022 sequences, 50% of these are protein-coding (2511 sequences) and remaining 50% are lncRNA (2511 sequences). The training set consists of 13758 sequences, out of which 50% are protein-coding (6879 sequences) and remaining 50% are lncRNA (6879 sequences). A Random Forest model was trained using iRF classifier. 400 RF were generated to predict the class for each of the test sequence. The 6879 lncRNA sequences included in the training set were extracted from the Refseq reference dataset. Whereas the 6879 protein-coding sequences were extracted from the RNA-seq dataset.

Using the parameters mentioned in Section 2.8.1, 13758 sequences were trained using iRF classifier with 400 RFs. Prediction on 5022 test set sequences reveals an overall accuracy of

89.05%, sensitivity of 90.5%, specificity of 90.84%, F1-score of 88.84%, PPV of 90.5%, NPV of 87.69% and MCC of 0.78.

4.3.2 Performance benchmarking results on Z. mays lncRNA sequences

A 10-Fold CV analysis was performed on *Z. mays* B73 RNA-seq dataset to evaluate the prediction accuracy of the framework against the known CPC tools. 13758 training set sequences and 5022 test set sequences were used performing cross validation analysis. These sequences were concatenated to generate 18780 transcript sequences out of which 9390 were protein-coding sequences and remaining 9390 were lncRNA transcript sequences.

For creating the folds, 10% of the sequences were used for creating each fold which resulted in 1878 test set sequences and 16902 training set sequences. Prediction accuracy of each fold was benchmarked on the framework, PLEK, CPAT, lncScore and CPC2 tools. Since, lncScore failed to generate the prediction results, the performance benchmarking results of the framework were compared against PLEK, CPAT and CPC2 on the non-shuffled and the shuffled datasets. Therefore, lncScore has been excluded from the analysis. Based on the feature selection results obtained from LiRF-FS analysis, 31F feature set was used for performing benchmarking analysis.

Results from the 10-Fold CV show that the prediction accuracy of the framework with 31 features shows superior performance when compared against PLEK, CPAT and CPC2 on all the folds (Figure 4.25). In the D1 dataset (Figure 4.25a), Prediction accuracies of CPAT show 2.79% difference on the first fold. On the second fold, this difference increases to 3.05%. Whereas prediction accuracy difference between the framework and PLEK is comparatively much higher with an average difference of 11.58% against the framework from Folds-1 to 7. However, folds-8 to 10 displayed smaller differences in accuracy with an average of 2.32%.

In fold-8, the accuracy of the framework decreases from 97.55% to 93.23% whereas for CPAT, the accuracy also decreases from 95.6% to 88.37%. PLEK, however, shows an increase in the accuracy from 85.98% to 90.33%. Folds-9 and 10 do not show significant changes for the framework and PLEK, but the accuracy for CPAT decreases further to 80.32%. CPC2, on the other hand, exhibited lowest prediction accuracies in folds-1 to 7 with a mean value of 47.1%. The accuracies increased to 68.63% and 70.64% in folds 9 and 10 with accuracy differences of 51.14%, 37.06% and 22.42% against the framework.

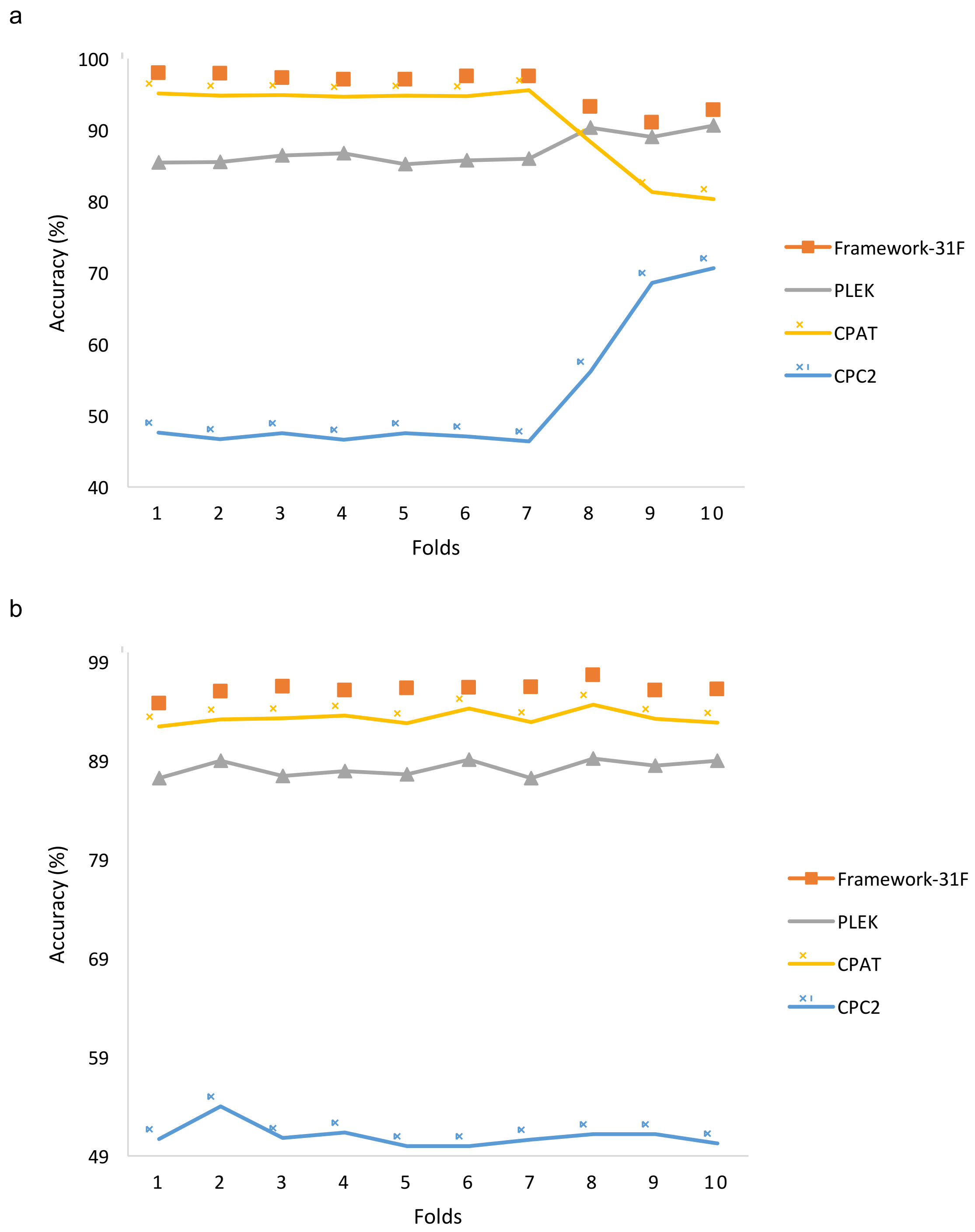
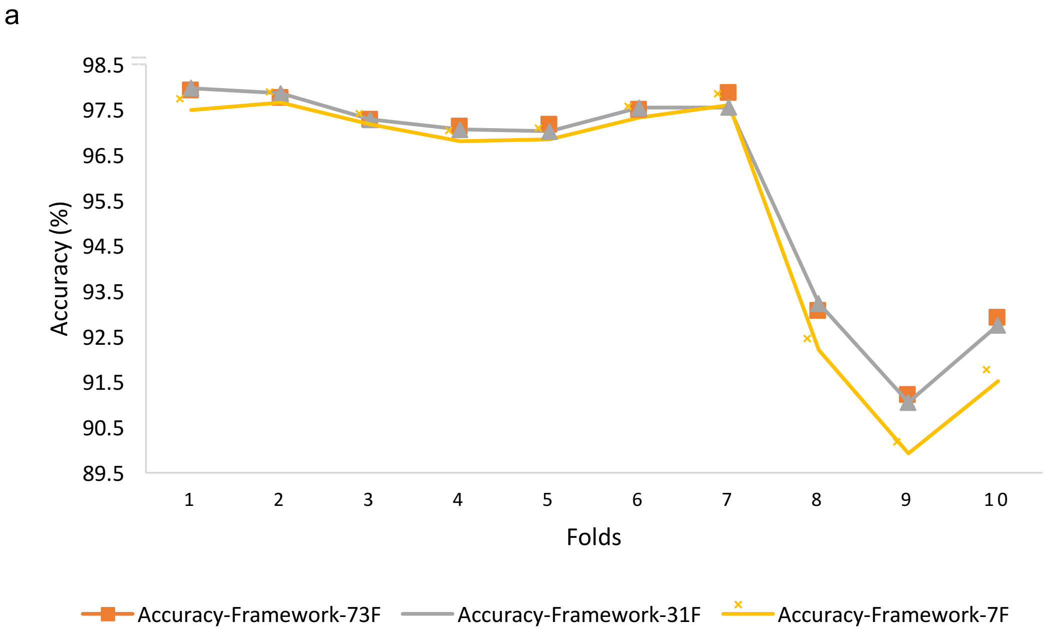


Figure 4.25: Performance benchmarking of the framework using 10-Fold CV with 31 features against CPC tools on *Z. mays* B73 annotated (a) D1 non-shuffled dataset, and (b) D2 shuffled dataset.

Results from D2 dataset (Figure 4.25b) also demonstrate a higher performance of the framework when compared with PLEK, CPAT and CPC2 tools. With randomized dataset, the lncRNA sequences in the last three folds were evenly dispersed which resulted in non-deviating values in all the folds. Accuracy results obtained from the framework generated a least prediction accuracy of 94.78% in the first fold, where CPAT, PLEK and CPC2 generated accuracies of 92.49%, 87.27% and 50.74% respectively. CPC2 displayed slight increase in the accuracy in fold-2 with a value of 54.03%. However, the value decreased again in successive folds exhiniting a mean accuracy of 50.69%. The accuracy of the framework increased in the subsequent folds to 97.65% in fold-8 where other tools also showed the similar pattern. In fold-8, a difference of 2.96% was observed between the framework and CPAT, whereas a difference of 8.44% was observed for the framework and PLEK. D1 dataset displayed an average accuracy of 95.93% for the framework, 87.12% for PLEK, 91.48% for CPAT and 52.51% for CPC2 whereas D2 dataset exhibited an average of 96.24% for the framework, 88.24% for PLEK, 93.33% for CPAT and 51.03% for CPC2.



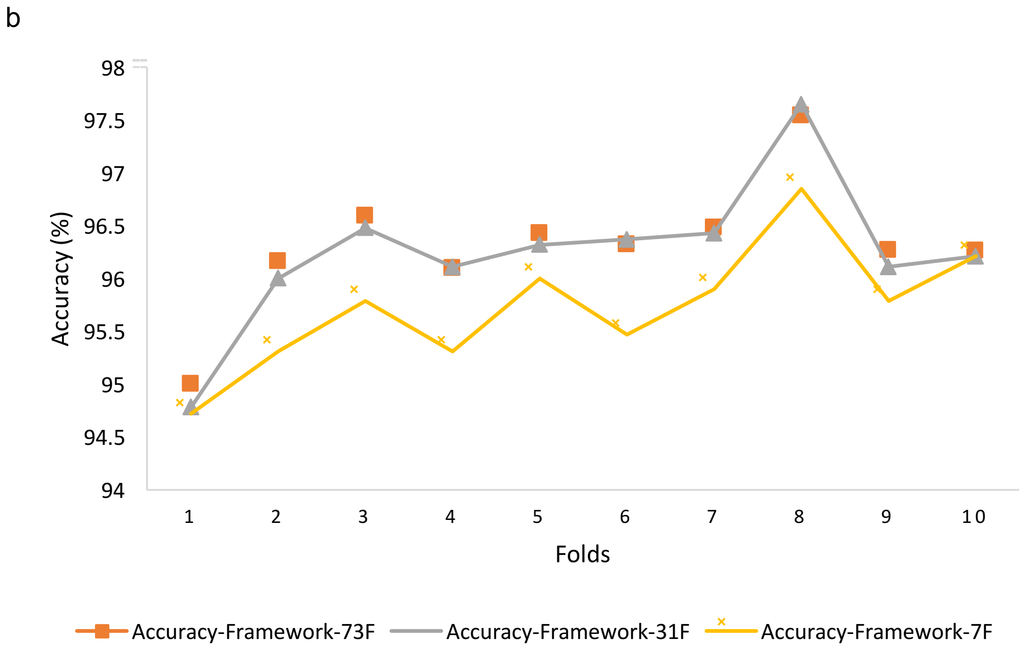


Figure 4.26: Performance comparison of optimal features from LiRF-FS on *Z. mays* dataset. (a) D1 non-randomized dataset, and (b) D1 randomized dataset.

Results from LiRF-FS were compared to evaluate the prediction performance of different feature sets in *Z. mays* dataset. Three feature sets, namely, 73F, 31F and 7F were compared amongst each other using 10-fold CV analysis on two datasets (Figure 4.26). Accuracy results demonstrate non-significant differences between the feature sets. However, minor deviation in accuracy can be observed between the accuracies obtained between these sets. Accuracy obtained from 31F feature set shows identical performance as observed by 73F whereas the accuracy from the 7F feature set shows marginal differences in some folds (Figure 4.26a). These differences can be noticed in folds-8, 9 and 10 where the difference of more than 1% occurs.

The shuffled D2 dataset, however, shows a slightly dissimilar trend in contrast to D1 (Figure 4.26b). The difference between the 31F and 7F ranges from 0.3% to 0.8% in folds-2 to 9. folds-1 and 10 does not display any change in the accuracy between the different feature sets.

Optimal features from the feature selection of 6-plants were implemented on *Z. mays* dataset. Resulting 7F and 31F feature set were extracted and labelled prediction was performed. Results from the *Z. mays* B73 RNA-seq dataset using 31F feature set suggests that the framework exhibited superior performance in predicting the lncRNA sequences in the non-shuffled and the shuffled datasets.

4.3.3 Repeated K-Fold Cross-Validation analysis of Z. mays lncRNA transcripts

As discussed in Section 4.2.8, a repeated 10-fold CV was performed on *Z. mays* Ensembl Genomes 39 AGPv4-annotated dataset to evaluate the predictive power and performance evaluation of the 31F feature set in the framework. Results from repeated 10-fold CV analysis (Figure 4.27) shows that the accuracy values fluctuated between 96.10% and 96.3% producing an average accuracy of 96.2%. The precision and recall values were comparatively higher generating values between 96.45% and 96.7%. The results clearly display that the framework identified the lncRNA sequences in *Z. mays* dataset with greater accuracy and precision in recognising true positive sequences.

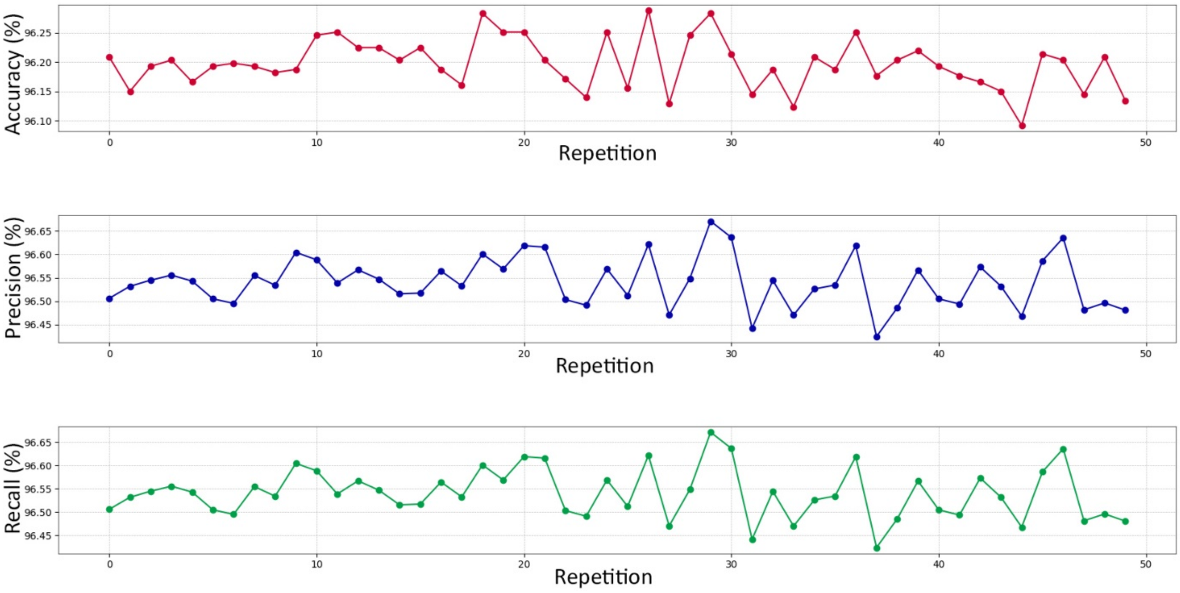


Figure 4.27: A repeated 10-Fold CV plots of Accuracy, Precision and Recall on *Z. mays* dataset.

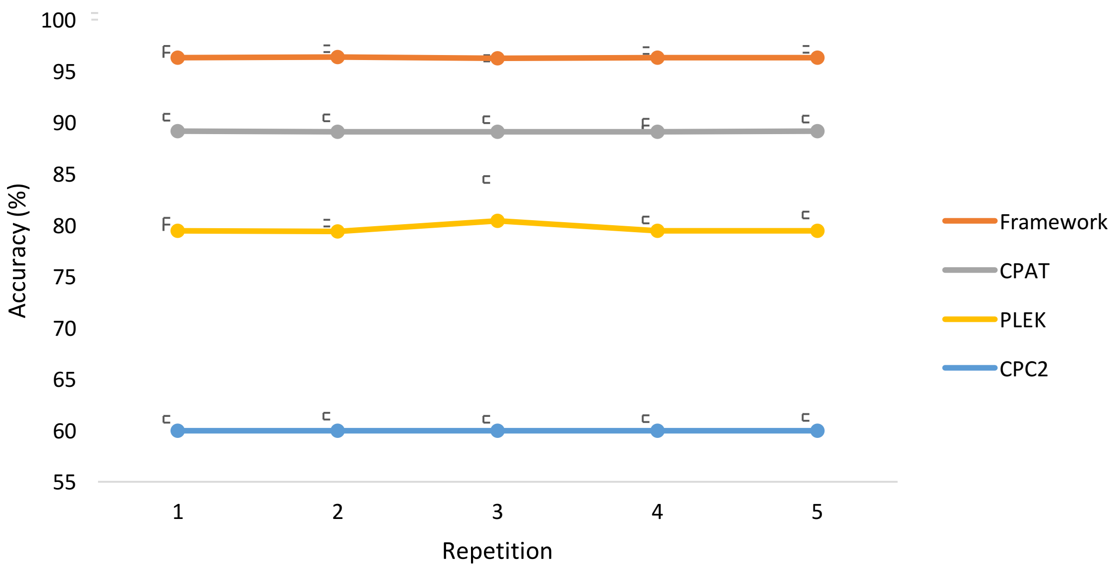


Figure 4.28: Performance benchmarking of the framework using repeated k-fold CV against CPC tools on *Z. mays* lncRNA annotated dataset. Mean accuracy values are plotted with error bars representing standard deviation.

Results from repeated 10-fold CV performed on Ensembl annotated sequences in *Z. mays* dataset shows relatively higher prediction accuracies of 96.27% by the framework. Whereas comparatively lower and stable accuracy values for CPAT were observed with maximum SD 1.06 thereby displaying an average accuracy of 89.07%. PLEK displayed a mean accuracy of 79.58% with an accuracy difference of 16.69% against the framework. Noticebly, in the third repetition, a larger SD was generated with accuracy values in the range 89.06±3.73%. CPC2 exhibited lowest accuracy values among all the tools under comparison where an average accuracy of 59.92% was produced. An accuracy difference of 36.35% was observed between the framework and CPC2 thereby presenting highest accuracy differences and greater precision in identification of lncRNAs in *Z. mays* dataset. Comparison with lncScore was excluded as lncScore failed to generate the prediction results. Comparative analysis of repeated k-fold CV with 50 iterations (Figure 4.27) and benchmarking performance (Figure 4.28) demonstrates that the accuracy of the framework fluctuates insignificantly with minute SD around the mean accuracy value. This pattern has been observed for other CPC tools also which is represented by a straight line. This validation test clearly indicates that with random selection of data, the prediction accuracy does not drop below a specific threshold value. Therefore, this test also provides a reliable measure for evaluating the performance when compared with other tools.

4.3.4 Sub-classification analysis of Z. mays lncRNA transcript sequences

Using the PBC algorithm, sub-classification analysis of lncRNA annotated obtained from Ensembl Genomes 38 database (version AGPv4). 2511 lncRNA sequences were obtained from the annotated GTF file for sub-classification analysis. Results were obtained based on Rules-1 and 2 of the PBC. Results from the PBC have been presented in Table 4.7.

Rule-1 results show that equal number of SOE and SOI sequences were classified. Identical results can be observed in the AOE and AOI classes where equal number of sequences were classified. However, the number of sequences is much less than the sense-overlap class. Compared to sense and antisense-overlap classes, 909 sequences have been classified as ANT class. Sequences classified in the BDP class is much less than that observed than the ANT class. Sequences classified as INT have the highest proportion of lncRNA sequences among all other classes.

Rule-2 reveals a higher proportion of SOE and SOI classified sequences as compared to those obtained from Rule-1 by increase of 64 sequences. Whereas the same cannot be detected for the antisense classes (AOE and AOI). The show minor increase of 11 sequences. Sequences classified as ANT, INT and BDP does not differ as the rules are only applicable for the sense and antisense overlap classification.

Table 4.7: Sub-classification statistics of the *Z. mays* B73 lncRNA transcript sequences based on Rule-1 and Rule-2.

Rules	**SOE**	**SOI**	**AOE**	**AOI**	**ANT**	**INT**	**BDP**
Rule-1	55	55	13	13	909	1682	166
Rule-2	119	118	24	24	909	1682	166

It can be clearly observed that classification results obtained from the *A. thaliana* dataset does not show similar pattern as observed in *Z. mays* classification. Results from the *A. thaliana* classification indicated higher proportion of INT sequences whereas the proportion of sequences classified as INT sequences is much higher for *Z. mays* data.

Results from the PBC approach were compared against the annotated lncRNA sequences from the *Z. mays* Ensembl Genomes 38 AGPv4 data (Figure 4.29). The database consists of 2551 lncRNA sequences with varying sequence lengths (i.e. sequence length < 200 bp and ≥ 200 bp). All 2551 sequences are annotated as lincRNA sequences. Individual chromosomal comparison shows a higher proportion of matching sequences in all the chromosomes. Chromosomes 1, 2, 4, 6, 7 and 9 shows significantly higher number of matching lincRNA sequences with an average matching percentage of 93.71% using Rule-1 and 2. Chromosomes 3, 5, 8 and 10 exhibits an average match of 85.34%. Overall the matching proportion of lincRNA sequences between the PBC analysis and Ensembl annotated results show an overall accuracy of 90.86%.

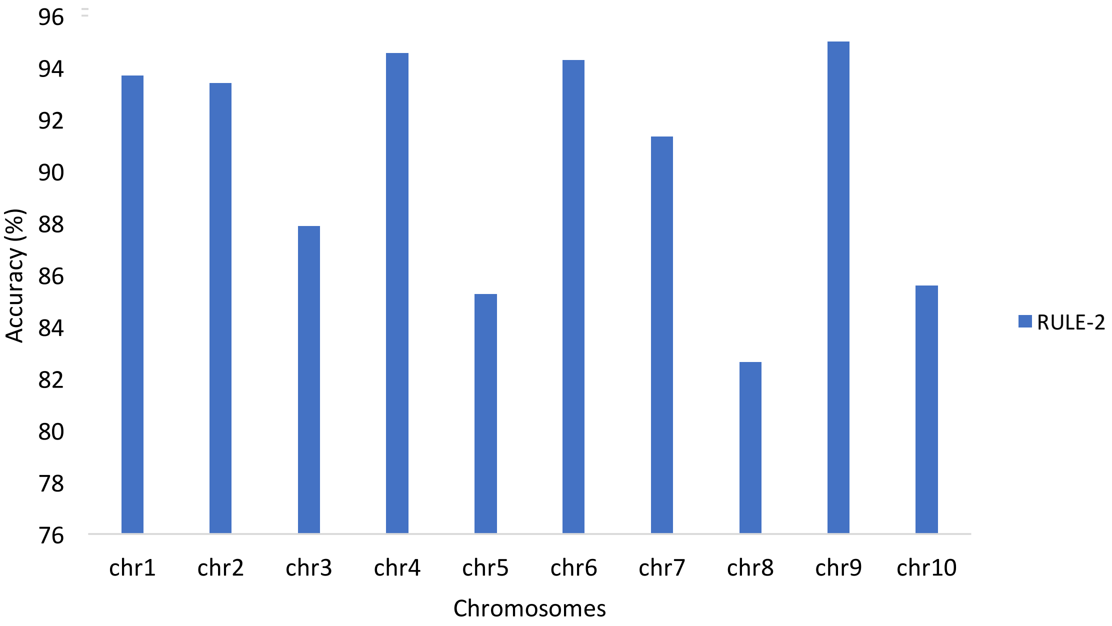


Figure 4.29: Prediction performance of PBC on *Z. mays* AGPv4 data for identification of lincRNAs in individual chromosomes using Rule-2.

Results from the PBC analysis validate the experimental results with higher accuracy in the *Z. mays* dataset. Prediction accuracy from the analysis also indicates that the PBC method is efficient and robust to the variations in the sequence data obtained from different plant species. Results demonstrated consistently higher accuracy in both the annotated lncRNA sequences obtained from public databases.

4.3.5 Function prediction based on co-expression data

2511 lncRNA sequences obtained from the Ensembl Genomes 38 AGPv4 database were extracted for BMRF function prediction analysis. Based on correlation of lncRNA FPKM values with the protein-coding genes, 2183 sequences were retained having non-zero FPKM values in ≥ 70% of sample pairs with PCC ≥ 0.9 and PCC ≤ -0.9. The LPCS matrix consisted of 378 lncRNA sequences having 38260 correlations with 5940 protein-coding genes.

PPI matrix consisted of 13940 protein-protein interaction pairs. BMRF analysis of the correlated lncRNA sequences produced 353 GOTerms which were associated with 376 lncRNAs. By applying a probability cutoff of 0.8, 287 lncRNAs displayed high association probability with 264 GOTerms.

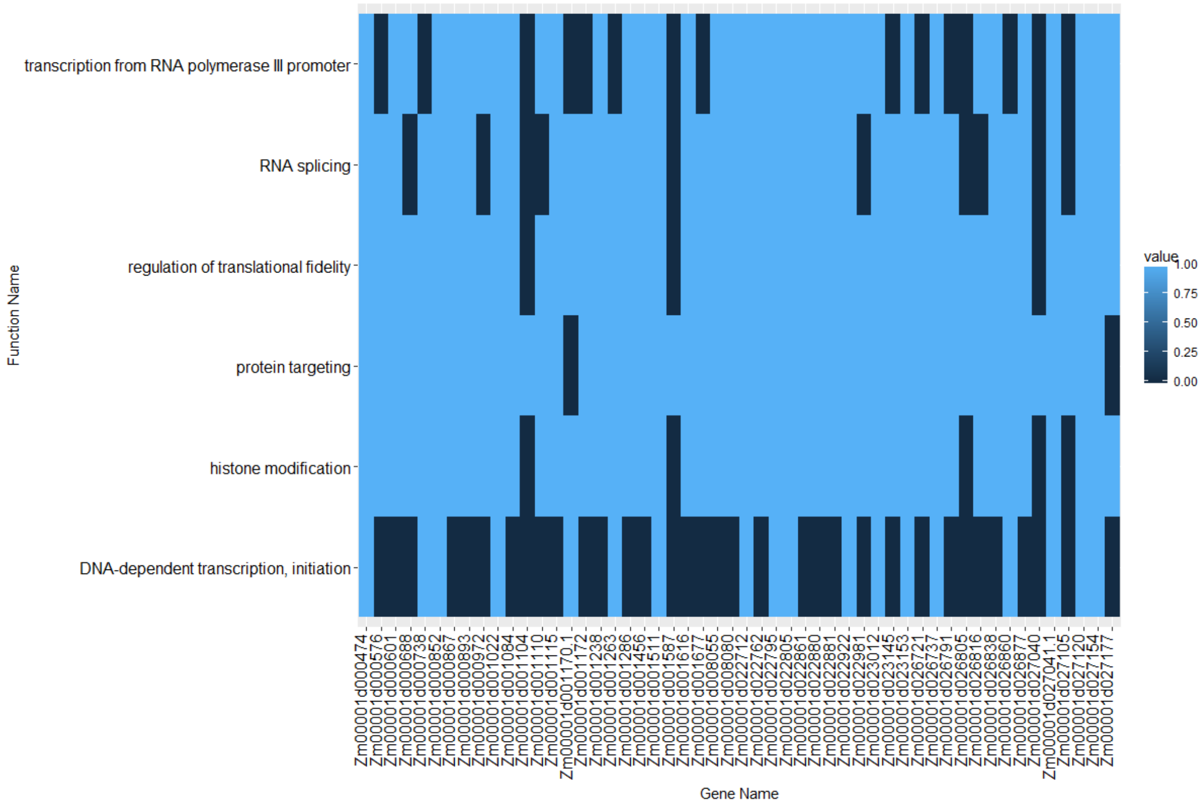


Figure 4.30: Heatmap of lncRNA sequences associated with experimentally determines functions in *Z. mays* B73 dataset. X-axis shows the lncRNA genes and the y-axis shows the function name. “Light blue” colour represents association and “dark blue” colour represents non-association.

Heatmap analysis was performed for detecting lncRNA sequence association with experimentally-determined functions in *Z. mays* B73 dataset (Figure 4.30). Results demonstrate that 50 sequences have been found to play primary roles in protein targeting and in the regulation of translational fidelity (Table 4.8). 47 genes were found to be associated with “histone modification”, whereas, 42 genes were found to be involved in “RNA splicing”. 37

genes were annotated with “transcription from RNA polymerase III promoter”, whereas 18 were found to have association with “DNA-dependent transcription, initiation”.

Table 4.8: List of non-DE genes associated with experimentally-determined molecular functions.

Molecular function	**Number of associated lncRNA genes**
Protein targeting	50
Regulation of translational fidelity	49
Histone modification	47
RNA splicing	42
Transcription from RNA polymerase III promoter	37
DNA-dependent transcription, initiation	18

4.3.6 Function prediction of DE sequences based on co-expression data

Results from the DE analysis was performed on the *Z. mays* B73 dataset. Significantly expressed sequences with q-values ≤ 0.05 were extracted in ≥ 4 sample pairs. 103 lncRNA sequences were found to be DE whereas 7631 protein-coding sequences were DE from a total of 7989 DE transcript sequences.

LCPS and PPI matrices were constructed to obtain results of the function prediction. FPKM values of the 103 lncRNA sequences were compared against 7631 protein-coding sequences. By applying a correlation cutoff of ≥ 0.8 and ≤ -0.8, 74710 positive correlations and 22733 negative correlations were obtained. Overall, 93 lncRNA sequences were found to be correlated with 6140 protein-coding sequences. This generated a total of 97443 correlations.

A PPI matrix was generated by obtaining the DE protein-coding sequences and obtaining significant interactions between other protein-coding genes. From the STRING database, 42 unique protein-protein interactions were obtained.

BMRF function prediction analysis was performed on 93 lncRNA sequences to obtain molecular function association. Based on a probability cutoff of 0.8, 93 lncRNA sequences were annotated with three GO function types: molecular function, biological process and cellular component. 10 GOTerms were found to be associated with 93 lncRNA sequences, which are: biosynthetic process, cellular amino acid metabolic process, dopamine neurotransmitter receptor activity, intracellular, oxidation-reduction process, pyridoxal phosphate binding, ribosome, ribosome biogenesis, structural constituent of ribosome and translation.

Heatmap analysis of the lncRNA-function association (Figure 4.31) shows that the majority of the lncRNAs were associated with 8 functions except association with “cellular amino acid metabolic process” and “dopamine neurotransmitter receptor activity”. Oxidation-reduction process consisted of much larger set of genes compared to other functions. The remaining 5 functions consisted of approximately 93 genes (Table 4.9).

Table 4.9: List genes associated with functions in DE *Z. mays* geneset.

Function	**Associated lncRNA genes**
dopamine neurotransmitter receptor activity	24
cellular amino acid metabolic process	14
oxidation-reduction process	57
biosynthetic process	4
pyridoxal phosphate binding	92
intracellular	93
ribosome	93
translation	93
ribosome biogenesis	93

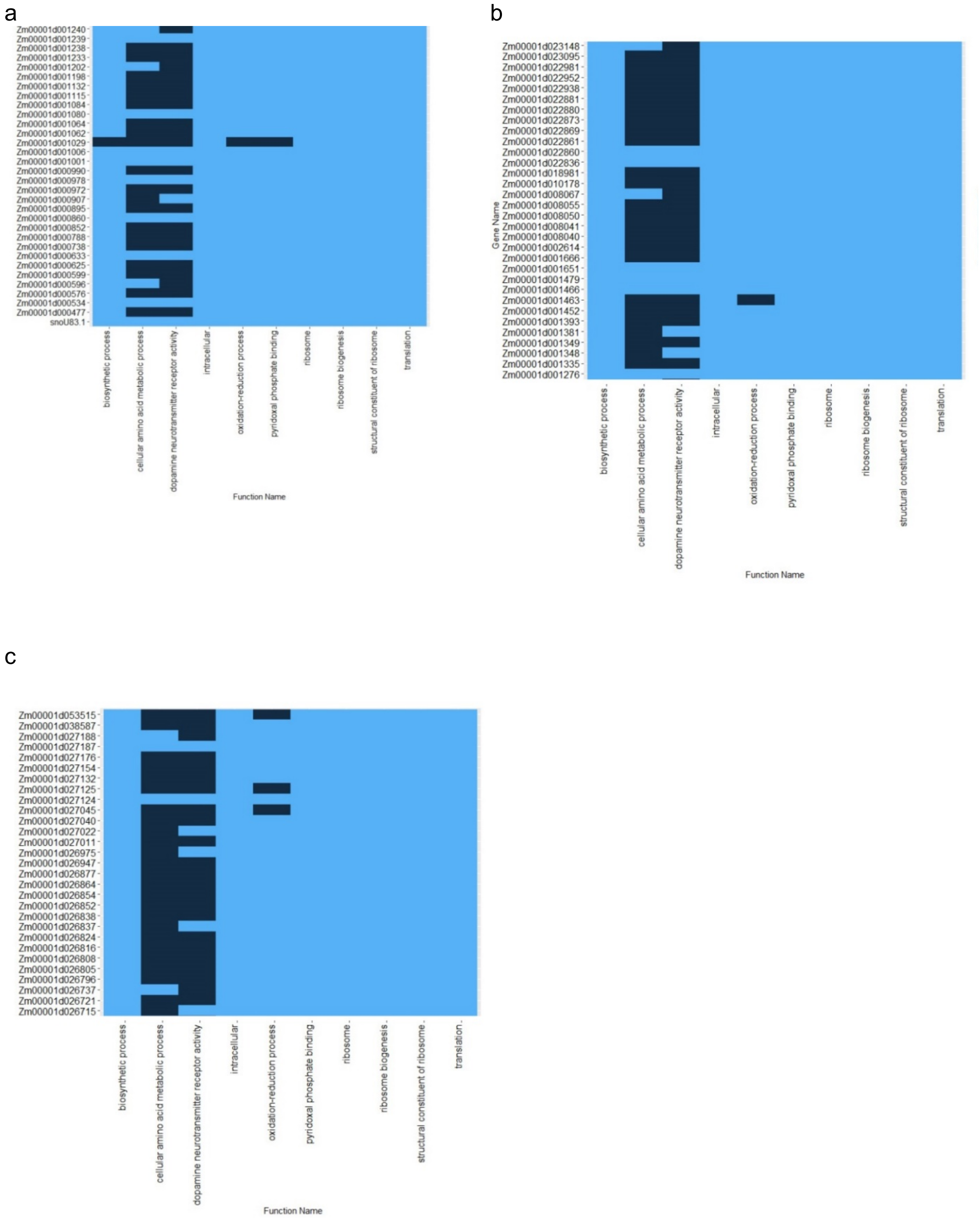
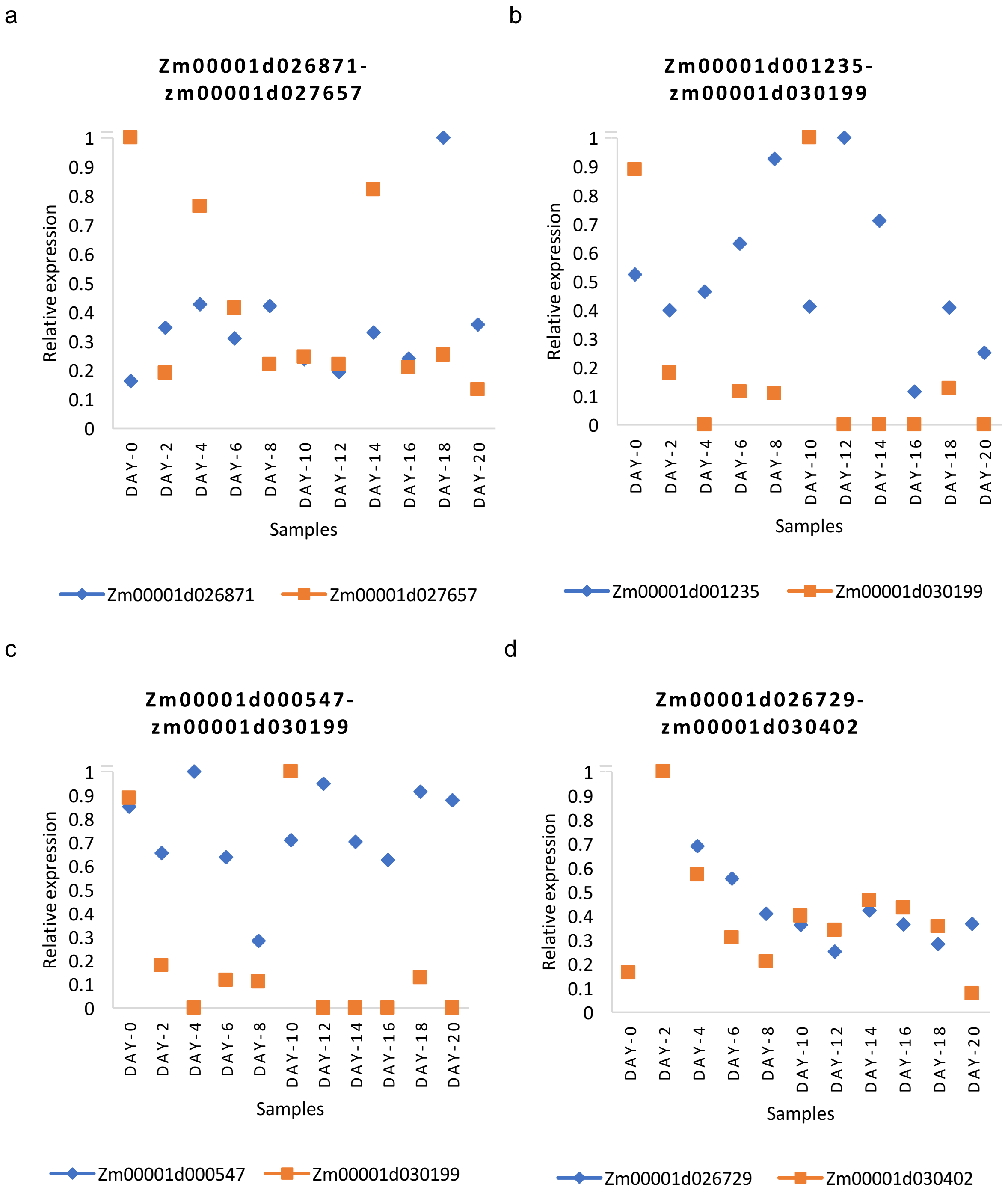


Figure 4.31: Heatmap of lncRNA sequences associated with functions in *Z. mays* B73 dataset. The complete heatmap is broken down to three maps (a – c). The genes are associated with molecular functions.

4.3.7 Function prediction of the lncRNA sequences based on NRLMF and BMRF analysis

For predicting the functions of the lncRNA sequences, a subset of 51 lncRNA transcripts and 400 protein-coding sequences were selected from a pool of 2511 known lncRNAs and 131496 protein-coding sequences. NRLMF analysis was conducted for obtaining physical interactions between lncRNA and protein-coding genes. A sequence similarity matrix was constructed between *Z. mays* and *H. sapiens* sequences generating lncRNA-lncRNA and protein-protein similarity matrices. From *H. sapiens*, 50 lncRNAs and 400 protein-coding sequences were selected. An adjacency matrix was formed between the lncRNA and protein-coding genes (i.e. a 100 × 800 matrix).

2762 interactions were obtained ranging between 0.9054 and 0.1591. Out of 2762, 280 novel interactions were obtained with scores ≥ 0.7. Correlation analysis of lncRNA and protein-coding co-expression data shows correlations ≥ -0.5 for lncRNA sequences (Figure 4.32). Figure 4.32 illustrates relative expression of the NRLMF-derived lncRNA (blue coloured) and protein-coding (orange coloured) genes predicted to have interactions with scores > 0.7. The relative expression of genes suggests that most of the protein-coding genes displayed variable expression when compared with lncRNAs during the growth phase. Protein-coding genes showed relatively lower expression whereas lncRNAs exhibited higher expression during Days-2 to 8 and Days-12 to 20 for Zm00001d001235, Zm00001d000547 and Zm00001d027131 (Figure 4.32b, c and f). Zm00001d026729 (lncRNA) Zm00001d030402 (protein) showed similar profile generating PCC of 0.53. Results advocate that coexpression of lncRNA and protein-coding genes were particularly observed on Days-0, 2, 8 and 10 with significantly higher expression of both lncRNA and proteins.


a
Zm00001d026871-zm00001d027657
Relative expression
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
DAY-0
DAY-2
DAY-4
DAY-6
DAY-8
DAY-10
DAY-12
DAY-14
DAY-16
DAY-18
DAY-20
Samples
Zm00001d026871
Zm00001d027657
b
Zm00001d001235-zm00001d030199
Relative expression
Samples
Zm00001d001235
Zm00001d030199
c
Zm00001d000547-zm00001d030199
Relative expression
Samples
Zm00001d000547
Zm00001d030199
d
Zm00001d026729-zm00001d030402
Relative expression
Samples
Zm00001d026729
Zm00001d030402

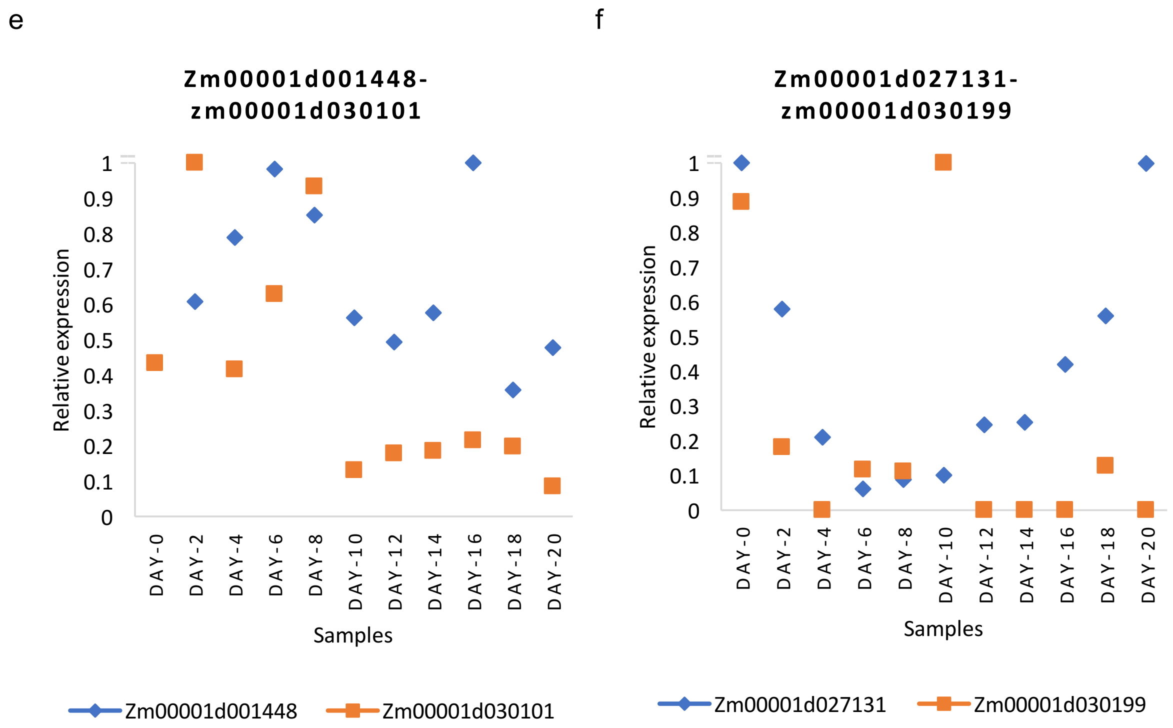


Figure 4.32: Relative gene expression of lncRNA and protein-coding genes predicted to have interactions in *Z. mays* B73 dataset.

As discussed in Section 4.2.14, sequence similarity in the scatter plot was computed by matching the one *Z. mays* sequence against n *H. sapiens* sequences (n=400) (Figure 4.33). The highest matching *H. sapiens* sequence was selected as having highest similarity. Scatter plots of the sequence similarities for *Z. mays* lncRNA sequences range between 50–58% whereas for protein-coding genes, the similarities lie between 52–57%. As mentioned previously, the computation of LPI pairs was performed on a smaller subset of data for evaluation and demonstration of function prediction approach using NRLMF, a larger set of sequences should yield higher sequence similarities.

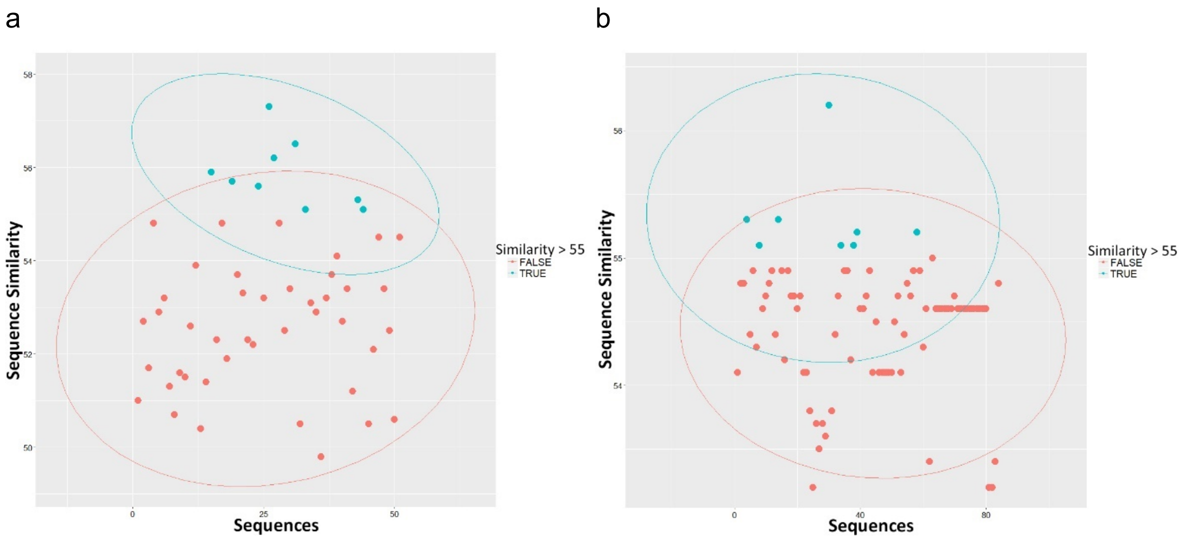


Figure 4.33: Scatter plots of sequence similarities in *Z. mays* dataset. (a) lncRNA-lncRNA SSM plot, and (b) protein-protein SSM plot.

Results from the BMRF analysis of the LPI interaction data obtained from NRLMF analysis produced function association with “ATP binding” for LPIs with scores ≥ 0.7. Filtering the LPIs with scores ≥ 0.5 generated in association of cellular component (CC) and biological process (BP) functions to the lncRNA genes. These were found to be associated with “integral to membrane” and “oxidation-reduction process” functions.

Another analysis was conducted with dissimilar protein-coding sequences to obtain additional functions of lncRNA sequences. NRLMF analysis generated 182 novel LPIs with scores ≥ 0.7 where 8 proteins were found to interact with 51 lncRNAs. 9191 PPIs were obtained from the protein-coding genes in LPIs. BMRF analysis generated 182 gene-GOTerm association with 51 genes. However, all 51 genes were found to be associated with “ATP binding” molecular function having probability values of 1.0. Out of 51 lncRNAs, Zm00001d026838 and Zm00001d001466 were found to be DE.

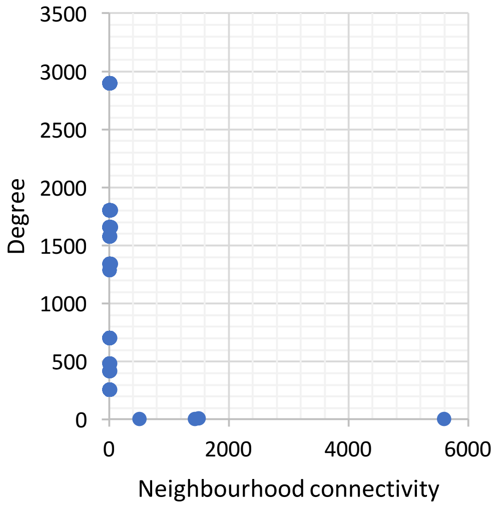
The protein-coding genes predicted to have stronger interaction with the lncRNA sequences are involved in the regulation of protein metabolic process, post-translational protein modification, transcription regulator activity, intracellular functions, DNA-dependent transcription initiation, ligand-activated sequence-specific DNA binding RNA polymerase II transcription factor activity, ribosome, structural constituent of ribosome, translation and response to freezing. A comprehensive review on the analysis of lncRNA cellular mechanisms/functions provide an insight into the type of lncRNA-protein interactions and regulatory functions (Signal, Gloss and Dinger, 2016). These functions associated with the protein-coding genes imply similar functions of lncRNAs with regulatory mechanisms in transcriptional and translation activities.

4.3.8 LPI-PPI network analysis of Z. mays B73 dataset

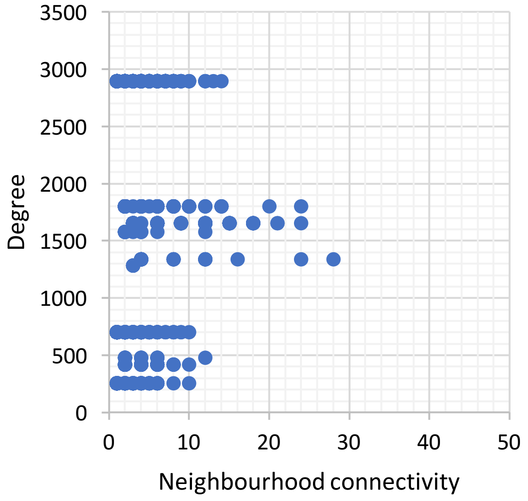
An examination of the node degrees in the LPI-PPI regulatory network shows the power-law distribution with a slope of -0.827 and R^2 = 0.537 (Figure 4.34a). Correlation analysis of the number of nodes in the network exhibited PCC of 0.964. The node degree distribution shows that 250–3000 nodes were found to be connected to 2–30 neighbouring nodes. Higher PCC and R^2 value indicates that the data fitted across the power-law slope shows higher goodness of fit. From the Figure 4.34a, 1492 nodes were found to be connected to ~9 nodes, whereas 5589 and 1427 nodes were found to be connected to ~1–2 neighbouring nodes. The gene regulatory network demonstrates that the network consists of single connections. However, fewer nodes with multiple connectivity were also identified which could potentially regulate the activity of other nodes/genes and could thus serve as important regulators in the biological process.

Betweenness Centrality (BC) and Closeness Centrality (CC) analysis (Figure 4.35) of the data demonstrated higher degree of neighbours displayed BC between 0.0 and 0.95 whereas the CC values ranged between 0.25 and 0.5. The CC values clearly demonstrate large percentage of shortest paths between the nodes and all the other nodes in the network. The centrality measures obtained in *Z. mays* are identical to those centrality values obtained in *A. thaliana* LPI-PPI data. The distribution of the shortest path lengths (Figure 4.34b) reveals path length of 2.5 for majority of nodes connected in the network. This shows that the genes are highly interconnected regulating the expression of other genes. This shows that the lncRNA and protein-coding sequences regulate each other with a higher degree. The BC and CC profiles produced for the *Z. mays* dataset also displayed similarity and indicates greater similarity to many biological networks.

a



Degree
3500
3000
2500
2000
1500
1000
500
0
0
2000
4000
6000
Neighbourhood connectivity



Degree
3500
3000
2500
2000
1500
1000
500
0
0
10
20
30
40
50
Neighbourhood connectivity

b

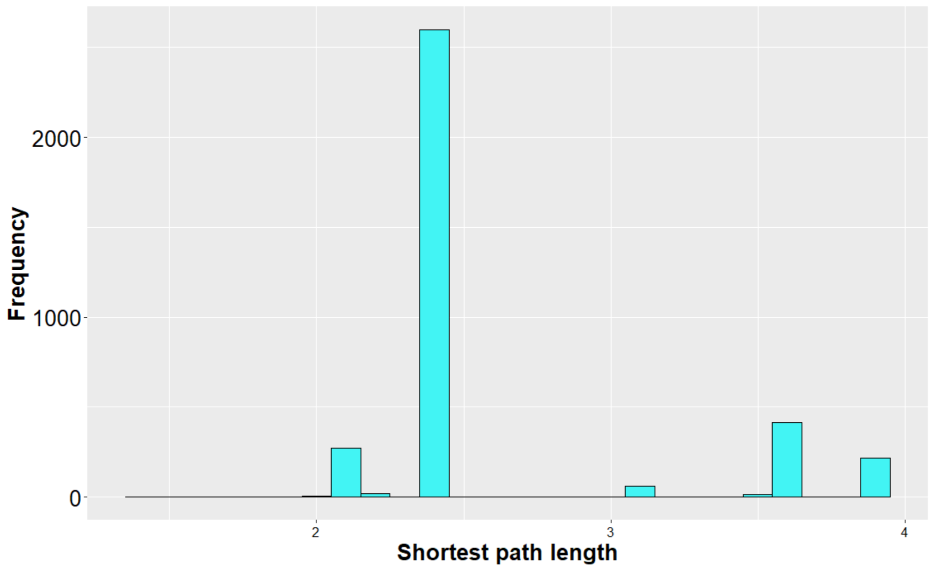


Figure 4.34: Illustrations of the regulatory network characteristics in *Z. mays* data. The (a) degree distribution and (b) shortest path length distribution of the regulatory network shown as independent plots.

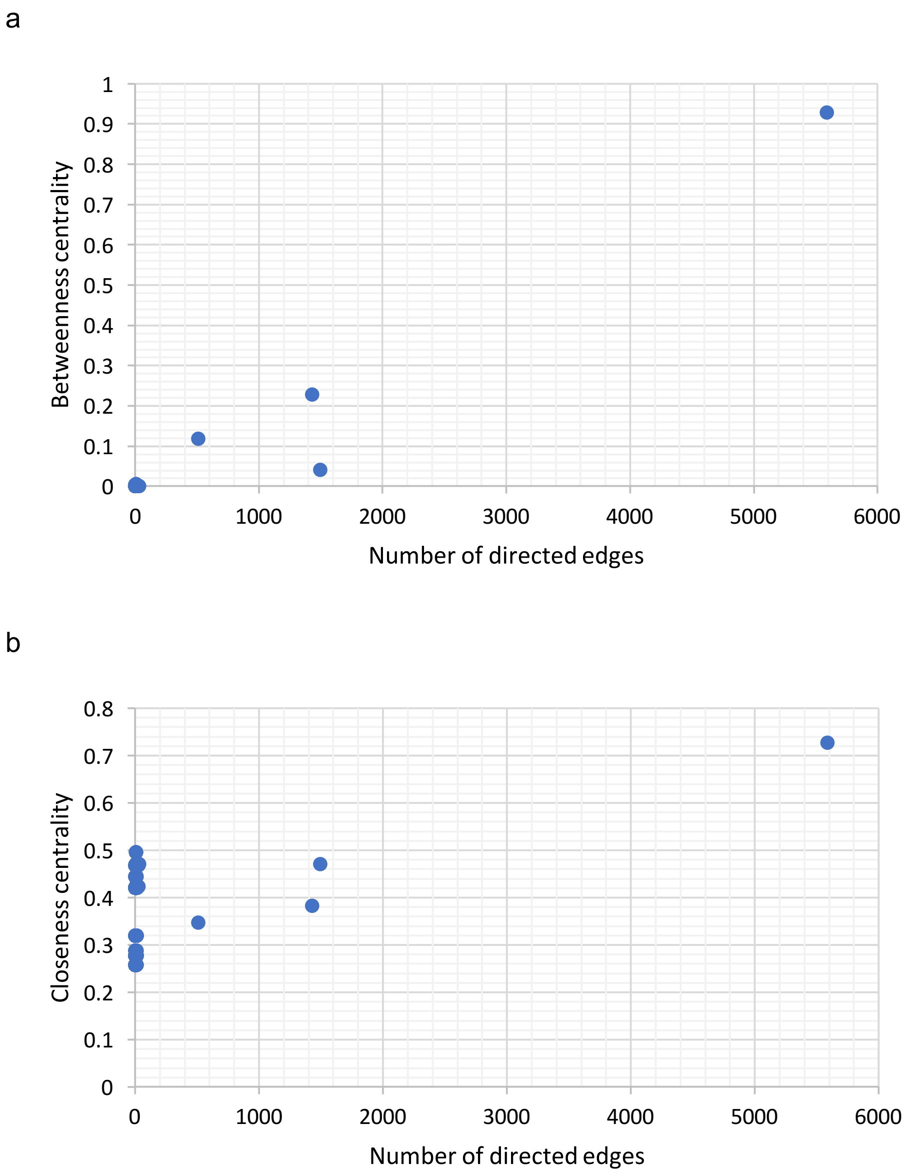


Figure 4.35: Illustration of (a) betweenness centrality and (b) closeness centrality in *Z. mays* data.

The centrality measures obtained from the LPI-PPI network demonstrates the closeness of the nodes and higher degree of connectivity between them. The high degree of connectivity indicates that a lncRNA gene can regulate more than one protein-coding genes or vice versa. The results obtained from the LPI-PPI network analysis clearly exhibits the non-randomness of the network connectivity and higher similarity to biological networks.

4.4 Summary

In this chapter, two case studies have been conducted. The first case study involved identification of novel flowering genes from the RNA-seq *A. thaliana* apical-shoot dataset based on significant expression of genes during the flowering transition phase. The case study also involved demonstration of the use of computational framework in accurate classification and function prediction of lncRNA genes from the consensus transcript sequences. The second case study was focused on identification and function prediction of lncRNA genes from B73 *Z. mays* dataset. The performance of the framework was evaluated and compared against other state-of-the-art tools with varying lncRNA transcripts obtained from PLncDB and Ensembl databases. Results of the performance evaluation of the framework have been presented on lncRNA prediction and sub-classification. Results from the function prediction of lncRNA genes in Arabidopsis and Maize species were discussed. Filtration and validation of the function prediction results were comprehensively presented.

CHAPTER 5: VISUALISATION

5.1 Introduction

This chapter presents the results of a Javascript-based web application for visualisation of lncRNA sequences derived from RNA-seq datasets. It discusses the results of the visualisation and intermediate results obtained when producing the input files.

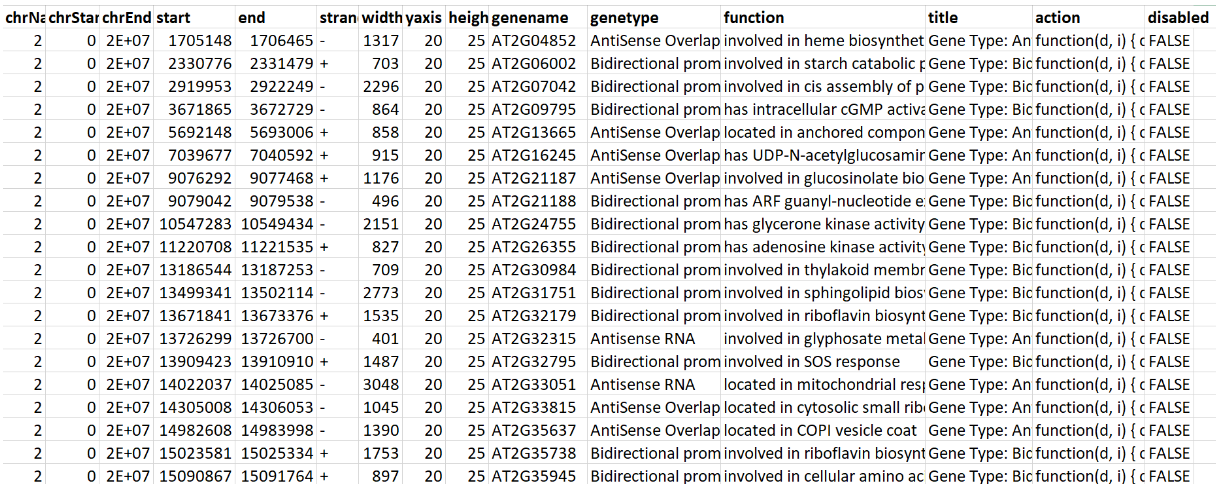
5.2 Visualisation of lncRNA sequences from RNA-seq datasets

For development of visualisation application for lncRNA sequences derived from RNA-seq datasets, lncRNA sequences were annotated using PBC and BMRF approaches. Table 5.1 shows the resulting lncRNA annotation file. Results from the analysis consists of transcript ID, chromosome number, start and end positions of lncRNA sequence, gene name, strand (sense ('+') or antisense ('-')), chromosome length, lncRNA sub-class (i.e. gene type) and gene function.

Table 5.1: lncRNA annotation file from PBC sub-classification and BMRF approach.

Transcript ID	chr	start	end	Gene name	strand	length	Gene type	Function
AT1G01448	1	163431	166239	AT1G01448	+	30427671	Bidirectional promoter	involved in glycoprotein catabolic process
AT1G02952	1	665354	666367	AT1G02952	+	30427671	AntiSense Overlap Exonic	located in mitochondrial respiratory chain
AT1G04295	1	1147781	1148435	AT1G04295	+	30427671	Bidirectional promoter	involved in negative regulation of translational initiation
AT1G07119	1	2184347	2186539	AT1G07119	+	30427671	AntiSense Overlap Exonic	required for nuclear-transcribed mRNA catabolic process
AT1G07728	1	2395461	2397345	AT1G07728	+	30427671	AntiSense Overlap Intronic	has L-tyrosine:2-oxoglutarate aminotransferase activity
AT1G09421	1	3038631	3039326	AT1G09421	+	30427671	AntiSense Overlap Intronic	located in intracellular membrane-bounded organelle

Annotated results were used an input for producing CSV file. The “format_annotation.py” script was used for producing D3-specific sequence annotation file for each chromosome (Figure 5.1). These individual files were used by the “index.html” file for producing the visualisation. As described in Chapter-2 Table 2.11, the visualisation was constructed using D3.js Javascript library to produce a user interactive “index.html” webpage.



chrNa	chrStar	chrEnd	start	end	stran	width	yaxis	heigh	genename	genetype	function	title	action	disabled
2	0	2E+07	1705148	1706465	-	1317	20	25	AT2G04852	AntiSense Overlap	involved in heme biosynthet	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	2330776	2331479	+	703	20	25	AT2G06002	Bidirectional prom	involved in starch catabolic p	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	2919953	2922249	-	2296	20	25	AT2G07042	Bidirectional prom	involved in cis assembly of p	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	3671865	3672729	-	864	20	25	AT2G09795	Bidirectional prom	has intracellular cGMP activa	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	5692148	5693006	+	858	20	25	AT2G13665	AntiSense Overlap	located in anchored compon	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	7039677	7040592	+	915	20	25	AT2G16245	AntiSense Overlap	has UDP-N-acetylglucosamir	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	9076292	9077468	+	1176	20	25	AT2G21187	AntiSense Overlap	involved in glucosinolate bio	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	9079042	9079538	-	496	20	25	AT2G21188	Bidirectional prom	has ARF guanyl-nucleotide e	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	10547283	10549434	-	2151	20	25	AT2G24755	Bidirectional prom	has glycerone kinase activity	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	11220708	11221535	+	827	20	25	AT2G26355	Bidirectional prom	has adenosine kinase activity	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	13186544	13187253	-	709	20	25	AT2G30984	Bidirectional prom	involved in thylakoid membr	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	13499341	13502114	-	2773	20	25	AT2G31751	Bidirectional prom	involved in sphingolipid bios	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	13671841	13673376	+	1535	20	25	AT2G32179	Bidirectional prom	involved in riboflavin biosynt	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	13726299	13726700	-	401	20	25	AT2G32315	Antisense RNA	involved in glyphosate metal	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	13909423	13910910	+	1487	20	25	AT2G32795	Bidirectional prom	involved in SOS response	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	14022037	14025085	-	3048	20	25	AT2G33051	Antisense RNA	located in mitochondrial res	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	14305008	14306053	-	1045	20	25	AT2G33815	AntiSense Overlap	located in cytosolic small rib	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	14982608	14983998	-	1390	20	25	AT2G35637	AntiSense Overlap	located in COPI vesicle coat	Gene Type: An	function(d, i) { c	FALSE
2	0	2E+07	15023581	15025334	+	1753	20	25	AT2G35738	Bidirectional prom	involved in riboflavin biosynt	Gene Type: Bic	function(d, i) { c	FALSE
2	0	2E+07	15090867	15091764	+	897	20	25	AT2G35945	Bidirectional prom	involved in cellular amino ac	Gene Type: Bic	function(d, i) { c	FALSE

Figure 5.1: D3-specific lncRNA annotation data for visualisation.

The results of the graphical visualisation can be observed in Figures 5.2, 5.3, 5.4 and 5.5 which demonstrates the visualisation of several lncRNA sub-classes. Figure 5.2 shows the overall view or zoomed-out chromosomal views of chromosomes-1 and 4. This view provides an overall picture of the several lncRNA sequences and their positions on the chromosomes. It can be observed that the lncRNA sequences are scattered across the chromosome. The picture shows relative alignment of lncRNA sequences which displays:

(1) Intergenic lncRNA represented by brilliant arctic blue,

(2) Antisense lncRNA represented by blue violet colour,

(3) Antisense overlap exonic represented by burly wood colour,

(4) Antisense overlap intronic represented by chartreuse colour,

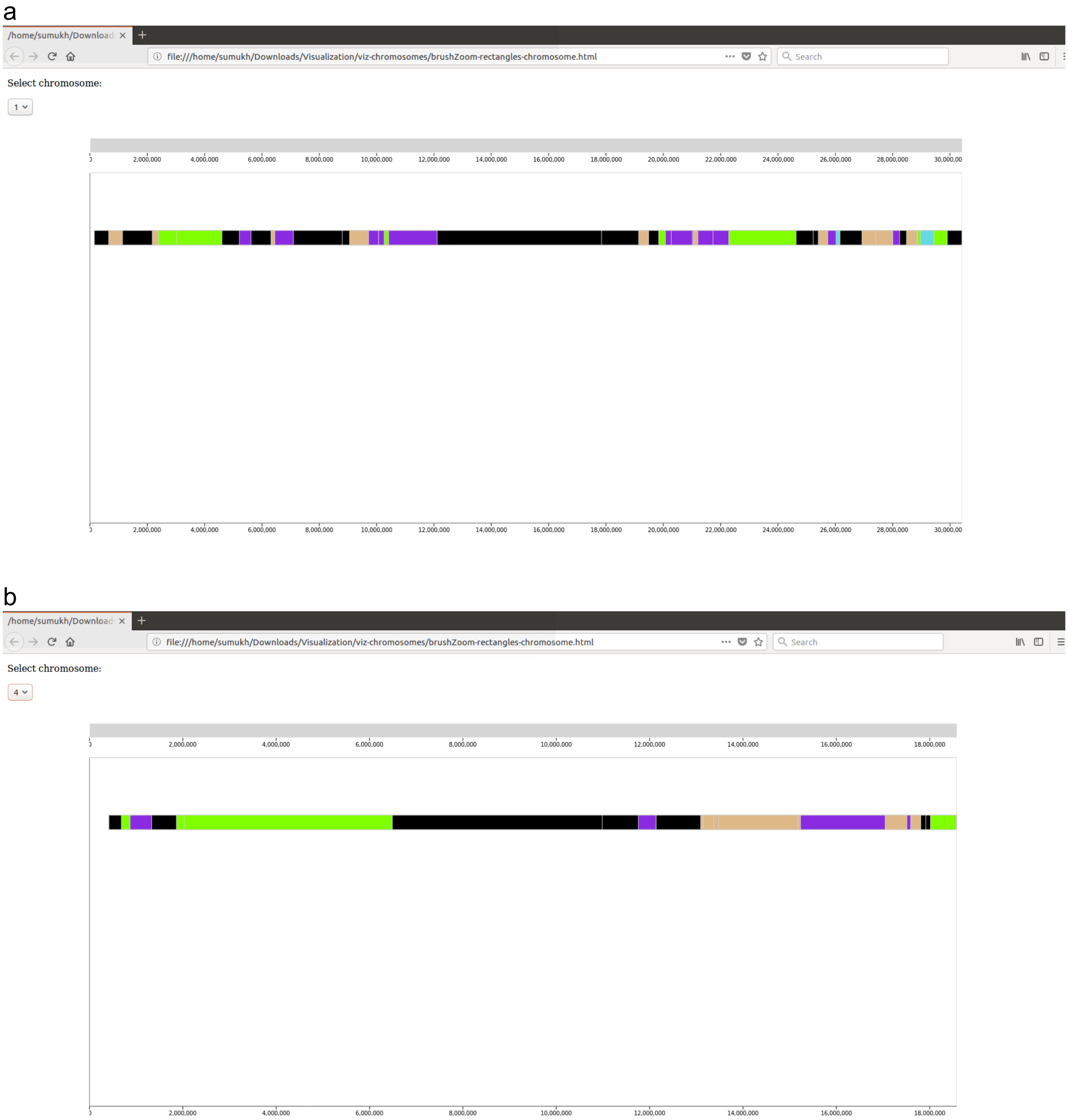
(5) Sense overlap exonic represented by dark cyan colour,

(6) Sense overlap intronic represented by dark blue colour,

(7) Bidirectional promoter represented by black colour.

It shows how several lncRNA classes are relatively aligned on the chromosome.

The current graphical implementation provides graphical visualisation on a single track thereby merging protein-coding RNAs and lncRNAs into single layer.



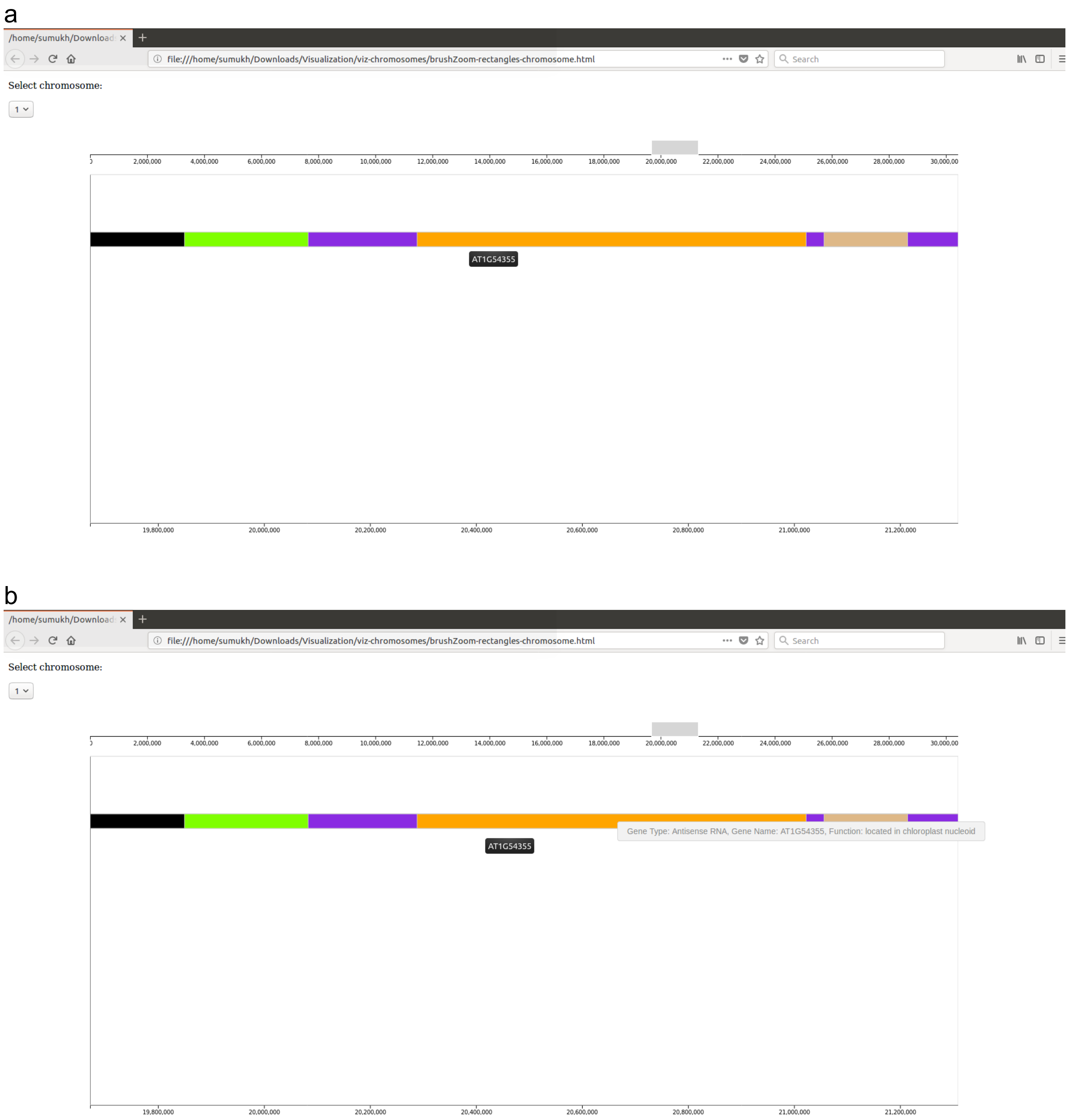
Select chromosome: 1

Select chromosome: 4

Figure 5.2: Visualisation of *A. thaliana* annotated lncRNA sequences showing chromosomal view of (a) chromosome 1, and (b) chromosome 4.

The D3-based graphical application also provides sequence view visualisation and annotation of lncRNA sequences. Figure 5.3 displays a zoomed-in view of chromosome-1 genomic sequence. The view shows the capability of the application to display the annotation information. Figure 5.3a shows the gene name/transcript ID whereas Figure 5.3b shows the complete annotation information of the gene. When the user hovers over the sequence, the gene name “AT1G54355” is displayed and the sequence is represented by orange colour. Upon right click on the sequence, complete annotation information is generated and exhibited which shows gene type as “Antisense RNA”, gene name as “AT1G54355” and function as “located in chloroplast nucleoid”.

The visualisation also provides a navigational coordinate view and coordinate view. The navigational view can be observed above the sequence view whereas the coordinate view can be observed below the sequence view. The navigational view is displayed by a small grey box between the genomic coordinates 18,000,000 bp and 22,000,000 bp. The grey box denotes the navigation box which can be navigated across the chromosome whereas the coordinate view exhibits the start and end coordinates of the grey box. The coordinate view displays the sequence starting from 19,700,000 bp to 21,300,000 bp.



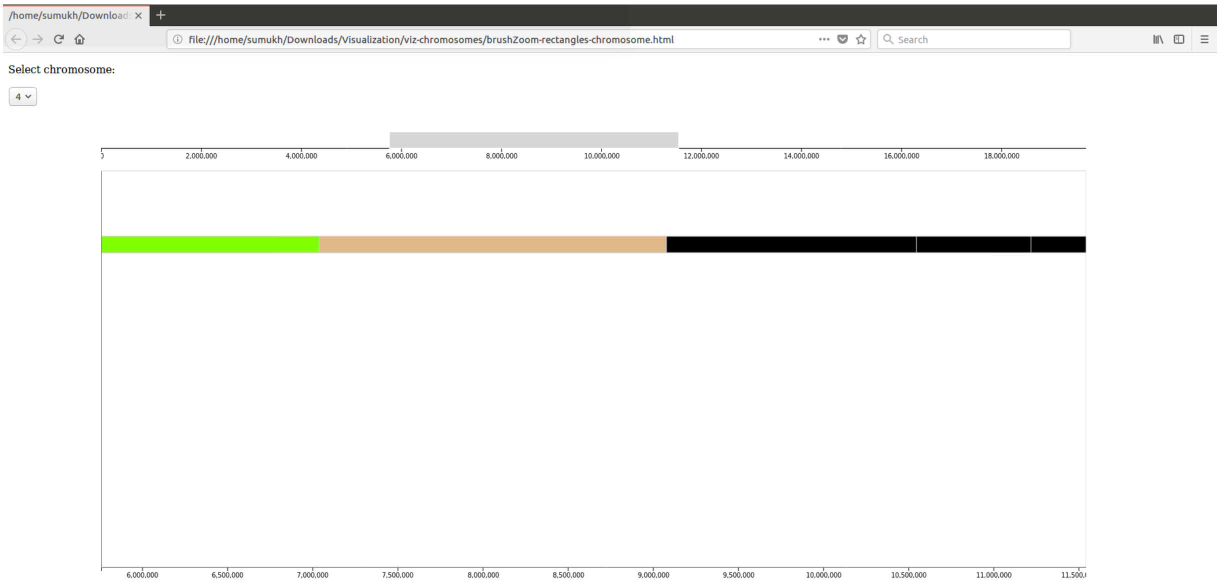
file:///home/sumukh/Downloads/Visualization/viz-chromosomes/brushZoom-rectangles-chromosome.html

Select chromosome: 1

Gene Type: Antisense RNA, Gene Name: AT1G54355, Function: located in chloroplast nucleoid

Figure 5.3: Visualisation of *A. thaliana* lncRNA sequences showing (a) semi-annotation of AT1G54355 sequence on chromosome 1, and (b) functional annotation of AT1G54355 sequence on chromosome 1.

As the user navigates the grey box, the sequence-view and coordinate view changes proportionately. The grey box can be resized to view a much larger portion of the genome which can be seen from the Figure 5.4. The screenshot illustrates that by resizing the grey navigational box, the sequence coordinates dynamically changes. The view also displays the antisense overlap intronic, exonic and bidirectional promoter sequences represented by their respective colours.



/home/sumukh/Download ×

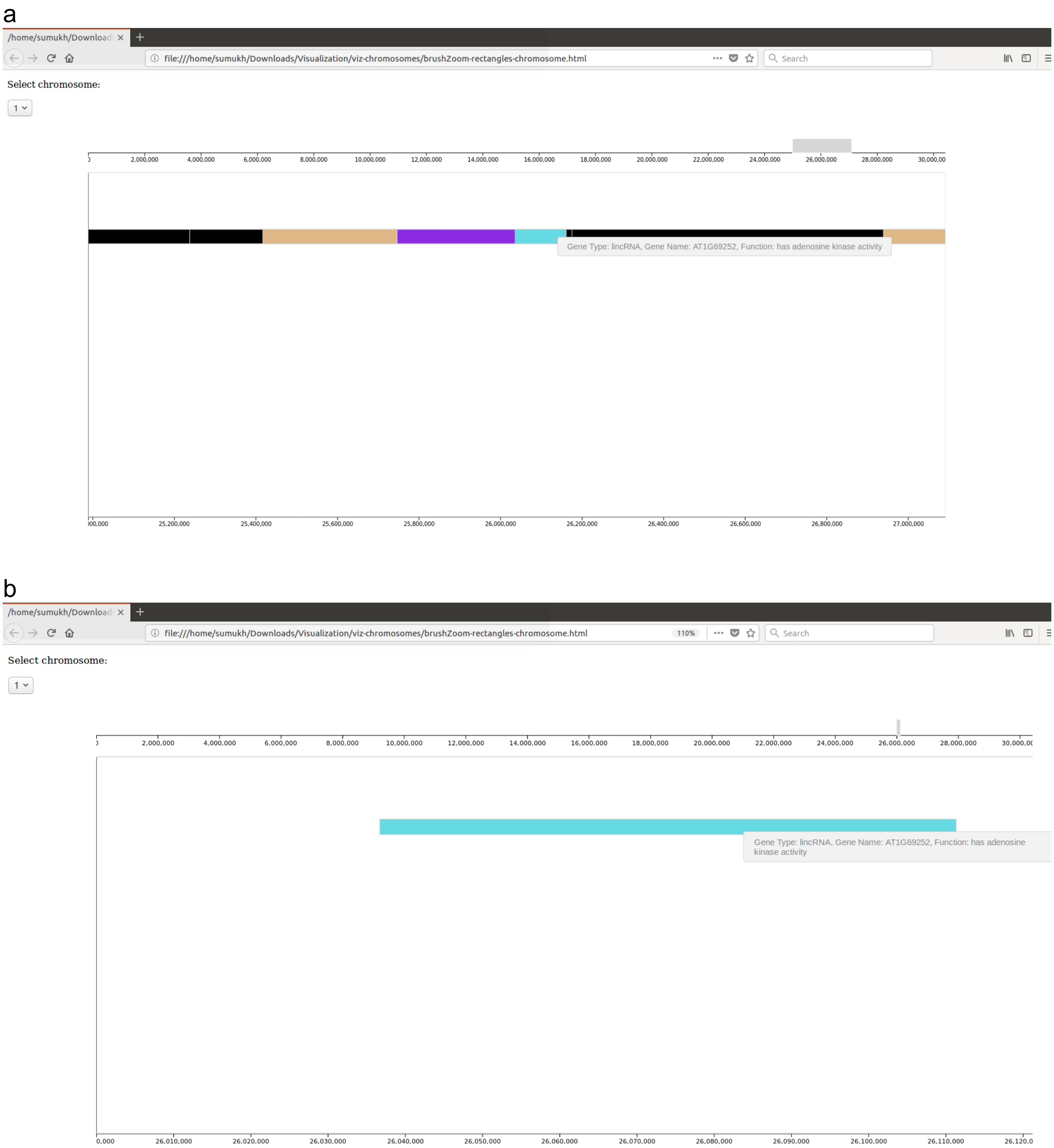
file:///home/sumukh/Downloads/Visualization/viz-chromosomes/brushZoom-rectangles-chromosome.html

Select chromosome:

4

Figure 5.4: Visualisation of *A. thaliana* lncRNA sequences showing zoomed-in view of chromosome-4.

Another feature of the visualisation application can be observed from Figure 5.5 which shows the relative alignment of sequences when the sequence is zoomed. Since the application constructs the visualisation based on sequence coordinates, a clustered view can be observed which places other lncRNA classes alongside Figure 5.4a. However, to observe an individual sequence, additional zoom is required which provides an independent sequence view Figure 5.5b. From the results, it can be clearly observed that intergenic lncRNA represented by brilliant arctic blue is aligned next to antisense overlap and bidirectional promoter sequences. Since the distance between the sequences are less, a discrete sequence view cannot be observed. When the sequence is zoomed-in, an independent view can be produced. Figure 5.5b shows the lincRNA sequence “AT1G69252” annotated with adenosine kinase activity function.



file:///home/sumukh/Downloads/Visualization/viz-chromosomes/brushZoom-rectangles-chromosome.html

Select chromosome:

Gene Type: lincRNA, Gene Name: AT1G69252, Function: has adenosine kinase activity

file:///home/sumukh/Downloads/Visualization/viz-chromosomes/brushZoom-rectangles-chromosome.html

Select chromosome:

Gene Type: lincRNA, Gene Name: AT1G69252, Function: has adenosine kinase activity

Figure 5.5: Visualisation of *A. thaliana* lncRNA sequences showing (a) lincRNA sequence on chromosome 1, and (b) Antisense RNA sequences on chromosome 2.

5.3 Summary

A major challenge in the analysis of RNA-seq data is the identification of lncRNA sequences among the plethora of RNA transcripts. With the advent of NGS technologies, RNA-seq experiments have led to an increase in the catalogue of lncRNA sequences. Despite this, visualisation and interpretation of the RNA-seq data still represents an unresolved challenge for the researchers looking for identification and functional characterisation of lncRNAs. Furthermore, a wide range of targeted softwares/tools have been developed for the visualisation of RNA-seq data, however, there has been significantly less emphasis on the tools responsible for visualisation of lncRNAs harboring functional annotation.

Currently, there are number of tools available for visualisation of specific aspects of the data which can be broadly classified into three categories: (1) Track-based, (2) network-based, and (3) data analysis based. Track-based tools such as UCSC genome browser (Kent *et al.*, 2002), IGV (Thorvaldsdóttir, Robinson and Mesirov, 2013), Ensembl (Fernández-Suárez and Schuster, 2010), GBrowse (Stein, 2013) and Artemis (Rutherford *et al.*, 2000) allow visualisation of mapped sequence reads, mutations and polymorphisms and continuous-value characteristic data such as DNA methylation, ChIP-Seq enrichment data, etc. Usage of track-based tools is practically limited by the screen space available. Since, most of the developed tools are web-based and Java-based desktop applications, the applications do not require higher computational resources. On the other hand, network-based tools such as Cytoscape (Shannon *et al.*, 2003) allow generation of two-dimensional or three-dimensional representations of the interactions, thereby providing flexibility to overlay gene expression data. Data analysis based tools offer integrated analysis and visualisation of RNA-seq data such as iSeq (Zhang *et al.*, 2018) and BrowserGenome.org (Schmid-Burgk and Hornung, 2015). Overall, while all the tools described above can be useful for visualisation and analysis of RNA-seq data typically rely on the availability of current lncRNA prediction tools for lncRNA identification and visualisation. Moreover, majority of the tools fail to provide comprehensive functional annotation of lncRNA genes. Therefore, tools that allow visualisation of annotated lncRNA sequences are still lacking.

To address the above-mentioned factors with genome-wide exploration and annotation of lncRNA sequences, the computational framework integrates a Javascript application which allows visualisation of lncRNA sequences with annotated information. The annotated information includes gene name, gene type, sub-class and molecular/regulatory function. The application is similar in functionality to currently available genome browsers such as UCSC Human Genome Browser. Current implementation of the application includes easier identification of various lncRNA sub-classes represented by various colours, easier navigation of and visualisation of individual chromosomes, chromosomal navigation functionality allowing visualisation of individual genomic segments. The application merges the protein-coding RNAs and lncRNA sequences into single layer or <div> element, due to which multiple samples cannot be visualised on a single track.

The application directly integrates the annotation information generated from the PBC classification and BMRF analysis through a Python application, which creates a D3.js compatible web-based format. The output CSV file can then be used by the Javascript application for conversion to visual elements. The application offers several advantages over other applications. First, the application does not require higher computational resources and is light-weight since it is developed on a D3 Javascript platform which can be executed on a single core CPU-based system with browser-based environment. Second, the application does not integrate reference genomes for visualisation. Usage of reference genomes drastically increases the computation times for generation of graphical elements. The application however, retrieves the chromosomal coordinates and creates a scale for individual chromosomes based on its length. Third, the application colours various lncRNAs according to the sub-class which helps in easier identification and interpretation. Fourth, since the application creates a visualisation based on Python and Javascript scripts, the application can easily be downloaded and exported to another system. Fifth, the application can easily display data from other organisms (i.e. plants, bacteria, mammals, etc.) with appropriate user-provided annotation.

CHAPTER 6: DISCUSSION

6.1 Introduction

This chapter presents a general discussion on the work and also discusses briefly the impact of the framework in lncRNA identification, classification and function prediction in RNA-seq datasets. This study recognizes accurate identification of lncRNA sequences as an important component for sub-classifying and predicting their molecular regulatory mechanisms. It also acknowledged that identifying the optimal set of features is essential for accurate identification. On this backdrop and given the fact that lncRNA sequences can be identified from a mixed set of protein-coding and non-coding sequences, several sequence and codon-bias features were used for construction of machine learning feature groups. Features were extracted from FASTA sequences. Using a feature optimisation method, an optimal feature set was extracted. The optimal feature sets were used for identifying lncRNA transcript sequences from protein-coding sequences.

The research employed a PBC algorithm for sub-classification of lncRNA sequences based on genomic coordinates. Results of the PBC algorithm were validated against the known lncRNA genomic annotations. Nevertheless, a function prediction algorithm was developed which employed a Bayesian network approach for predicting the regulatory mechanisms. Application of the tools and methods developed were applied on two plant RNA-seq datasets for comparison of developed methods with known coding potential computation tools.

6.2 Overview of the lncRNA identification approach

The present study introduced a computational framework for accurate identification, classification and function prediction of lncRNA sequences in plant RNA-seq datasets. An ensemble of 73 sequence and codon-bias features were constructed based on features identified in published experimental studies (Clarke, 1970; Fickett, 1982; Ikemura, 1982; Sharp, Tuohy and Mosurski, 1986; Karlin and Mrázek, 1996; Suzuki, Saito and Tomita, 2004; Wan *et al.*, 2004; Roymondal, Das and Sahoo, 2009; Amit *et al.*, 2012). The numerical features were constructed by extracting the data from individual FASTA sequence. The computational framework employed a feature optimisation method called LASSO-iterative Random Forests Feature Selection (LiRF-FS) for identifying an optimal feature set from training and validation sets. Using multiple FASTA sequences consisting of protein-coding and lncRNA genes, a feature matrix of 73 features and a binary class label was constructed using the feature extraction module of the framework.

LASSO employs a $\ell 1$-regularisation approach which leads to the generation of sparse features. At each value of λ, non-zero beta coefficients are generated which corresponds to the selection of features. LASSO shrinks the less important feature's coefficients to zero which leads to removal of zero coefficient features from the corresponding feature set. The beta-coefficient values are calculated on each λ value. The selected features at each λ are iteratively tested on the validation set matrix to compute the prediction accuracy of identification of lncRNA transcripts from protein-coding transcripts. The optimal feature set is obtained by selecting the feature set that produces the prediction accuracy between the tolerance accuracy value and the maximum prediction accuracy value. The LiRF-FS algorithm integrates the $\ell 1$-regularisation approach of LASSO with an iRF classification algorithm on every single λ value based on four parameters: λ_{lower}, λ_{upper}, $\lambda_{\mathrm{step-size}}$ and $tolerance$. λ_{lower} and λ_{upper} defines the lower and upper limits of λ values whereas $\lambda_{\mathrm{step-size}}$ defines the step size between each λ value. Tolerance defines the threshold value for selection of λ value. Based on maximum prediction accuracy, the algorithm constructs an array of λ values and searches for minimum and maximum number of features having prediction accuracy between the tolerance value and maximum prediction accuracy value.

For obtaining the optimal feature set in plant species, training and validation sets were constructed from 6-plant species: *A. thaliana*, *B. napus*, *B. rapa*, *B. oleracea*, *Z. mays* and *O. sativa*. Non-zero β-coefficient values generated on training set sequences were used for iteratively predicting the lncRNA and protein-coding sequences in validation set on each λ value. The algorithm generated a liver-shaped plot (Figure 3.9) with accuracy values of the primary vertical axis, feature set on the secondary vertical axis and λ values on the horizontal axis. Based on a tolerance value of 0.5, the algorithm selected two optimal feature sets consisting of 7 and 31 features. Whereas for the mammals, the algorithm produced 11 and 21 features with a peak accuracy values of 90.37%. The LiRF-FS method selected *hexamer score*, *mean ORF coverage*, *ORF coverage*, *Fickett score*, *Fop*, *RCB* and *SCUO* features in both the feature sets. The method additionally selected 5 codon-bias features and 10 RSCU codon-bias features along with 7 sequence-based features.

Optimal features selected using the LiRF-FS method were used for identification of lncRNA transcripts in the *A. thaliana* and *Z. mays* RNA-seq datasets. The lncRNA sequence prediction was performed using an iRF classifier with 73F, 7F and 31F feature sets and benchmarked against popular CPC tools. Benchmarking was performed using a 10-fold CV and repeated 10-fold CV. Results from the lncRNA prediction on both datasets demonstrated comparable performance of the 31F and 73F feature sets, whereas a slightly lower prediction performance of the 7F feature set on the *A. thaliana* EST (Figure 4.11) and *Z. mays* B73 (Figure 4.26) datasets. The prediction accuracy of the 31F feature set in both the datasets produced higher accuracy values on some folds. Improved prediction performance of the 31F feature set indicates preferential selection of codon-bias RSCU features in 31F feature set. The 31F feature set also selected all the sequence and ORF-based features along with RSCU features which suggests that *ORF length*, *ORF coverage*, *GC content*, *Hexamer score* and *Fickett score* are important discriminating features for identifying the lncRNA sequences from coding and other non-coding sequences.

6.3 Performance of the features for lncRNA identification

The potential of lncRNA identification using sequence and codon-bias features was measured on 8 plants and 2 mammalian datasets. The prediction of test set sequences was performed using an iRF classifier and compared against RF and SVM classifiers. The results demonstrate similar performance metrics of iRF and RF classifiers, whereas a slightly lower metric values were observed using the SVM. Comparison of accuracy, precision, sensitivity and F1-score exhibit lower performance of SVM in *B. Napus*, *B. rapa*, *B. oleracea*, *H. sapiens*, *M. musculus* and 6-plants datasets. However, comparison of computational speed in the model training step demonstrates better performance of SVM than the iRF and RF classifiers. The prediction using RF classifier relies on random generation of several decision trees. Since the iRF classifier iteratively produces n decision trees in each iteration, the time required for model training will proportionately increase with an increase in the number of iterations. Results from the iRF classifier prediction indicates similar prediction accuracies when compared to those obtained from the RF classifier.

Analysis of the AUC scores using iRF classifier in the plant and mammalian species showed higher true positive rate in plant species generating average AUC of 99.23. An average AUC of 96.82 was produced in *H. sapiens* and *M. musculus* species indicating greater accuracy in lncRNA identification as compared to mammalian species.

The iRF classifier implements an RIT function (Shah and Meinshausen, 2014; Basu *et al.*, 2018) for determining prevalent feature combinations/interactions in genomics datasets. The method was implemented for identifying feature combinations in 6-plants and 2-mammalian species. Results from the RIT analysis can potentially provide the selection of important features for the identification of lncRNA sequences. The analysis produced maximum of 5 feature combinations in both datasets in which higher order feature combinations generated greater prediction accuracy values (Figure 3.6) as compared to lower order feature combinations. The analysis suggested *ORF length*, *ORF coverage*, *Fickett score*, *RCB* and *SCUO* predominantly produced higher accuracy in plants, whereas *Hexamer score*, *ORF length*, *Fickett score* and *CGG*^{RSCU} produced higher accuracy in mammals. Removal of the features from the combination produced significantly lower accuracy values on order-3 and order-2 combinations. Since, RIT suggested an ensemble of feature combination, it failed to generate higher order combinations. This limits the analysis to detection of order-5 combinations due to which potentially significant features remains hidden. However, the analysis suggested a list of features required for lncRNA identification.

The performance of individual features was also measured using the iRF classifier to identify its prediction performance in plants and mammalian datasets (Figure 3.7). Results from the individual feature performance produced similar features obtained from RIT analysis. In addition, it generated a comprehensive map of individual prediction accuracies. The bar chart displayed higher accuracy values with several RSCU features in mammals. Selection of synonymous codon-bias features suggests that lncRNA prediction in mammalian datasets primarily depends on the frequency of synonymous codons observed in the transcript sequences. Whereas, in plants, identification of lncRNA sequences does not require selection of codon-bias features.

Based on LiRF-FS feature selection, frequency of selection of individual features across various λ values was observed for correlating the frequency of features having non-zero coefficient values with selection of optimal features (Figure 3.14). From the frequency analysis, three clusters were obtained having features with different frequency ranges. Results from the analysis generated the majority of sequence-based features with highest frequency values in plants and mammals dataset, whereas RSCU codon-bias features were selected with moderate and lower frequencies. In both species, *hexamer score*, *ORF coverage*, *mean ORF coverage*, *transcript length* and *Fickett score* were found to have been commonly selected. Apart from the sequence-based features, certain commonly selected codon-bias features, namely, *RCB*, *CUB*, *SCUO* and *GGG*^{RSCU} also displayed higher frequencies. Analysis of cluster-2 revealed 25 features with moderate frequencies which included *GC content*, *GTG*^{RSCU}, *CCG*^{RSCU}, *TAC*^{RSCU}, *CAC*^{RSCU}, *AAA*^{RSCU}, *GAC*^{RSCU} and *TGG*^{RSCU} commonly found in plants and mammals. Cluster-3 contained 30 RSCU features with lower frequency in multiple species. Twelve out of 30 features were found to have been commonly selected in both the species. Frequency of the features with non-zero coefficient values indicates higher selection of sequence-based features as compared to codon-bias features.

From the LiRF-FS feature selection analysis on 6-plants and 2-mammals data, several optimal sequence and codon-bias features were obtained with a maximum number of optimal features. The selection of optimal features in the LiRF-FS approach is based on the accuracy of lncRNA prediction in validation set transcript sequences. Other feature selection methods such as mRMR, Chi-square, Information Gain and UDFS, assigns relevance score or rank to each feature by considering each feature separately, due to which dependency between the features is ignored during model fitting. This leads to poor generalisation and over-fitting. Whereas feature selection from the iRF-RIT method provides limited knowledge of feature combinations.

Results from LiRF-FS feature selection and the application of selected features on prediction of lncRNA sequences extracted from RNA-seq datasets, demonstrated its wider application as well as potential to predict long non-coding RNA sequences in multiple plant species. The LiRF-FS method computes the beta coefficient values of the features on each λ value. Selection of the optimal features is also based on its speed of shrinkage. The faster the shrinkage of the coefficient values toward zero, the less likely are their chances of selection in the final feature set. This is attributed to the diamond-shaped constraint region of the LASSO regression (Figure 3.11). Whereas Ridge regression does not allow the shrinkage of the beta-coefficient values to zero due to its circle-shaped constrain region. The non-zero shrinkage behavior of Ridge regression prevents identification of optimal features. The trace path analysis of the coefficient values in 6-plants and 2-mammals species exhibits the points of beta-coefficient shrinkage (Figure 3.12). The majority of the features selected in the optimal feature set has been selected with $|\mathrm{coef}|/\max|\mathrm{coef}|$ values closer to zero. Features with greater $|\mathrm{coef}|/\max|\mathrm{coef}|$ values often do not get selected. This shows that the feature selection can also be derived based on frequency of shrinkage of coefficient values.

6.4 Performance of the framework against CPC tools

The prediction accuracies obtained from performance bechmarking on reference datasets demonstrates comparable performance with the other CPC tools. Since the CPC tools were primarily designed and their performance was validated on GENCODE, NONCODE and Refseq datasets, the accuracy values confirm their performance on these datasets. In contrast, the performance of the framework was at par with the state-of-the-art CPC tools displaying higher accuracy values on plant datasets.

The performance of the computational framework for prediction of lncRNA test set sequences was benchmarked against popular and powerful coding potential computation tools: CPAT, lncScore, PLEK and CPC2. To evaluate the robustness of the prediction accuracy of the framework, lncRNA sequences were extracted from several different sources. Transcript length distribution of TAIR10-annotated and EST-derived lncRNA transcripts demonstrate the degree of sequence length variation in lncRNA transcripts (Figure 4.8). Sequences derived from the TAIR10 annotation data ranges between 200 bp and 8000 bp whereas sequences derived from EST analysis ranges widely between 200 bp and 7.8×10^5 bp. Additionally, ORF count of EST-lncRNA sequences reveal counts greater than 700 ORFs per frame. Such extremely long lncRNA sequences are generally misclassified as protein-coding transcripts, due to which the overall prediction accuracy decreases.

The efficiency and robustness of the framework was tested by predicting the test set sequences in shuffled and non-shuffled datasets. Results from the 10-fold CV benchmarking on the *A. thaliana* and *Z. mays* datasets indicate that the 31F set obtained from the LiRF-FS approach outperformed other tools with greater precision in identifying the lncRNA transcripts. The 31F set demonstrated an average difference of 14.6% with PLEK, 22.7% with CPAT, 9.2% with lncScore and 28.47% with CPC2 on *A. thaliana* TAIR10 D2 dataset. Both state-of-the-art tools, CPAT and CPC2 exhibited lowest prediction accuracies of 55.51% and 49.75% respectively, thereby exhibiting poor prediction performance on the TAIR10 and EST datasets respectively. EST datasets exhibited an average difference of 14.48% with PLEK, 22.61% with CPAT, 13.45% with lncScore, and 26.75% with CPC2. The overall prediction accuracy difference ranged between 9% and 30% for *A. thaliana* whereas a significantly higher difference range of 3% – 50% was observed for *Z. mays*. Comparison of prediction accuracy between the 73F, 31F and 7F sets reveals better performance of 31F on some folds when compared with 73F in *A. thaliana* transcript sequences. In *Z. mays*, 31F and 73F displayed similar performance with negligible differences, thus indicating better selection of maximal number of optimal features as compared to minimal number of optimal features by the LiRF-FS method.

The robustness of the framework was further evaluated and benchmarked against CPC tools with repeated k-fold CV analysis. Results from the analysis clearly showed superior performance of the framework with stable accuracy values in TAIR10 and EST datasets. Results clearly demonstrate lower prediction accuracies in lncRNA prediction by lncScore, CPC2, CPAT and PLEK tools where a decrease was observed from TAIR10 to EST-annotated lncRNA sequences. The accuracy values from the framework showed an average marginal deviation of 1.81% between the datasets which certainly indicates its higher efficiency and robustness among currently popular CPC tools. Additionally, higher prediction accuracies of 96.27% with 31F feature set in *Z. mays* dataset signifies its precision in identifying lncRNAs in plant species.

6.5 lncRNA sub-classification of lncRNA transcripts

For sub-classification of lncRNA transcript sequences into seven different types, a position-based mapping algorithm has been developed. The algorithm classifies the lncRNA sequences based on overlapping and non-overlapping of genomic coordinates. The genomic coordinates of lncRNA exonic (E) and intronic (I) sequences are compared against the coordinates of the protein-coding exonic and intronic sequences. The classification is also based on additional parameters such as chromosome name and DNA strand. The algorithm extracts the ORFs from each transcript sequence. The E and I sequences used for coordinate overlapping are extracted from the ORF sequences. The mapping of lncRNA E and I coordinates against protein-coding E and I sequences provides greater precision and accuracy when applied on genome-wide scale. This allows identification of several lncRNA classes.

The PBC algorithm was developed based on two rules: (1) Rule 1 classifies the lncRNA sequences based on position of transcript sequences on sense (lncRNA and mRNA on ‘+’ DNA strand) and antisense (lncRNA and mRNA on ‘-’ DNA strand) strands, and (2) Rule 2 classifies the lncRNA sequences based on position of transcript sequences on sense (lncRNA and mRNA on ‘+’ DNA strand or lncRNA and mRNA on ‘-’ DNA strand) and antisense (lncRNA on ‘+’ and mRNA on ‘-’, or vice versa) strands. Based on these rules, classification was performed into seven different types, namely, Sense Overlapping Intronic (SOI), Sense Overlapping Exonic (SOE), Antisense Overlapping Intronic (AOI), Antisense Overlapping Exonic (AOE), Antisense (ANT), Bidirectional Promoter (BDP) and Intergenic (INT). The classification was performed on HS and MM GENCODE sequences to identify the number of identical matches and additional classification of “Processed Transcript” and “To be Experimentally Confirmed” sequences.

The PBC classification of *H. sapiens* lncRNA sequences generated a 61.18% match with GENCODE annotation whereas classification of *M. musculus* generated a match of 54% on different chromosomes. The dissimilarity of matching sequences between PBC and GENCODE results is attributed to the classification rules defined by the method. Density distribution analysis (Figure 3.17) of antisense lncRNA sequences from GENCODE does not classify the sequences based on their genomic annotation. PBC classification of lncRNA sequences is exclusively based on the genomic annotation and overlapping of sequences whereas the GENCODE annotation does not follow the genomic coordinate rules, due to which the match decreases. Classification based on Rule-2 generated a higher proportion of INT and ANT sequences in *H. sapiens* and *M. musculus* sequences. However, the proportion of sequences classified as INT was comparatively higher than the ANT class. Furthermore, a higher percentage of sequences classified as BDP were obtained in both the datasets. This demonstrates a large amount of sequences which were found to occur within 1000 bp of the transcriptional start sites of the mRNA sequences.

To further evaluate the classification accuracy of the PBC approach, the classification was also performed on the TAIR10-annotated *A. thaliana* and Ensembl-annotated *Z. mays* RNA-seq derived transcript sequences. Classification results from the PBC annotation were compared against the experimentally annotated lncRNA transcripts from the TAIR10 and Ensembl Genomes 39 AGPv4 databases. Classification performance across the chromosomes measured an average accuracy of 72.55% for *A. thaliana* Natural Antisense Transcripts (NATs) and 90.86% for *Z. mays* with long intergenic lncRNAs (lincRNAs) sequences. Sub-classification analysis generated a higher matching percentage of ANT sequences in *A. thaliana* and INT sequences in *Z. mays* sequences. Analysis of annotation information shows a higher proportion of ANT sequences in *A. thaliana* and INT sequences in *Z. mays* which indicates greater accuracy and precision of PBC approach in plant species. Comparison of matching percentages of Rule-1 and Rule-2 generated greater precision with the Rule-2 based PBC approach. This clearly suggests partial-dependence of lncRNA strand-specific overlapping and hence, is dependent on the Rule-2 PBC approach.

The sub-classification of lncRNA transcripts from the RNA-seq datasets has been previously conducted by Wucher et al. (Wucher *et al.*, 2017). The authors developed an alignment-free classifier tool called FEELnc for identifying and annotating lncRNAs based on an RF classifier. The tool filters out coding and other ncRNAs to retain probable lncRNA sequences based on GTF file derived from Cufflinks analysis (Trapnell *et al.*, 2012), thereby constructing known mRNA and lncRNA GTF files. It identifies potential candidate lncRNAs based on intrinsic sequence features based on reference genome. The tool has two major drawbacks: (1) as the coding potential computation is performed primarily based on the reference sequence, accurate identification of lncRNA sequences in transcriptomic datasets cannot be achieved. (2) Since the tool predominantly classifies the sequences based on GTF information, precise determination of various sub-classes cannot be achieved. PBC-based approach attempts to solve the problem by classifying the transcripts based on sequences from BAM file. Furthermore, it evaluates the degree of overlap by aligning the lncRNA and mRNA E and I sequences using Smith-Waterman pairwise-sequence alignment algorithm (Pearson, 1991) and provides an alignment score.

Pan *et al.* (2015) classified circular RNAs using Multiple Kernel Learning (MKL) approach. In addition, the tool performed multi-class classification based on known set of lncRNAs (antisense, lincRNA, circularRNA and processed transcripts). Multi-class classification generated lower prediction accuracy of 60.4% whereas an accuracy of 77.8% was achieved for identification of circular RNAs on the human test set sequences. Since, a limited number of lncRNA classes are involved in model training, a comprehensive identification of other lncRNA classes cannot be performed. Moreover, due to the unavailability of source codes and failure to execute on linux-based system, validation tests could not be conducted which additionally limits their usage.

Results from the PBC classification generated a higher matching percentage of lncRNA transcripts of the plant species as compared to those obtained in humans and mice. Classification analysis generated a higher match with ANT and INT sequences in *A. thaliana* and *Z. mays*, respectively. The application of the PBC algorithm clearly demonstrates its applicability on the plant species.

6.6 Identification of novel flowering genes in *A. thaliana* apical-shoot dataset

Recent progress in determination of DGE in RNA-seq data using several bioinformatics tools enabled easier identification of genes from samples. A number of tools for processing and analyzing RNA-seq data have been developed. These include Cufflinks, edgeR, DESeq, RSEM and others which claim accurate identification of DEGs. However, the accuracy can only be determined by comparison of results obtained from several computational tools with those obtained from published experimental studies. Using recently published tools for RNA-seq data, a comparative analysis of results obtained from Cufflinks-Cuffdiff2, DESeq and edgeR was performed and analysis of intersection of DEGs from two or more tools was recommended in order to obtain more robust results (Zhang *et al.*, 2014). The framework integrates a computational pipeline for identification of novel DEGs from plant RNA-seq datasets. In this case study, a computational approach was developed for the identification of DEGs in *A. thaliana* RNA-seq time-series datasets which includes quality checking, adapter trimming, reference alignment, DEG analysis, alternative splicing classification, DEG merging, GO enrichment and pathway analysis (Figure 2.1).

The first step in identification of DEGs is to perform accurate genome alignment. Inaccurate parameters often result in the generation of incorrect read counts from the data which could potentially result in erroneous downstream processing. Previous investigations used default values for processing RNA-seq data (A. V. Klepikova *et al.*, 2015) which included similar minimum intron length values of 70 nt for plants and mammals (Goodall and Filipowicz, 1990). However, mean, medium and minimum intron length in *A. thaliana* and *O. sativa* were found to be much lower (Deutsch and Long, 1999; Wang and Brendel, 2006) than the previously identified and established value of 70 nt. Therefore, to correctly identify DEGs from the data, custom parameter values were applied to generate precise alignment of samples against the reference genome.

The key step of RNA-seq data analysis is to identify DEGs using appropriate statistical models. Once the FPKM counts from the sequencing reads were obtained, these were used for finding DEGs using Cuffdiff, DESeq and edgeR. Usage of Cuffdiff, DESeq and edgeR methods increase statistical power and help in rationale comparison and thus confirming the suitability of the results. Results show that both Cuffdiff and edgeR displayed significant numbers of DEGs in the floral transition sample pairs S7-S10, S7-S12 and S7-S13 (Table 4.1). Expression profiles of the DEGs were compared against known flowering genes *FLC* and *LFY*. Consistent with the published experimental results, results obtained from the current study produced higher mean PCC of 0.86 and 0.88 for *FLC* and *LFY*, respectively which is consistent with published results (Michaels, 1999; A. V. Klepikova *et al.*, 2015). Apart from the known flowering genes, several other experimentally-validated genes responsible for flower development in *A. thaliana* were found to be DE in the apical-shoot dataset.

By the overlapping of DEGs obtained from Cuffdiff, DESeq and edgeR, 690 genes were found to be commonly expressed. To identify the novel DE flowering genes, functional enrichment was conducted for identification of genes associated with highly enriched biological processes. Functional enrichment analysis resulted in determination of several gene associated with GO terms. The resulting gene-GOterm associations were filtered having q-value ≤ 0.05. The genes were found to be associated with several biological and molecular functions which include association with glucosinolate biosynthesis, mitosis, meiosis, cell cycle development, flower development, mismatch repair, etc. Additionally, a comparison of expression profiles against cell-cycle related genes was also carried out to obtain the degree of variation between those obtained from Klepikova et al. (2015) which included *CDKA*, *CDKB*, *CDKC*, *CDKD*, *CDKP*, *CDPT* and cyclin genes. Results from the comparison showed that most of the CDK genes exhibited moderate correlation with an average ranging between 0.60 – 0.70, while some CDKs displayed particularly higher correlation above 0.90. 15 genes were found to have poor correlation ranging between 0.20 – 0.60. The expression profiles of poorly correlated genes showed lower expression during transition phase from Klepikova et al. (2015). Experimental results clearly shows that certain genes such as CSK1 is constitutively expressed during mitotic and endoreduplication cycles (Jacqmard *et al.*, 1999).

Results from PPI network analysis showed most of the DEGs during the transition phase regulate other DEGs which provide induced resistance and protection against external factors such as stress, pathogens, herbivores, temperature variations, etc. A recent study on the relationship of glucosinolates to flowering in *A. thaliana* suggests that the presence of the *MAM1* gene affects glucosinolate accumulation and flowering time in the absence of *APOP2* and *APOP3* genes and leads to production of C3 glucosinolates (Jensen *et al.*, 2015). Results from the PPI network analysis clearly show that *MAM1* regulates several other genes in glucosinolates and displays a high expression profile correlation of 0.75 to *FLC* which supports the hypothesis of glucosinolate production and protection during flowering phase.

To determine the similarities in the expression profiles of 690 genes and their degree of regulation by *FLC* and *LFY* genes, a correlation analysis was conducted. Apart from identification of genes involved in regulation of glucosinolate compounds, several novel flowering genes were identified by clustering of 690 commonly expressed DEGs. PPI network analysis revealed 76 novel genes showing stronger regulation and displaying the highest correlation in expression with *FLC* and *LFY* genes. Out of 76 genes, 55 and 3 genes showed no regulation by *FLC* and *LFY*, respectively. From the research study performed in RNA-seq derived *A. thaliana* dataset, an approach has been proposed for determination of novel DEGs which were found to be involved in the regulation of flower development.

6.7 lncRNA function determination in plant datasets

Functional determination of lncRNA genes has been studied previously in mammalian species and several computational models have been proposed which includes determination of lncRNA functional similarities and lncRNA-disease associations using LRLSLDA (Chen and Yan, 2013), LncDisease (Wang *et al.*, 2016a), IRWRLDA (Chen *et al.*, 2016b), LFSCM (Chen, 2015) and FMLNCSIM (Chen *et al.*, 2016a). However, the studies conducted were mostly focused on the prediction of functional similarity and disease associations based on lncRNA-disease association data and lncRNA-protein interaction data in mammalian genomes. Therefore, current prediction models limit their usage for predicting regulatory functions in plant species.

The computational framework integrates NRLMF-based derivation of lncRNA-protein interaction in plant datasets for predicting the functions of lncRNAs. Novel interactions of lncRNA and proteins were determined with scores ranging between 0.7 and 0.9 demonstrating strong probability of interaction. Interacting lncRNAs and proteins were utilized for function prediction using Bayesian-based regulatory network-based approach. Interactions were determined based on logistic matrix factorisation approach by employing sequence similarities between target lncRNA and protein sequences in plants and known lncRNAs and proteins in humans obtained from the NPInter database (Wu *et al.*, 2006). NRLMF predicts the interactions by identifying neighbouring genes having higher sequence similarity with NPInter sequences. Based on the LPI pairs, PPI pairs and protein-associated GO terms, BMRF computes probability of association of GO term with the lncRNA gene. The proposed method inspired by Liu *et al.* (2017) extends the work for determination of potential LPI pairs and integrates with the Bayesian approach (Kourmpetis *et al.*, 2010) for associating functions to lncRNAs.

Based on Gene Co-expression Networks (GCNs) approach, candidate gene shares similar functionality when co-expressed with another gene (van Dam *et al.*, 2017). This method has successfully been applied for associating potential regulatory roles in various diseases (Liu, Li and Li, 2014). The NRLMF-BMRF module of the framework integrates the transcriptional coexpression data of lncRNAs and proteins by reinforcing the results from NRLMF approach and provides an additional layer for generating reliable outcome.

Various research studies using co-expression analysis focused on plant datasets include determination of DE “stress-tolerant” lncRNA genes in modern and wild wheats based on co-expression with miRNA genes and identification of regulatory mechanisms of lncRNAs in Maize based on co-expression of lncRNA, mRNA and miRNA genes (Xu *et al.*, 2017). However, currently known methods based on co-expression of genes do not focus on derivation of potential interactions between the two pair of genes. Failure to determine the interactions weakens the hypothesis due to which accurate predictions cannot be made.

Results of LPI pairs derived from co-expression-based study reveals a mismatch with the results from NRLMF analysis. Co-expression-based analysis was primarily conducted based on the assumption of higher correlation in functional similarity of lncRNAs with protein-coding genes. Although, the LPI pairs derived from higher correlation in co-expression showed considerable match with the experimentally-published results, results from the NRLMF-based analysis provides a more reliable approach for identifying the true-positive LPI pairs.

Thus, identification of lncRNA interacted proteins is essential for understanding complex functions of lncRNAs (Derrien *et al.*, 2012; Washietl, Kellis and Garber, 2014). Determination of reliable LPI pairs is dependent on greater sequence similarity scores of a target gene sequence with the known gene sequence. The higher the sequence similarity, the greater the chance of obtaining true positive pairs. Analysis of few LPI pairs from gene co-expression correlation analysis (Figure 4.20 and 4.32) indicates that for a given LPI pair, considerable co-expression can be observed along with lower relative expression values which reduces the overall PCC ranging between -0.2 to 0.6.

Using co-expression based NRLMF-BMRF approach, the lncRNA genes were found to have been associated with several regulatory mechanisms such as regulation of DNA replication, gene expression, cell division, DNA-templated transcription and vernalisation response. Results from the analysis showed that the lncRNAs in *A. thaliana* are primarily associated with nuclear functions, regulation in heterochromatin assembly, cell differentiation, regulation of proteasomal complex and sequence-specific DNA transcription factor activity (Figure 4.22). An analysis of *Z. mays* lncRNA data revealed regulatory function association in “ATP binding”. However, based on the LPI data, the lncRNA can also be associated with DNA transcriptional regulatory functions, translational regulation, regulation of protein metabolic process, and response to stress. A review of lncRNA functional mechanisms using computational techniques illustrates potential roles in DNA methylation/chromatin remodeling, RNA translation/splicing, miRNA binding, protein scaffolding, protein modification through phosphorylation, protein stability by promoting the degradation of vimentin (Wang *et al.*, 2015; Signal, Gloss and Dinger, 2016). The roles identified and listed by Signal *et al.* (2016) correlates with the functions identified in ATH as illustrated in Figure 4.22 and Appendix Table A.1. Although the correlation in relative expression values obtained from co-expression analysis does not produce a reasonable match with those obtained from NRLMF approach, the functions predicted through both the methods indicates clear correlation of transcriptional and post-translational regulatory functions predicted using BMRF.

Furthermore, an examination of the node degrees from the regulatory network analysis of the LPI-PPI datasets revealed power-law distributions with a slope ranging between -0.8 and -0.93 and R^2 ranging between 0.5 and 0.7. These results indicate that the regulatory network constructed is similar to many biological networks and is well characterised by co-expression regulation principles (Nacher and Akutsu, 2007). These parameters distinguish the generated regulatory network from the randomly generated networks. To identify the important vertices or hub nodes, BC and CC measures were performed which measures the centrality in a graph. Network analysis of the LPI-PPI ATH and ZM genesets revealed an average BC of 0.475 for the nodes. CC measures the centrality by computing the lengths of the shortest paths between the nodes in the network. The results clearly demonstrate smaller CC measures ranging between 0.25-0.5 in both the species, which indicates that the network consisted of nodes connected over shorter distances.

The experimentally-published lncRNA functions were used for confirming function associations in *A. thaliana* and *Z. mays* datasets. Results from the LPI-PPI BMRF analysis produced association of several regulatory functions to lncRNA genes. Experimetally-published lncRNA functions (Liu *et al.*, 2015; Signal, Gloss and Dinger, 2016) show that the lncRNAs predominantly functions as transcription and translational regulators, predicted to have potential roles in DNA methylation, heterochromatin assembly, cellular differentiation and in protein modification with the majority of functions associated in the nucleus of a cell. Results obtained from NRLMF-BMRF analysis thus validate their roles as identified from various computational and experimental studies published and reported.

6.8 Summary

This chapter provided an overall discussion on the lncRNA identification implemented in the framework and the results obtained by its application on plant and mammalian reference and RNA-seq datasets. The performance of sequence and codon-bias features on multiple species was also discussed. This was followed by a discussion of the results on the performance evaluation of the framework against several coding-potential computation tools with emphasis on the prediction accuracy, sensitivity, specificity, F1-score, NPV and MCC metrics. Results from the sub-classification of the lncRNA sequences obtained from the GENCODE, Refseq and Ensembl databases were discussed providing details on the *de-novo* classification performed on the lncRNA sequences from the datasets and intersection of the results from the PBC and database-annotated sequences.

This chapter also provided results and detailed analysis of the identification of novel DE flowering genes from *A. thaliana* data. Furthermore, a discussion on function prediction of lncRNAs in *A. thaliana* and *Z. mays* was performed providing details of the results from the analysis and synopsis of the centrality measures for evaluating the regulatory network. It included interpretation of the results from the identification of novel lncRNA-protein interactions derived from NRLMF analysis in *A. thaliana* and *Z. mays*. Additionally, it discussed the results from co-expression based BMRF anlaysis and demonstration of filtering algorithm on co-expression derived data.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

In the preceding chapters it has been shown that the use of our proposed computational framework for identification, sub-classification and function annotation of lncRNA genes in plant species is beneficial.

The motivation for the development of this framework was the identification of a gap in the area of lncRNA prediction and function annotation in plant genomes. The framework addresses this through accurate identification of lncRNA sequences in plant species. The work also presents a novel approach for the sub-classification of lncRNA sequences based on its genomic coordinates. Additionally, the work encompasses the computational prediction approach for identification of molecular/regulatory functions of lncRNAs in plants. Accurate prediction of lncRNAs remains one of the major open problems in plant genomes. Therefore, accurate and efficient computational methods are required to predict lncRNAs in plants to further investigate their roles.

LncRNAs have been found to perform various functions in several biological processes. In order to interpret the lncRNA functionality, it is therefore convenient to classify the lncRNAs. Current methods developed for classification of lncRNAs rely on the construction of learning models. The data obtained for generating the training set is, in particular, derived from an annotated set of lncRNA sequences obtained from human and mouse genomes. Due to the limited availability of annotated lncRNA sequences for constructing the training set, current methods cannot be applied for classification of lncRNA sequences in the plant genomes. The proposed method provides instead a learning-free approach and classifies the lncRNA transcripts based on its FASTA sequence derived from transcriptomic datasets and their relative coordinates with protein-coding genes.

Furthermore, current methods for prediction of molecular functions typically focus on the mammalian genomes and potential roles in diseases. Less attention has been given to the development of computational methods for function annotation of lncRNAs in plants. The proposed framework incorporates a computational pipeline for predicting the functions based on lncRNA-protein interactions and co-expression of genes derived from transcriptomic data.

In the present chapter, the results of the complete study have been summarised and the directions for the future course of actions/work have been outlined.

A detailed analysis of the computational framework included comparison of plants and mammalian transcript sequences based on GC content, transcript lengths and number of individual base pairs. The analysis included comparison of different classifiers for the classification of mRNA and lncRNA sequences obtained from the GENCODE and Refseq databases. Results showed comparable performance of the RF and iRF classifiers and ability of the features to classify the sequences obtained from multiple species. Results from the LiRF-FS method generated two separate optimal feature sets. The selection of optimal features primarily depends on its prediction accuracy lying above the threshold value. Selection of the features was performed on a mixed set of transcript sequences obtained from 6 plants and 2 mammalian species to account for sequences with varying characteristics. A comparison of individual features was also performed to identify their performance using an iRF classifier. Intersection of the results from the feature selection and individual feature performance indicated reliable selection by the LiRF-FS method. Results from the LiRF-FS were compared against other feature selection methods and showed comparable performance offering better selection of features generating higher prediction accuracy. Finally, an in-depth analysis of the PBC classification algorithm was conducted on humans and mouse annotated FASTA sequences for comparison of PBC-based classification results. Results demonstrated a match of ~60% with Rule-2 classification. The mismatch in the classification is primarily attributed to the rules defined for classification of the sequences. Contrastingly, PBC successfully classified the "Processed Transcripts" and "To be Experimentally Confirmed" lncRNA transcripts into seven different classes.

Furthermore, a detailed analysis of the framework was conducted on *A. thaliana* and *Z. mays* transcript sequences for performance evaluation and prediction of potential regulatory functions of lncRNAs. Benchmarking evaluation of the framework on *A. thaliana* and *Z. mays* datasets against popular CPC tools demonstrated superior performance based on 10-fold CV and repeated 10-fold CV exhibiting accuracy difference of 8-30% increase in *A. thaliana* data, whereas 3-50% increase in the accuracy was observed in *Z. mays* data. LncRNA sequences obtained from TAIR10 and EST-derived sources typically ranges widely between 200 bp and 7.8×10^5 bp. Identification of such extremely long sequences becomes trivial however they are often misclassified with protein-coding genes.

Results from the benchmarking analysis showed improved performance with feature selection. Results demonstrate that prediction accuracy of the maximum optimal features is higher than that obtained from the minimum optimal features with an observable difference of 2-3%

between both the feature sets. The selection of codon-bias features improves the classification performance and therefore presents species-specific preferential selection of codons. Based on the sequence similarities of lncRNA and protein-coding genes, novel lncRNA and protein interactions were derived using the NRLMF approach. Top scoring interactors were used for predicting the functions using the BMRF approach which calculates the probability of function association from the network of connected genes. The experimentally-determined molecular functions from plant species provide a list of most probable regulatory functions associated with lncRNA. Based on the known RNA-Protein interactions from the NPInter database, novel interactions in plants were derived which demonstrated their potential role in the regulation of several biological processes such as DNA transcription, methylation, cell cyle processes, DNA damage repair, chromatin modification, DNA replication and gene expression.

Application of a keyword-filtering algorithm on the co-expression of lncRNA and protein-coding genes showed association of several lncRNA genes in *A. thaliana* and *Z. mays* serving as transcriptional regulators, splicing regulators, involved in histone acetylation, phosphorylation, methylation and ubiquitination, translational elongation, and in post-translational gene silencing. A single lncRNA gene can be associated with multiple functions, however, it is essential to determine the most probable function from these. By utilizing the function list as a bag-of-words, the keywords can be matched against the list of predicted molecular functions of lncRNA genes. This generated a close match with the experimentally-verified function list having higher association probability.

The validity of the regulatory network has been confirmed by assessing the Betweenness Centrality and Closeness Centrality measures. Studies revealed a power-law distribution with an average R^2 of 0.75 indicating a relatively good fit. Small variations in the centrality values would not affect the statistical accuracy of the regulatory network model significantly. Overall, the NRLMF-BMRF approach provided reasonable results, which were in line with the betweenness and closeness centrality measures as presented in Chapter 6.

Overall, deploying the framework offers several benefits over the currently used methods. First, apart from commonly known distinguishing sequence-based features such as ORF length, GC content and Fickett score, it takes advantage of codon-biased features to increase discriminative power. Second, it implements a powerful semi-supervised optimisation approach for selection of principal features which can be applied to any species. Third, an integrative approach of LiRF-FS and codon-biased features provides insights into preferential selection of species-specific synonymous codons in the classification process. Fourth, implementation of coordinate-based mapping algorithm for sub-classification provides valuable insights into different features of lncRNAs and their underlying functional mechanisms in non-model species. Fifth, the model determines novel interactions between lncRNA and proteins based on sequence similarities and logistic matrix factorisation approach. Sixth, it provides functional annotation for the predicted lncRNAs using integrated NRLMF and BMRF analysis which takes advantage of lncRNA-protein interactions and co-expression data obtained from RNA-seq data. Application and filtering the LPI pairs based on co-expression analysis strengthens the function prediction approach and assists in determining true positive LPI pairs, thereby providing accurate function prediction.

One of the drawbacks of the framework is that in order to generate species-specific optimal features, a comprehensive range of λ values are required which is dependent on the adjustment of the λ_{upper}, λ_{lower} and $\lambda_{\mathrm{step-size}}$ parameters. An adjustment of these parameters is required since the β coefficient values are dependent on the values of λ. Another drawback is the adjustment of the tolerance value for the computation of threshold prediction accuracy values. Smaller tolerance values restrict the scanning of optimal λ values producing reasonably similar accuracy with minimal difference from the maximum prediction accuracy, whereas larger tolerance values create a large array of optimal λ values having a range of prediction accuracies with larger difference. With an increase in the parameter values mentioned above, the time required for the scanning the non-zero β coefficient values increases. However, a wider range of λ values provides a larger search space which increases the efficiency of feature selection. Since the lncRNA function prediction was performed on a random subset of lncRNA sequences in *A. thaliana* and *Z. mays* datasets, moderate sequence similarities were observed due to the limitation of data analysis on a smaller subset.

The lncRNA identification and classification is predominantly based on the extraction of the consensus FASTA sequences, based on the variants identified by the bcftools which are computed based on the alignment of the reads against the reference genome. The consensus FASTA sequence obtained therefore provides a probable FASTA sequence of the lncRNA or mRNA transcript associated with certain probability. This does not guaranty it will represent the exact genomic sequence of the species-specific sample. The sub-classification of lncRNAs is primarily based on the relative position of the lncRNA with the protein-coding sequences. For accurately classifying the lncRNA genes, a comprehensive genome-wide array of protein-coding genes is required. An incomplete list of protein-coding genes would lead to false positive results and inaccurate classification. Current implementation of the PBC algorithm classifies the lncRNA sequences using a single processor which significantly increases the computation time. Similarly, the time required for computation of sequence similarities for deriving novel lncRNA and protein interactions is limited to single processor. This can be improved by scripting for a multi-processor environment.

Regarding lncRNA visualisation, an attempt has been made for construction of an integrated solution to provide a browser-based light-weight visualisation application. However, the current implementation has few limitations which could be addressed in the near future. One current limitation associated with the lncRNA visualisation is the lack of an online web-based visualisation suitable for content sharing with other users. As mentioned and discussed in Section 5.3, the application does not allow visualisation of multiple RNA-seq samples for comparative analysis. However, an advantage with an offline web-based version is faster generation of graphical elements and quicker navigation of the data.

Overall, the original aims/targets of the thesis have been fulfilled. It has been demonstrated that the proposed methods can be used for genome-wide identification, classification and annotation of the lncRNA and mRNA genes in plant species. The proposed methods suggest the use of a computational pipeline which is flexible and user-friendly that may find application in the area of genomics. Therefore, the proposed methods should be considered as an alternative to currently developed tools for DE mRNA and lncRNA identification, classification and function annotation where accuracy is of prime importance.

The ideas presented in this work can be further developed in several ways. Since, current implementation of the framework excludes non-coding RNAs shorter than 200 bp, one direction of future work is to integrate identification of other non-coding RNA types which include miRNA, siRNA, piRNA, snRNA, snoRNA and Circular RNA. A computational pipeline developed for the identification of DE mRNA genes was applied on *A. thaliana* apical shoot dataset. The pipeline can thus be applied on other non-model species such as Brassica Napus, however fine-tuning of the parameters for sequence alignment, transcript quantification and DGE is required.

The current work undertaken for the identification of DEGs involved sequence mapping using Tophat2 mapper (Kim *et al.*, 2013). The currently developed method can be updated by replacing Tophat2 with other mappers such as STAR (Dobin *et al.*, 2013) or HISAT2 (Kim, Langmead and Salzberg, 2015). Similarly, the differential expression analysis step can also be updated and strengthened by integrating other tools such as DESeq2 (Love, Anders and Huber, 2014). The lncRNA sub-classification and LPI analysis can be accelerated with the integration of multi-processing Python and R libraries. The algorithm can be designed to utilize multiple processors for faster computation. The function prediction analysis of lncRNA sequences using NRLMF and BMRF methods can be undertaken on larger datsets which should likely yield higher sequence similarity values between lncRNA-lncRNA and protein-protein sequences.

Development of the computational framework offers the potential to identify genome-wide lncRNA transcript sequences in model and non-model plants. Furthermore, it offers identification of several classes of lncRNAs currently unexplored in several species which could provide a catalogue of annotated lncRNA sequences similar to the currently available mammalian databases. One of the major and crucial contributions includes derivation of novel lncRNA-protein interactors, and selection of interactors based on co-expression of genes, which can potentially help in determination of lncRNA regulatory functions providing insights into their molecular mechanisms and relationships in several biological processes.

BIBLIOGRAPHY

Abdi, H. (2003) ‘Partial Least Squares (PLS) Regression’, in *Encyclopedia for research methods for the social sciences*, pp. 792–795. doi: 10.4135/9781412950589.n690.

Achawanantakun, R., Chen, J., Sun, Y. and Zhang, Y. (2015) ‘LncRNA-ID: Long non-coding RNA IDentification using balanced random forests’, *Bioinformatics*, 31(24), pp. 3897–3905. doi: 10.1093/bioinformatics/btv480.

Amit, M., Donyo, M., Hollander, D., Goren, A., Kim, E., Gelfman, S., Lev-Maor, G., Burstein, D., Schwartz, S., Postolsky, B., Pupko, T. and Ast, G. (2012) ‘Differential GC Content between Exons and Introns Establishes Distinct Strategies of Splice-Site Recognition’, *Cell Reports*, 1(5), pp. 543–556. doi: 10.1016/j.celrep.2012.03.013.

Anders, S. and Huber, W. (2010) ‘Differential expression analysis for sequence count data.’, *Genome biology*, 11(10), p. R106. doi: 10.1186/gb-2010-11-10-r106.

Anders, S., Pyl, P. T. and Huber, W. (2015) ‘HTSeq-A Python framework to work with high-throughput sequencing data’, *Bioinformatics*, 31(2), pp. 166–169. doi: 10.1093/bioinformatics/btu638.

Anders, S., Reyes, A. and Huber, W. (2012) ‘Detecting differential usage of exons from RNA-seq data’, *Genome Research*, 22(10), pp. 2008–2017. doi: 10.1101/gr.133744.111.

Ariel, F., Jegu, T., Latrasse, D., Romero-Barrios, N., Christ, A., Benhamed, M. and Crespi, M. (2014) ‘Noncoding transcription by alternative rna polymerases dynamically regulates an auxin-driven chromatin loop’, *Molecular Cell*, 55(3), pp. 383–396. doi: 10.1016/j.molcel.2014.06.011.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M. and Sherlock, G. (2000) ‘Gene ontology: Tool for the unification of biology’, *Nature Genetics*, pp. 25–29. doi: 10.1038/75556.

Avila Cobos, F., Anckaert, J., Volders, P. J., Everaert, C., Rombaut, D., Vandesompele, J., De Preter, K. and Mestdagh, P. (2017) ‘Zipper plot: Visualizing transcriptional activity of genomic regions’, *BMC Bioinformatics*, 18(1). doi: 10.1186/s12859-017-1651-7.

Bache, K. and Lichman, M. (2013) ‘UCI Machine Learning Repository’, *University of California Irvine School of Information*, p. 0. doi: University of California, Irvine, School of Information and Computer Sciences.

Bánfai, B., Jia, H., Khatun, J., Wood, E., Risk, B., Gundling, W. E., Kundaje, A., Gunawardena, H. P., Yu, Y., Xie, L., Krajewski, K., Strahl, B. D., Chen, X., Bickel, P., Giddings, M. C., Brown, J. B. and Lipovich, L. (2012) ‘Long noncoding RNAs are rarely translated in two human cell lines’, *Genome Research*. doi: 10.1101/gr.134767.111.

Bao, L. and Cui, Y. (2005) ‘Prediction of the phenotypic effects of non-synonymous single nucleotide polymorphisms using structural and evolutionary information’, *Bioinformatics*. doi: 10.1093/bioinformatics/bti365.

Bardou, F., Ariel, F., Simpson, C. G., Romero-Barrios, N., Laporte, P., Balzergue, S., Brown, J. W. S. and Crespi, M. (2014) ‘Long Noncoding RNA Modulates Alternative Splicing Regulators in Arabidopsis’, *Developmental Cell*, 30(2), pp. 166–176. doi:

10.1016/j.devcel.2014.06.017.

Barth, C. and Jander, G. (2006) ‘Arabidopsis myrosinases TGG1 and TGG2 have redundant function in glucosinolate breakdown and insect defense’, *Plant Journal*, 46(4), pp. 549–562. doi: 10.1111/j.1365-313X.2006.02716.x.

Basu, S., Kumbier, K., Brown, J. B. and Yu, B. (2018) ‘Iterative random forests to discover predictive and stable high-order interactions.’, *Proceedings of the National Academy of Sciences of the United States of America*. National Academy of Sciences, 115(8), pp. 1943–1948. doi: 10.1073/pnas.1711236115.

Bellucci, M., Agostini, F., Masin, M. and Tartaglia, G. G. (2011) ‘Predicting protein associations with long noncoding RNAs’, *Nature Methods*, pp. 444–445. doi: 10.1038/nmeth.1611.

Ben-Hur, A., Ong, C. S., Sonnenburg, S., Schölkopf, B. and Rätsch, G. (2008) ‘Support vector machines and kernels for computational biology’, *PLoS Computational Biology*. doi: 10.1371/journal.pcbi.1000173.

Benjamini, Y. and Hochberg, Y. (1995) ‘Controlling the false discovery rate: a practical and powerful approach to multiple testing’, *Journal of the Royal Statistical Society*, pp. 289–300. doi: 10.2307/2346101.

Betel, D., Wilson, M., Gabow, A., Marks, D. S. and Sander, C. (2008) ‘The microRNA.org resource: Targets and expression’, *Nucleic Acids Research*, 36(SUPPL. 1). doi: 10.1093/nar/gkm995.

Bhan, A. and Mandal, S. S. (2014) ‘Long noncoding RNAs: emerging stars in gene regulation, epigenetics and human disease’, *ChemMedChem*, pp. 1932–1956. doi: 10.1002/cmdc.201300534.

Bindea, G., Mlecnik, B., Hackl, H., Charoentong, P., Tosolini, M., Kirilovsky, A., Fridman, W. H., Pagès, F., Trajanoski, Z. and Galon, J. (2009) ‘ClueGO: A Cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks’, *Bioinformatics*, 25(8), pp. 1091–1093. doi: 10.1093/bioinformatics/btp101.

Birol, I., Jackman, S. D., Nielsen, C. B., Qian, J. Q., Varhol, R., Stazyk, G., Morin, R. D., Zhao, Y., Hirst, M., Schein, J. E., Horsman, D. E., Connors, J. M., Gascoyne, R. D., Marra, M. A. and Jones, S. J. M. (2009) ‘De novo transcriptome assembly with ABySS.’, *Bioinformatics (Oxford, England)*, 25(21), pp. 2872–7. doi: 10.1093/bioinformatics/btp367.

Bolger, A. M., Lohse, M. and Usadel, B. (2014) ‘Trimmomatic: A flexible trimmer for Illumina sequence data’, *Bioinformatics*, 30(15), pp. 2114–2120. doi: 10.1093/bioinformatics/btu170.

Boutell, M. R., Luo, J., Shen, X. and Brown, C. M. (2004) ‘Learning multi-label scene classification’, *Pattern Recognition*, 37(9), pp. 1757–1771. doi: 10.1016/j.patcog.2004.03.009.

Ter Braak, C. J. F. and Vrugt, J. A. (2008) ‘Differential Evolution Markov Chain with snooker updater and fewer chains’, *Statistics and Computing*, 18(4), pp. 435–446. doi: 10.1007/s11222-008-9104-9.

Breiman, L. (2001) ‘Random Forests’, *Machine Learning*, 45(1), pp. 5–32. doi: 10.1023/A:1010933404324.

Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (1984) *Classification and Regression Trees*, *The Wadsworth statisticsprobability series*. doi:

10.1371/journal.pone.0015807.

Buermans, H. P. J. and den Dunnen, J. T. (2014) ‘Next generation sequencing technology: Advances and applications’, *Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease*, 1842(10), pp. 1932–1941. doi: 10.1016/j.bbadis.2014.06.015.

Bulmer, M. (1988) ‘Are codon usage patterns in unicellular organisms determined by a mutation-selection balance?’, *J Evol Biol*, 1(1), pp. 15–26. doi: 10.1046/j.1420-9101.1988.1010015.x.

Cabili, M., Trapnell, C., Goff, L., Koziol, M., Tazon-Vega, B., Regev, A. and Rinn, J. L. (2011) ‘Integrative annotation of human large intergenic noncoding RNAs reveals global properties and specific subclasses’, *Genes and Development*, 25(18), pp. 1915–1927. doi: 10.1101/gad.17446611.

Cagirici, H. B., Alptekin, B. and Budak, H. (2017) ‘RNA Sequencing and Co-expressed Long Non-coding RNA in Modern and Wild Wheats’, *Scientific Reports*, 7(1). doi: 10.1038/s41598-017-11170-8.

Carninci, P., Kasukawa, T., Katayama, S., Gough, J., Frith, M. C., Maeda, N., Oyama, R., Ravasi, T., Lenhard, B., Wells, C., Kodzius, R., Shimokawa, K., Bajic, V. B., Brenner, S. E., Batalov, S., Forrest, A. R. R., Zavolan, M., Davis, M. J., Wilming, L. G., Aidinis, V., Allen, J. E., Ambesi-Impiombato, A., Apweiler, R., Aturaliya, R. N., Bailey, T. L., Bansal, M., Baxter, L., Beisel, K. W., Bersano, T., Bono, H., Chalk, A. M., Chiu, K. P., Choudhary, V., Christoffels, A., Clutterbuck, D. R., Crowe, M. L., Dalla, E., Dalrymple, B. P., de Bono, B., Della Gatta, G., di Bernardo, D., Down, T., Engstrom, P., Fagiolini, M., Faulkner, G., Fletcher, C. F., Fukushima, T., Furuno, M., Futaki, S., Gariboldi, M., Georgii-Hemming, P., Gingeras, T. R., Gojobori, T., Green, R. E., Gustincich, S., Harbers, M., Hayashi, Y., Hensch, T. K., Hirokawa, N., Hill, D., Huminiecki, L., Iacono, M., Ikeo, K., Iwama, A., Ishikawa, T., Jakt, M., Kanapin, A., Katoh, M., Kawasawa, Y., Kelso, J., Kitamura, H., Kitano, H., Kollias, G., Krishnan, S. P. T., Kruger, A., Kummerfeld, S. K., Kurochkin, I. V, Lareau, L. F., Lazarevic, D., Lipovich, L., Liu, J., Liuni, S., McWilliam, S., Madan Babu, M., Madera, M., Marchionni, L., Matsuda, H., Matsuzawa, S., Miki, H., Mignone, F., Miyake, S., Morris, K., Mottagui-Tabar, S., Mulder, N., Nakano, N., Nakauchi, H., Ng, P., Nilsson, R., Nishiguchi, S., Nishikawa, S., Nori, F., Ohara, O., Okazaki, Y., Orlando, V., Pang, K. C., Pavan, W. J., Pavesi, G., Pesole, G., Petrovsky, N., Piazza, S., Reed, J., Reid, J. F., Ring, B. Z., Ringwald, M., Rost, B., Ruan, Y., Salzberg, S. L., Sandelin, A., Schneider, C., Schönbach, C., Sekiguchi, K., Semple, C. A. M., Seno, S., Sessa, L., Sheng, Y., Shibata, Y., Shimada, H., Shimada, K., Silva, D., Sinclair, B., Sperling, S., Stupka, E., Sugiura, K., Sultana, R., Takenaka, Y., Taki, K., Tammoja, K., Tan, S. L., Tang, S., Taylor, M. S., Tegner, J., Teichmann, S. A., Ueda, H. R., van Nimwegen, E., Verardo, R., Wei, C. L., Yagi, K., Yamanishi, H., Zabarovsky, E., Zhu, S., Zimmer, A., Hide, W., Bult, C., Grimmond, S. M., Teasdale, R. D., Liu, E. T., Brusic, V., Quackenbush, J., Wahlestedt, C., Mattick, J. S., Hume, D. A., Kai, C., Sasaki, D., Tomaru, Y., Fukuda, S., Kanamori-Katayama, M., Suzuki, M., Aoki, J., Arakawa, T., Iida, J., Imamura, K., Itoh, M., Kato, T., Kawaji, H., Kawagashira, N., Kawashima, T., Kojima, M., Kondo, S., Konno, H., Nakano, K., Ninomiya, N., Nishio, T., Okada, M., Plessy, C., Shibata, K., Shiraki, T., Suzuki, S., Tagami, M., Waki, K., Watahiki, A., Okamura-Oho, Y., Suzuki, H., Kawai, J., Hayashizaki, Y., FANTOM Consortium and RIKEN Genome Exploration Research Group and Genome Science Group (Genome Network Project Core Group) (2005) ‘The transcriptional landscape of the mammalian genome.’, *Science (New York, N.Y.)*, 309(5740), pp. 1559–63. doi: 10.1126/science.1112014.

Chang, C. and Lin, C. (2011) ‘LIBSVM : A Library for Support Vector Machines’, *ACM*

Transactions on Intelligent Systems and Technology (TIST), 2, pp. 1–39. doi: 10.1145/1961189.1961199.

Chen, J., Zeng, B., Zhang, M., Xie, S., Wang, G., Hauck, A. and Lai, J. (2014) ‘Dynamic Transcriptome Landscape of Maize Embryo and Endosperm Development’, *Plant Physiology*, 166(1), pp. 252–264. doi: 10.1104/pp.114.240689.

Chen, X. (2015) ‘Predicting lncRNA-disease associations and constructing lncRNA functional similarity network based on the information of miRNA’, *Scientific Reports*, 5. doi: 10.1038/srep13186.

Chen, X., Huang, Y.-A., Wang, X.-S., You, Z.-H. and Chan, K. C. C. (2016a) ‘FMLNCSIM: fuzzy measure-based lncRNA functional similarity calculation model’, *Oncotarget*, 7(29). doi: 10.18632/oncotarget.10008.

Chen, X. and Yan, G. Y. (2013) ‘Novel human lncRNA-disease association inference based on lncRNA expression profiles’, *Bioinformatics*, 29(20), pp. 2617–2624. doi: 10.1093/bioinformatics/btt426.

Chen, X., You, Z.-H., Yan, G.-Y. and Gong, D.-W. (2016b) ‘IRWRLDA: improved random walk with restart for lncRNA-disease association prediction’, *Oncotarget*, 7(36), pp. 57919–57931. doi: 10.18632/oncotarget.11141.

Chen, Y., Pane, A. and Schüpbach, T. (2007) ‘cutoff and aubergine Mutations Result in Retrotransposon Upregulation and Checkpoint Activation in Drosophila’, *Current Biology*, 17(7), pp. 637–642. doi: 10.1016/j.cub.2007.02.027.

Chen, C., Farmer, A.-D., Langley, R.-J., Mudge, J., Crow, J.-A., May, G.-D., Huntley, J., Smith, A.-G. and Retzel, E.-F. (2010) ‘Meiosis-specific gene discovery in plants: RNA-Seq applied to isolated Arabisopsis male meiocytes’, *BMC Plant Biololgy*, 17, pp. 10-280. doi: 10.1186/1471-2229-10-280.

Chen, Y. T. and Chen, M. C. (2011) ‘Using chi-square statistics to measure similarities for text categorization’, *Expert Systems with Applications*, 38(4), pp. 3085–3090. doi: 10.1016/j.eswa.2010.08.100.

Cheng, J., Tegge, A. N. and Baldi, P. (2008) ‘Machine Learning Methods for Protein Structure Prediction’, *IEEE Reviews in Biomedical Engineering*, 1, pp. 41–49. doi: 10.1109/RBME.2008.2008239.

Clancy, S. and Brown, W. (2008) ‘Translation : DNA to mRNA to Protein’, *Nature Education*.

Clarke, B. (1970) ‘Darwinian evolution of proteins’, *Science*, 168, pp. 1009–1011. doi: 10.1126/science.168.3934.1009.

Community, N. (2011) ‘NumPy Reference’, *October*, 1(October), pp. 1–1146. doi: citeulike-article-id:11894772.

Conesa, A., Götz, S., García-Gómez, J. M., Terol, J., Talón, M. and Robles, M. (2005) ‘Blast2GO: A universal tool for annotation, visualization and analysis in functional genomics research’, *Bioinformatics*, 21(18), pp. 3674–3676. doi: 10.1093/bioinformatics/bti610.

Consortium, I. H. G. S. (2001) ‘Initial sequencing and analysis of the human genome.’, *Nature*, 409, pp. 860–921. doi: 10.1038/35057062.

Croft, D., Mundo, A. F., Haw, R., Milacic, M., Weiser, J., Wu, G., Caudy, M., Garapati, P., Gillespie, M., Kamdar, M. R., Jassal, B., Jupe, S., Matthews, L., May, B., Palatnik, S., Rothfels, K., Shamovsky, V., Song, H., Williams, M., Birney, E., Hermjakob, H., Stein, L. and D'Eustachio, P. (2014) 'The Reactome pathway knowledgebase', *Nucleic Acids Research*, 42(D1), pp. D481-7. doi: 10.1093/nar/gkt1102.

Csorba, T., Questa, J. I., Sun, Q. and Dean, C. (2014) 'Antisense *COOLAIR* mediates the coordinated switching of chromatin states at *FLC* during vernalization', *Proceedings of the National Academy of Sciences*, 111(45), pp. 16160–16165. doi: 10.1073/pnas.1419030111.

Dai, M., Thompson, R. C., Maher, C., Contreras-Galindo, R., Kaplan, M. H., Markovitz, D. M., Omenn, G. and Meng, F. (2010) 'NGSQC: cross-platform quality analysis pipeline for deep sequencing data', *BMC Genomics*, 11(Suppl 4), p. S7. doi: 10.1186/1471-2164-11-S4-S7.

van Dam, S., Võsa, U., van der Graaf, A., Franke, L. and de Magalhães, J. P. (2017) 'Gene co-expression analysis for functional classification and gene–disease predictions', *Briefings in Bioinformatics*. doi: 10.1093/bib/bbw139.

Deng, M., Zhang, K., Mehta, S., Chen, T. and Sun, F. (2002) 'Prediction of protein function using protein-protein interaction data.', *Proceedings / IEEE Computer Society Bioinformatics Conference. IEEE Computer Society Bioinformatics Conference*, 1(6), pp. 197–206. doi: 10.1089/106652703322756168.

Derrien, T., Johnson, R., Bussotti, G., Tanzer, A., Djebali, S., Tilgner, H., Guernec, G., Martin, D., Merkel, A., Knowles, D. G., Lagarde, J., Veeravalli, L., Ruan, X., Ruan, Y., Lassmann, T., Carninci, P., Brown, J. B., Lipovich, L., Gonzalez, J. M., Thomas, M., Davis, C. A., Shiekhattar, R., Gingeras, T. R., Hubbard, T. J., Notredame, C., Harrow, J. and Guigó, R. (2012) 'The GENCODE v7 catalog of human long noncoding RNAs: Analysis of their gene structure, evolution, and expression', *Genome Research*, 22(9), pp. 1775–1789. doi: 10.1101/gr.132159.111.

Deutsch, M. and Long, M. (1999) 'Intron-exon structures of eukaryotic model organisms', *Nucleic Acids Research*, 27(15), pp. 3219–3228. doi: 10.1093/nar/27.15.3219.

Dieci, G., Preti, M. and Montanini, B. (2009) 'Eukaryotic snoRNAs: A paradigm for gene expression flexibility', *Genomics*, pp. 83–88. doi: 10.1016/j.ygeno.2009.05.002.

van Dijk, E. L., Auger, H., Jaszczyszyn, Y. and Thermes, C. (2014) 'Ten years of next-generation sequencing technology', *Trends in Genetics*, 30(9), pp. 418–426. doi: 10.1016/j.tig.2014.07.001.

Ding, J., Lu, Q., Ouyang, Y., Mao, H., Zhang, P., Yao, J., Xu, C., Li, X., Xiao, J. and Zhang, Q. (2012) 'A long noncoding RNA regulates photoperiod-sensitive male sterility, an essential component of hybrid rice', *Proceedings of the National Academy of Sciences*, 109(7), pp. 2654–2659. doi: 10.1073/pnas.1121374109.

Dinger, M. E., Pang, K. C., Mercer, T. R. and Mattick, J. S. (2008) 'Differentiating protein-coding and noncoding RNA: Challenges and ambiguities', *PLoS Computational Biology*. doi: 10.1371/journal.pcbi.1000176.

Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M. and Gingeras, T. R. (2013) 'STAR: Ultrafast universal RNA-seq aligner', *Bioinformatics*, 29(1), pp. 15–21. doi: 10.1093/bioinformatics/bts635.

Du, Z., Zhou, X., Ling, Y., Zhang, Z. and Su, Z. (2010) 'agriGO: A GO analysis toolkit for the agricultural community', *Nucleic Acids Research*, 38(SUPPL. 2). doi: 10.1093/nar/gkq310.

Durgabai, R. P. L. (2014) ‘Feature Selection using ReliefF Algorithm’, *International Journal of Advanced Research in Computer and Communication Engineering*, 3(10), pp. 8215–8218. Available at: www.ijarcce.com.

Dyer, B. D., Kahn, M. J. and Leblanc, M. D. (2008) ‘Classification and regression tree (CART) analyses of genomic signatures reveal sets of tetramers that discriminate temperature optima of archaea and bacteria’, *Archaea*. doi: 10.1155/2008/829730.

Dykes, I. M. and Emanueli, C. (2017) ‘Transcriptional and Post-transcriptional Gene Regulation by Long Non-coding RNA’, *Genomics, Proteomics and Bioinformatics*. doi: 10.1016/j.gpb.2016.12.005.

Efroymson, M. A. (1960) ‘Multiple regression analysis’, *Mathematical methods for digital computers*, 1, pp. 191–203. doi: 10.1177/1753193411414639.

Eid, J., Fehr, A., Gray, J., Luong, K., Lyle, J., Otto, G., Peluso, P., Rank, D., Baybayan, P., Bettman, B., Bibillo, A., Bjornson, K., Chaudhuri, B., Christians, F., Cicero, R., Clark, S., Dalal, R., DeWinter, A., Dixon, J., Foquet, M., Gaertner, A., Hardenbol, P., Heiner, C., Hester, K., Holden, D., Kearns, G., Kong, X., Kuse, R., Lacroix, Y., Lin, S., Lundquist, P., Ma, C., Marks, P., Maxham, M., Murphy, D., Park, I., Pham, T., Phillips, M., Roy, J., Sebra, R., Shen, G., Sorenson, J., Tomaney, A., Travers, K., Trulson, M., Vieceli, J., Wegener, J., Wu, D., Yang, A., Zaccarin, D., Zhao, P., Zhong, F., Korlach, J. and Turner, S. (2009) ‘Real-time DNA sequencing from single polymerase molecules’, *Science*, 323(5910), pp. 133–138. doi: 10.1126/science.1162986.

ENCODE Consortium (2007) ‘Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project.’, *Nature*, 447(7146), pp. 799–816. doi: 10.1038/nature05874.

Fan, X. N. and Zhang, S. W. (2015) ‘lncRNA-MFDL: identification of human long non-coding RNAs by fusing multiple features and using deep learning’, *Molecular Biosystems*, 11(3), pp. 892–897. doi: 10.1039/c4mb00650j.

Fernández-Suárez, X. M. and Schuster, M. K. (2010) ‘Using the ensembl genome server to browse genomic sequence data’, *Current Protocols in Bioinformatics*. doi: 10.1002/0471250953.bi0115s16.

Fickett, J. W. (1982) ‘Recognition of protein coding regions in DNA sequences’, *Nucleic Acids Research*, 10(17), pp. 5303–5318. doi: 10.1093/nar/10.17.5303.

Fickett, J. W. and Tung, C. S. (1992) ‘Assessment of protein coding measures.’, *Nucleic acids research*, 20(24), pp. 6441–6450. doi: 10.1093/nar/20.24.6441.

Franco-Zorrilla, J. M., Valli, A., Todesco, M., Mateos, I., Puga, M. I., Rubio-Somoza, I., Leyva, A., Weigel, D., García, J. A. and Paz-Ares, J. (2007) ‘Target mimicry provides a new mechanism for regulation of microRNA activity’, *Nature Genetics*, 39(8), pp. 1033–1037. doi: 10.1038/ng2079.

Freese, N. H., Norris, D. C. and Loraine, A. E. (2016) ‘Integrated genome browser: Visual analytics platform for genomics’, *Bioinformatics*, 32(14), pp. 2089–2095. doi: 10.1093/bioinformatics/btw069.

Friedman, J. H. and Popescu, B. E. (2008) ‘Predictive learning via rule ensembles’, *Annals of Applied Statistics*, 2(3), pp. 916–954. doi: 10.1214/07-AOAS148.

Friedman, R. C., Farh, K. K. H., Burge, C. B. and Bartel, D. P. (2009) ‘Most mammalian mRNAs are conserved targets of microRNAs', *Genome Research*, 19(1), pp. 92–105. doi: 10.1101/gr.082701.108.

Frith, M. C., Forrest, A. R., Nourbakhsh, E., Pang, K. C., Kai, C., Kawai, J., Carninci, P., Hayashizaki, Y., Bailey, T. L. and Grimmond, S. M. (2006) 'The abundance of short proteins in the mammalian proteome', *PLoS Genetics*, 2(4), pp. 515–528. doi: 10.1371/journal.pgen.0020052.

Galperin, M. Y. (2008) 'The molecular biology database collection: 2008 update', *Nucleic Acids Research*, 36(SUPPL. 1). doi: 10.1093/nar/gkm1037.

Gehrmann, T., Loog, M., Reinders, M. J. T. and De Ridder, D. (no date) 'Conditional Random Fields for Protein Function Prediction'.

Geman, S. and Geman, D. (1984) 'Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6), pp. 721–741. doi: 10.1109/TPAMI.1984.4767596.

Genome.gov (2014) *A Brief Guide to Genomics - DNA, Genes and Genomes*, *Genome.gov*.

Geyer, C. J. (1991) 'Markov Chain Monte Carlo Maximum Likelihood', *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, (1), pp. 156–163.

Gligorijević, V., Barot, M. and Bonneau, R. (2018) 'deepNF: Deep network fusion for protein function prediction', *Bioinformatics*, p. bty440. doi: 10.1093/bioinformatics/bty440.

Goldstein, B. A., Hubbard, A. E., Cutler, A. and Barcellos, L. F. (2010) 'An application of Random Forests to a genome-wide association dataset: methodological considerations & new findings.', *BMC genetics*. doi: 10.1186/1471-2156-11-49.

Gong, Y., Huang, H. T., Liang, Y., Trimarchi, T., Aifantis, I. and Tsirigos, A. (2017) 'lncRNA-screen: An interactive platform for computationally screening long non-coding RNAs in large genomics datasets', *BMC Genomics*, 18(1). doi: 10.1186/s12864-017-3817-0.

Goodall, G. J. and Filipowicz, W. (1990) 'The minimum functional length of pre-mRNA introns in monocots and dicots', *Plant Molecular Biology*, 14(5), pp. 727–733. doi: 10.1007/BF00016505.

Google.com (2014) 'Google Maps JavaScript API v3', *Google Developers*, pp. 1–141. Available at: https://developers.google.com/maps/documentation/javascript/.

Grabherr, M. G., Haas, B. J., Yassour, M., Levin, J. Z., Thompson, D. A., Amit, I., Adiconis, X., Fan, L., Raychowdhury, R., Zeng, Q., Chen, Z., Mauceli, E., Hacohen, N., Gnirke, A., Rhind, N., Di Palma, F., Birren, B. W., Nusbaum, C., Lindblad-Toh, K., Friedman, N. and Regev, A. (2011) 'Full-length transcriptome assembly from RNA-Seq data without a reference genome', *Nature Biotechnology*, 29(7), pp. 644–652. doi: 10.1038/nbt.1883.

Granitto, P. M., Furlanello, C., Biasioli, F. and Gasperi, F. (2006) 'Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products', *Chemometrics and Intelligent Laboratory Systems*, 83(2), pp. 83–90. doi: 10.1016/j.chemolab.2006.01.007.

Guo, Q., Cheng, Y., Liang, T., He, Y., Ren, C., Sun, L. and Zhang, G. (2015) 'Comprehensive analysis of lncRNA-mRNA co-expression patterns identifies immune-associated lncRNA biomarkers in ovarian cancer malignant progression', *Scientific Reports*, 5. doi: 10.1038/srep17683.

Guttman, M., Amit, I., Garber, M., French, C., Lin, M. F., Feldser, D., Huarte, M., Zuk, O., Carey, B. W., Cassady, J. P., Cabili, M. N., Jaenisch, R., Mikkelsen, T. S., Jacks, T., Hacohen, N., Bernstein, B. E., Kellis, M., Regev, A., Rinn, J. L. and Lander, E. S. (2009) ‘Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals’, *Nature*, 458(7235), pp. 223–227. doi: 10.1038/nature07672.

Guyon, I. and Elisseeff, A. (2003) ‘An Introduction to Variable and Feature Selection’, *Journal of Machine Learning Research (JMLR)*, 3(3), pp. 1157–1182. doi: 10.1016/j.aca.2011.07.027.

Haas, B. J., Papanicolaou, A., Yassour, M., Grabherr, M., Blood, P. D., Bowden, J., Couger, M. B., Eccles, D., Li, B., Lieber, M., Macmanes, M. D., Ott, M., Orvis, J., Pochet, N., Strozzi, F., Weeks, N., Westerman, R., William, T., Dewey, C. N., Henschel, R., Leduc, R. D., Friedman, N. and Regev, A. (2013) ‘De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis’, *Nature Protocols*, 8(8), pp. 1494–1512. doi: 10.1038/nprot.2013.084.

Hajjari, M. and Salavaty, A. (2015) ‘HOTAIR: an oncogenic long non-coding RNA in different cancers.’, *Cancer biology & medicine*, 12(1), pp. 1–9. doi: 10.7497/j.issn.2095-3941.2015.0006.

Hall, M. (1999) ‘Correlation-based Feature Selection for Machine Learning’, *Methodology*, 21i195-i20(April), pp. 1–5. doi: 10.1.1.149.3848.

Hall, M. and Smith, L. (1998) ‘Practical feature subset selection for machine learning’, *Computer Science*, 98, pp. 181–191.

Hao, Y., Wu, W., Shi, F., Dalmolin, R. J. S., Yan, M., Tian, F., Chen, X., Chen, G. and Cao, W. (2015) ‘Prediction of long noncoding RNA functions with co-expression network in esophageal squamous cell carcinoma’, *BMC Cancer*, 15(1). doi: 10.1186/s12885-015-1179-z.

Hardcastle, T. J. and Kelly, K. A. (2010) ‘baySeq: empirical Bayesian methods for identifying differential expression in sequence count data.’, *BMC bioinformatics*, 11(1), p. 422. doi: 10.1186/1471-2105-11-422.

Harikumar, G. and Bresler, Y. (1996) ‘A new algorithm for computing sparse solutions to linear inverse problems’, in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, pp. 1331–1334 vol. 3. doi: 10.1109/ICASSP.1996.543672.

Harrow, J., Frankish, A., Gonzalez, J. M., Tapanari, E., Diekhans, M., Kokocinski, F., Aken, B. L., Barrell, D., Zadissa, A., Searle, S., Barnes, I., Bignell, A., Boychenko, V., Hunt, T., Kay, M., Mukherjee, G., Rajan, J., Despacio-Reyes, G., Saunders, G., Steward, C., Harte, R., Lin, M., Howald, C., Tanzer, A., Derrien, T., Chrast, J., Walters, N., Balasubramanian, S., Pei, B., Tress, M., Rodriguez, J. M., Ezkurdia, I., Van Baren, J., Brent, M., Haussler, D., Kellis, M., Valencia, A., Reymond, A., Gerstein, M., Guigó, R. and Hubbard, T. J. (2012) ‘GENCODE: The reference human genome annotation for the ENCODE project’, *Genome Research*, 22(9), pp. 1760–1774. doi: 10.1101/gr.135350.111.

Hastie, T., Tibshirani, R. and Friedman, J. (2009) ‘The Elements of Statistical Learning’, *Elements*, 1, pp. 337–387. doi: 10.1007/b94608.

Heather, J. M. and Chain, B. (2016) ‘The sequence of sequencers: The history of sequencing DNA’, *Genomics*. doi: 10.1016/j.ygeno.2015.11.003.

Heo, J. B. and Sung, S. (2011) ‘Vernalization-mediated epigenetic silencing by a long intronic noncoding RNA’, *Science*, 331(6013), pp. 76–79. doi: 10.1126/science.1197349.

Hou, M., Tang, X., Tian, F., Shi, F., Liu, F. and Gao, G. (2016) ‘AnnoLnc: A web server for systematically annotating novel human lncRNAs’, *BMC Genomics*. doi: 10.1186/s12864-016-3287-9.

Hu, L., Xu, Z., Hu, B. and Lu, Z. J. (2016) ‘COME: a robust coding potential calculation tool for lncRNA identification and characterization based on multiple features.’, *Nucleic acids research*, p. gkw798. doi: 10.1093/nar/gkw798.

Huang, M. L., Hung, Y. H., Lee, W. M., Li, R. K. and Jiang, B. R. (2014) ‘SVM-RFE based feature selection and taguchi parameters optimization for multiclass SVM Classifier’, *Scientific World Journal*, 2014. doi: 10.1155/2014/795624.

Huang, X. and Yang, S.-P. (2005) ‘Generating a genome assembly with PCAP.’, *Current protocols in bioinformatics*, Chapter 11, p. Unit11.3. doi: 10.1002/0471250953.bi1103s11.

Huarte, M. (2015) ‘The emerging role of lncRNAs in cancer’, *Nature Medicine*, pp. 1253–1261. doi: 10.1038/nm.3981.

Ikemura, T. (1982) ‘Correlation between the abundance of yeast transfer RNAs and the occurrence of the respective codons in protein genes’, *Journal of Molecular Biology*, 158(4), pp. 573–597. doi: 10.1016/0022-2836(82)90250-9.

Ikemura, T. (1985) ‘Codon usage and tRNA content in unicellular and multicellular organisms.’, *Molecular Biology and Evolution*, 2(1), pp. 13–34. doi: 10.1093/oxfordjournals.molbev.a040335.

Illumina (2010) ‘Illumina sequencing technology’, *Technology Spotlight: Illumina Sequencing*, pp. 1–5. doi: 10.1016/S0167-7799(03)00189-6.

Jabnoune, M., Secco, D., Lecampion, C., Robaglia, C., Shu, Q. and Poirier, Y. (2013) ‘A Rice cis-Natural Antisense RNA Acts as a Translational Enhancer for Its Cognate mRNA and Contributes to Phosphate Homeostasis and Plant Fitness’, *The Plant Cell*, 25(10), pp. 4166–4182. doi: 10.1105/tpc.113.116251.

Jacqmard, A., De Veylder, L., Segers, G., De Almeida Engler, J., Bernier, G., Van Montagu, M. and Inze, D. (1999) ‘Expression of CKS1At in Arabidopsis thaliana indicates a role for the protein in both the mitotic and the endoreduplication cycle’, *Planta*, 207(4), pp. 496–504. doi: 10.1007/s004250050509.

Jensen, L. M., Jepsen, H. S. K., Halkier, B. A., Kliebenstein, D. J. and Burow, M. (2015) ‘Natural variation in cross-talk between glucosinolates and onset of flowering in Arabidopsis’, *Frontiers in Plant Science*, 6(SEPTEMBER), p. 697. doi: 10.3389/fpls.2015.00697.

Jiang, P., Wu, H., Wang, W., Ma, W., Sun, X. and Lu, Z. (2007) ‘MiPred: Classification of real and pseudo microRNA precursors using random forest prediction model with combined features’, *Nucleic Acids Research*. doi: 10.1093/nar/gkm368.

Jiang, Q., Ma, R., Wang, J., Wu, X., Jin, S., Peng, J., Tan, R., Zhang, T., Li, Y. and Wang, Y. (2015) ‘LncRNA2Function: a comprehensive resource for functional investigation of human lncRNAs based on RNA-seq data’, *BMC Genomics*, 16(Suppl 3), p. S2. doi: 10.1186/1471-2164-16-S3-S2.

Jin, J., Liu, J., Wang, H., Wong, L. and Chua, N. H. (2013) ‘PLncDB: Plant long non-coding

RNA database', *Bioinformatics*, pp. 1068–1071. doi: 10.1093/bioinformatics/btt107.

Jinek, M. and Doudna, J. A. (2009) 'A three-dimensional view of the molecular machinery of RNA interference', *Nature*, pp. 405–412. doi: 10.1038/nature07755.

Johnson, W. E., Li, C. and Rabinovic, A. (2007) 'Adjusting batch effects in microarray expression data using empirical Bayes methods', *Biostatistics*, 8(1), pp. 118–127. doi: 10.1093/biostatistics/kxj037.

Kang, Y. J., Yang, D. C., Kong, L., Hou, M., Meng, Y. Q., Wei, L. and Gao, G. (2017) 'CPC2: A fast and accurate coding potential calculator based on sequence intrinsic features', *Nucleic Acids Research*. doi: 10.1093/nar/gkx428.

Karlin, S. and Mrázek, J. (1996) 'What drives codon choices in human genes?', *Journal of molecular biology*, 262(4), pp. 459–72. doi: 10.1006/jmbi.1996.0528.

Kent, W. J., Sugnet, C. W., Furey, T. S., Roskin, K. M., Pringle, T. H., Zahler, A. M. and Haussler, a. D. (2002) 'The Human Genome Browser at UCSC', *Genome Research*, 12(6), pp. 996–1006. doi: 10.1101/gr.229102.

Kim, D. H., Xi, Y. and Sung, S. (2017) 'Modular function of long noncoding RNA, COLDAIR, in the vernalization response', *PLoS Genetics*, 13(7). doi: 10.1371/journal.pgen.1006939.

Kim, D., Langmead, B. and Salzberg, S. L. (2015) 'HISAT: A fast spliced aligner with low memory requirements', *Nature Methods*, 12(4), pp. 357–360. doi: 10.1038/nmeth.3317.

Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R. and Salzberg, S. L. (2013) 'TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions.', *Genome biology*, 14(4), p. R36. doi: 10.1186/gb-2013-14-4-r36.

Klepikova, A. V, Logacheva, M. D., Dmitriev, S. E. and Penin, A. A. (2015) 'RNA-seq analysis of an apical meristem time series reveals a critical point in Arabidopsis thaliana flower initiation', *BMC Genomics*, 16. doi: 10.1186/s12864-015-1688-9.

Kong, L., Zhang, Y., Ye, Z. Q., Liu, X. Q., Zhao, S. Q., Wei, L. and Gao, G. (2007) 'CPC: Assess the protein-coding potential of transcripts using sequence features and support vector machine', *Nucleic Acids Research*, 35(SUPPL.2). doi: 10.1093/nar/gkm391.

Kourmpetis, Y. A. I., Van Dijk, A. D. J., Bink, M. C. A. M., Van Ham, R. C. H. J. and Ter Braak, C. J. F. (2010) 'Bayesian markov random field analysis for protein function prediction based on network data', *PLoS ONE*, 5(2), p. e9293. doi: 10.1371/journal.pone.0009293.

Krishnakumar, V., Hanlon, M. R., Contrino, S., Ferlanti, E. S., Karamycheva, S., Kim, M., Rosen, B. D., Cheng, C. Y., Moreira, W., Mock, S. A., Stubbs, J., Sullivan, J. M., Krampis, K., Miller, J. R., Micklem, G., Vaughn, M. and Town, C. D. (2015) 'Araport: The Arabidopsis Information Portal', *Nucleic Acids Research*, 43(D1), pp. D1003–D1009. doi: 10.1093/nar/gku1200.

Kutmon, M., Riutta, A., Nunes, N., Hanspers, K., Willighagen, E. L., Bohler, A., Mélius, J., Waagmeester, A., Sinha, S. R., Miller, R., Coort, S. L., Cirillo, E., Smeets, B., Evelo, C. T. and Pico, A. R. (2015) 'WikiPathways: capturing the full diversity of pathway knowledge', *Nucleic Acids Research*, 44(October 2015), p. gkv1024. doi: 10.1093/nar/gkv1024.

Langmead, B. and Salzberg, S. L. (2012) 'Fast gapped-read alignment with Bowtie 2', *Nat Methods*, 9(4), pp. 357–359. doi: 10.1038/nmeth.1923.

Langmead, B., Trapnell, C., Pop, M. and Salzberg, S. (2009) ‘Ultrafast and memory-efficient alignment of short DNA sequences to the human genome’, *Genome biology*, 10(3), p. R25. doi: 10.1186/gb-2009-10-3-r25.

Law, M. H., Jain, A. K. and Figueiredo, M. A. T. (2000) ‘Feature Selection in Mixture-Based Clustering’, in *Advances in neural information processing systems*, pp. 625–632.

Lee, C. and Lee, G. G. (2006a) ‘Information gain and divergence-based feature selection for machine learning-based text categorization’, *Information Processing and Management*, pp. 155–165. doi: 10.1016/j.ipm.2004.08.006.

Lee, H., Tu, Z., Deng, M., Sun, F. and Chen, T. (2006b) ‘Diffusion Kernel-Based Logistic Regression Models for Protein Function Prediction’, *OMICS: A Journal of Integrative Biology*, 10(1), pp. 40–55. doi: 10.1089/omi.2006.10.40.

Leng, N., Dawson, J. A., Thomson, J. A., Ruotti, V., Rissman, A. I., Smits, B. M. G., Haag, J. D., Gould, M. N., Stewart, R. M. and Kendziorski, C. (2013) ‘EBSeq: An empirical Bayes hierarchical model for inference in RNA-seq experiments’, *Bioinformatics*, 29(8), pp. 1035–1043. doi: 10.1093/bioinformatics/btt087.

Li, A., Zhang, J. and Zhou, Z. (2014a) ‘PLEK: a tool for predicting long non-coding RNAs and messenger RNAs based on an improved k-mer scheme’, *BMC Bioinformatics*, 15(1), p. 311. doi: 10.1186/1471-2105-15-311.

Li, B. and Dewey, C. N. (2011) ‘RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome’, *BMC Bioinformatics*, 12(1), p. 323. doi: 10.1186/1471-2105-12-323.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G. and Durbin, R. (2009) ‘The Sequence Alignment/Map format and SAMtools’, *Bioinformatics*, 25(16), pp. 2078–2079. doi: 10.1093/bioinformatics/btp352.

Li, J., Witten, D. M., Johnstone, I. M. and Tibshirani, R. (2012) ‘Normalization, testing, and false discovery rate estimation for RNA-sequencing data’, *Biostatistics*, 13(3), pp. 523–538. doi: 10.1093/biostatistics/kxr031.

Li, S., Yu, X., Lei, N., Cheng, Z., Zhao, P., He, Y., Wang, W. and Peng, M. (2017) ‘Genome-wide identification and functional prediction of cold and/or drought-responsive lncRNAs in cassava’, *Scientific Reports*, 7. doi: 10.1038/srep45981.

Li, Y., Qiu, C., Tu, J., Geng, B., Yang, J., Jiang, T. and Cui, Q. (2014b) ‘HMDD v2.0: A database for experimentally supported human microRNA and disease associations’, *Nucleic Acids Research*, 42(D1). doi: 10.1093/nar/gkt1023.

Lin, M. F., Jungreis, I. and Kellis, M. (2011) ‘PhyloCSF: A comparative genomics method to distinguish protein coding and non-coding regions’, *Bioinformatics*, 27(13). doi: 10.1093/bioinformatics/btr209.

Lingyan, S., Pique-Regi, R., Asgharzadeh, S. and Ortega, A. (2009) ‘Microarray classification using block diagonal linear discriminant analysis with embedded feature selection’, in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 1757–1760.

Liu, C., Muchhal, U. S. and Raghothama, K. G. (1997) ‘Differential expression of TPS11, a phosphate starvation-induced gene in tomato’, *Plant Molecular Biology*, 33(5), pp. 867–874. doi: 10.1023/A:1005729309569.

Liu, H., Ren, G., Hu, H., Zhang, L., Ai, H., Zhang, W. and Zhao, Q. (2017) 'LPI-NRLMF: lncRNA-protein interaction prediction by neighborhood regularized logistic matrix factorization'. Available at: www.impactjournals.com/oncotarget.

Liu, W., Li, L. and Li, W. (2014) 'Gene co-expression analysis identifies common modules related to prognosis and drug resistance in cancer cell lines', *International Journal of Cancer*. doi: 10.1002/ijc.28935.

Liu, X., Hao, L., Li, D., Zhu, L. and Hu, S. (2015) 'Long Non-coding RNAs and Their Biological Roles in Plants', *Genomics, Proteomics & Bioinformatics*, 13(3), pp. 137–147. doi: 10.1016/j.gpb.2015.02.003.

Liu, X. L. (2017) 'Deep Recurrent Neural Network for Protein Function Prediction from Sequence', *arXiv*, pp. 1–38. doi: 10.1101/103994.

Liu, Y., Wu, M., Miao, C., Zhao, P. and Li, X. L. (2016) 'Neighborhood Regularized Logistic Matrix Factorization for Drug-Target Interaction Prediction', *PLoS Computational Biology*. doi: 10.1371/journal.pcbi.1004760.

Love, M. I., Anders, S. and Huber, W. (2014) *Differential analysis of count data - the DESeq2 package*, *Genome Biology*. doi: 110.1186/s13059-014-0550-8.

Lu, M., Shi, B., Wang, J., Cao, Q. and Cui, Q. (2010) 'TAM: A method for enrichment and depletion analysis of a microRNA category in a list of microRNAs', *BMC Bioinformatics*, 11. doi: 10.1186/1471-2105-11-419.

Ma, L., Bajic, V. B. and Zhang, Z. (2013) 'On the classification of long non-coding RNAs', *RNA Biology*, pp. 924–933. doi: 10.4161/rna.24604.

Mardis, E. R. (2017) 'DNA sequencing technologies: 2006-2016', *Nature Protocols*. doi: 10.1038/nprot.2016.182.

Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M. and Gilad, Y. (2008) 'RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays', *Genome Research*, 18(9), pp. 1509–1517. doi: 10.1101/gr.079558.108.

Marquardt, D. W. (1970) 'Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation', *Technometrics*, 12(3), pp. 591–612. doi: 10.1080/00401706.1970.10488699.

Martin, M. (2011) 'Cutadapt removes adapter sequences from high-throughput sequencing reads', *EMBnet.journal*, 17(1), p. 10. doi: 10.14806/ej.17.1.200.

McCarthy, D. J., Campbell, K. R., Lun, A. T. L. and Wills, Q. F. (2017) 'Scater: Pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R', *Bioinformatics*, 33(8), pp. 1179–1186. doi: 10.1093/bioinformatics/btw777.

McHugh, C. A., Chen, C.-K., Chow, A., Surka, C. F., Tran, C., McDonel, P., Pandya-Jones, A., Blanco, M., Burghard, C., Moradian, A., Sweredoski, M. J., Shishkin, A. A., Su, J., Lander, E. S., Hess, S., Plath, K. and Guttman, M. (2015) 'The Xist lncRNA interacts directly with SHARP to silence transcription through HDAC3', *Nature*, 521(7551), pp. 232–236. doi: 10.1038/nature14443.

Meinshausen, N. (2009) 'Forest Garrote', *Methods*, (2007), p. 16. doi: 10.1214/09-EJS434.

Meinshausen, N. (2010) 'Node harvest', *Annals of Applied Statistics*, 4(4), pp. 2049–2072.

doi: 10.1214/10-AOAS367.

Meister, G. and Tuschi, T. (2004) ‘Mechanisms of gene silencing by double-stranded RNA’, *Nature*, 431(September), pp. 343–349. doi: 10.1038/nature02873.

Merriman, B., Torrent, I. and Rothberg, J. M. (2012) ‘Progress in Ion Torrent semiconductor chip based sequencing’, *Electrophoresis*, pp. 3397–3417. doi: 10.1002/elps.201200424.

Michaels, S. D. (1999) ‘FLOWERING LOCUS C Encodes a Novel MADS Domain Protein That Acts as a Repressor of Flowering’, *The Plant Cell*, 11(5), pp. 949–956. doi: 10.1105/tpc.11.5.949.

Mitra, S. A., Mitra, A. P. and Triche, T. J. (2012) ‘A central role for long non-coding RNA in cancer’, *Frontiers in Genetics*. doi: 10.3389/fgene.2012.00017.

Moazed, D. (2009) ‘Small RNAs in transcriptional gene silencing and genome defence’, *Nature*, pp. 413–420. doi: 10.1038/nature07756.

Mohammadin, S., Nguyen, T.-P., van Weij, M. S., Reichelt, M. and Schranz, M. E. (2017) ‘Flowering Locus C (FLC) Is a Potential Major Regulator of Glucosinolate Content across Developmental Stages of Aethionema arabicum (Brassicaceae)’, *Frontiers in Plant Science*, 8, p. 876. doi: 10.3389/fpls.2017.00876.

Montojo, J., Zuberi, K., Rodriguez, H., Kazi, F., Wright, G., Donaldson, S. L., Morris, Q. and Bader, G. D. (2010) ‘GeneMANIA cytoscape plugin: Fast gene function predictions on the desktop’, *Bioinformatics*, 26(22), pp. 2927–2928. doi: 10.1093/bioinformatics/btq562.

Moore, M. J. (2005) ‘From birth to death: The complex lives of eukaryotic mRNAs’, *Science*, pp. 1514–1518. doi: 10.1126/science.1111443.

Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L. and Wold, B. (2008) ‘Mapping and quantifying mammalian transcriptomes by RNA-Seq’, *Nature Methods*, 5(7), pp. 621–628. doi: 10.1038/nmeth.1226.

Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C. and Morris, Q. (2008) ‘GeneMANIA: A real-time multiple association network integration algorithm for predicting gene function’, *Genome Biology*, 9(SUPPL. 1). doi: 10.1186/gb-2008-9-s1-s4.

Müllner, D. (2013) ‘fastcluster : Fast Hierarchical , Agglomerative’, *Journal of Statistical Software*, 53(9), pp. 1–18. doi: 10.18637/jss.v053.i09.

Mundry, R. and Nunn, C. L. (2009) ‘Stepwise Model Fitting and Statistical Inference: Turning Noise into Signal Pollution’, *The American Naturalist*, 173(1), pp. 119–123. doi: 10.1086/593303.

Muppirala, U. K., Honavar, V. G. and Dobbs, D. (2011) ‘Predicting RNA-Protein Interactions Using Only Sequence Information’, *BMC Bioinformatics*, 12(1). doi: 10.1186/1471-2105-12-489.

Nacher, J. C. and Akutsu, T. (2007) ‘Recent progress on the analysis of power-law features in complex cellular networks’, *Cell Biochemistry and Biophysics*, pp. 37–47. doi: 10.1007/s12013-007-0040-7.

Nakano, M., Komatsu, J., Matsuura, S. I., Takashima, K., Katsura, S. and Mizuno, A. (2003) ‘Single-molecule PCR using water-in-oil emulsion’, *Journal of Biotechnology*, 102(2), pp. 117–124. doi: 10.1016/S0168-1656(03)00023-3.

Naur, P., Petersen, B. L., Mikkelsen, M. D., Bak, S., Rasmussen, H., Olsen, C. E. and Halkier, B. A. (2003) ‘CYP83A1 and CYP83B1, two nonredundant cytochrome P450 enzymes metabolizing oximes in the biosynthesis of glucosinolates in Arabidopsis.’, *Plant physiology*, 133(1), pp. 63–72. doi: 10.1104/pp.102.019240.

Nelder, J. A. and Wedderburn, R. W. M. (1972) ‘Generalized Linear Models’, *J. R. Statist. Soc. A.*, 135(3), pp. 370–384. doi: 10.1080/01621459.2000.10474340.

O’Donovan, C., Martin, M. J., Gattiker, A., Gasteiger, E., Bairoch, A. and Apweiler, R. (2002) ‘High-quality protein knowledge resource: SWISS-PROT and TrEMBL.’, *Briefings in bioinformatics*. doi: 10.1093/bib/3.3.275.

Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H. and Kanehisa, M. (1999) ‘KEGG: Kyoto encyclopedia of genes and genomes’, *Nucleic Acids Research*, pp. 29–34. doi: 10.1093/nar/27.1.29.

Orman-Ligeza, B., Parizot, B., Gantet, P. P., Beeckman, T., Bennett, M. J. and Draye, X. (2013) ‘Post-embryonic root organogenesis in cereals: Branching out from model plants’, *Trends in Plant Science*, pp. 1360–1385. doi: 10.1016/j.tplants.2013.04.010.

Ostell, J. and McEntyre, J. (2007) ‘The NCBI Handbook’, *NCBI Bookshelf*, pp. 1–8. doi: 10.4016/12837.01.

Pagano, A., Castelnuovo, M., Tortelli, F., Ferrari, R., Dieci, G. and Cancedda, R. (2007) ‘New small nuclear RNA gene-like transcriptional units as sources of regulatory transcripts’, *PLoS Genetics*, 3(2), pp. 0174–0184. doi: 10.1371/journal.pgen.0030001.

Pauli, A., Rinn, J. L. and Schier, A. F. (2011) ‘Non-coding RNAs as regulators of embryogenesis’, *Nature Reviews Genetics*, pp. 136–149. doi: 10.1038/nrg2904.

Pauli, A., Valen, E., Lin, M. F., Garber, M., Vastenhouw, N. L., Levin, J. Z., Fan, L., Sandelin, A., Rinn, J. L., Regev, A. and Schier, A. F. (2012) ‘Systematic identification of long noncoding RNAs expressed during zebrafish embryogenesis’, *Genome Research*, 22(3), pp. 577–591. doi: 10.1101/gr.133009.111.

Pearson, W. R. (1991) ‘Searching protein sequence libraries: Comparison of the sensitivity and selectivity of the Smith-Waterman and FASTA algorithms’, *Genomics*. doi: 10.1016/0888-7543(91)90071-L.

Pedregosa, F. and Varoquaux, G. (2011) *Scikit-learn: Machine learning in Python*, *… of Machine Learning …*. doi: 10.1007/s13398-014-0173-7.2.

Peled, S., Leiderman, O., Charar, R., Efroni, G., Shav-Tal, Y. and Ofran, Y. (2016) ‘De-novo protein function prediction using DNA binding and RNA binding proteins as a test case’, *Nature Communications*, 7. doi: 10.1038/ncomms13424.

Peng, H., Ding, C. and Long, F. (2005a) ‘Minimum redundancy-maximum relevance feature selection’, *IEEE Intelligent Systems*, pp. 70–71. doi: 10.1111/j.0307-6946.2005.00650.x.

Peng, H., Long, F. and Ding, C. (2005b) ‘Feature selection based on mutual information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy’, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8), pp. 1226–1238. doi: 10.1109/TPAMI.2005.159.

Penny, G. D., Kay, G. F., Sheardown, S. A., Rastan, S. and Brockdorff, N. (1996) ‘Requirement for Xist in X chromosome inactivation’, *Nature*, 379(6561), pp. 131–137. doi: 10.1038/379131a0.

Perez, M. and Marwala, T. (2012) ‘Microarray data feature selection using hybrid genetic algorithm simulated annealing’, in *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel, IEEEI 2012*. doi: 10.1109/EEEI.2012.6377146.

Perron, U., Provero, P. and Molineris, I. (2017) ‘In silico prediction of lncRNA function using tissue specific and evolutionary conserved expression’, *BMC Bioinformatics*, 18. doi: 10.1186/s12859-017-1535-x.

Phillips, T. (2008) ‘The role of methylation in gene expression’, *Nature Education*, 1, p. 1.

Pian, C., Zhang, G., Chen, Z., Chen, Y., Zhang, J., Yang, T. and Zhang, L. (2016) ‘LncRNApred: Classification of long non-coding RNAs and protein-coding transcripts by the ensemble algorithm with a new hybrid feature’, *PLoS ONE*, 11(5). doi: 10.1371/journal.pone.0154567.

Prountzos, D. and Pingali, K. (2013) ‘Betweenness centrality’, in *Proceedings of the 18th ACM SIGPLAN symposium on Principles and practice of parallel programming - PPoPP ’13*, p. 35. doi: 10.1145/2442516.2442521.

Pruitt, K. D., Tatusova, T. and Maglott, D. R. (2007) ‘NCBI reference sequences (RefSeq): A curated non-redundant sequence database of genomes, transcripts and proteins’, *Nucleic Acids Research*, 35(SUPPL. 1). doi: 10.1093/nar/gkl842.

Pudil, P., Novovičová, J. and Kittler, J. (1994) ‘Floating search methods in feature selection’, *Pattern Recognition Letters*, 15(11), pp. 1119–1125. doi: 10.1016/0167-8655(94)90127-9.

Qu, S., Yang, X., Li, X., Wang, J., Gao, Y., Shang, R., Sun, W., Dou, K. and Li, H. (2015) ‘Circular RNA: A new star of noncoding RNAs’, *Cancer Letters*, pp. 141–148. doi: 10.1016/j.canlet.2015.06.003.

R Development Core Team (2016) ‘R: A Language and Environment for Statistical Computing’, *R Foundation for Statistical Computing Vienna Austria*, 0, p. {ISBN} 3-900051-07-0. doi: 10.1038/sj.hdy.6800737.

Rangan, P., Malone, C. D., Navarro, C., Newbold, S. P., Hayes, P. S., Sachidanandam, R., Hannon, G. J. and Lehmann, R. (2011) ‘PiRNA production requires heterochromatin formation in drosophila’, *Current Biology*, 21(16), pp. 1373–1379. doi: 10.1016/j.cub.2011.06.057.

Rinn, J. L. and Chang, H. Y. (2012) ‘Genome Regulation by Long Noncoding RNAs’, *Annual Review of Biochemistry*, 81(1), pp. 145–166. doi: 10.1146/annurev-biochem-051410-092902.

Robinson, M. D., McCarthy, D. J. and Smyth, G. K. (2010) ‘edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.’, *Bioinformatics (Oxford, England)*, 26(1), pp. 139–40. doi: 10.1093/bioinformatics/btp616.

Roth, A., Anisimova, M. and Cannarozzi, G. M. (2012) ‘Measuring codon usage bias’, in *Codon Evolution: Mechanisms and Models*. doi: 10.1093/acprof:osobl/9780199601165.003.0013.

Rothberg, J. M., Hinz, W., Rearick, T. M., Schultz, J., Mileski, W., Davey, M., Leamon, J. H., Johnson, K., Milgrew, M. J., Edwards, M., Hoon, J., Simons, J. F., Marran, D., Myers, J. W., Davidson, J. F., Branting, A., Nobile, J. R., Puc, B. P., Light, D., Clark, T. A., Huber, M., Branciforte, J. T., Stoner, I. B., Cawley, S. E., Lyons, M., Fu, Y., Homer, N., Sedova, M., Miao, X., Reed, B., Sabina, J., Feierstein, E., Schorn, M., Alanjary, M., Dimalanta, E., Dressman, D., Kasinskas, R., Sokolsky, T., Fidanza, J. A., Namsaraev, E., McKernan, K. J., Williams, A., Roth, G. T. and Bustillo, J. (2011) ‘An integrated semiconductor device enabling non-optical genome sequencing’, *Nature*, 475(7356), pp. 348–352. doi: 10.1038/nature10242.

Roymondal, U., Das, S. and Sahoo, S. (2009) ‘Predicting gene expression level from relative codon usage bias: An application to escherichia coli genome’, *DNA Research*, 16(1), pp. 13–30. doi: 10.1093/dnares/dsn029.

Rutherford, K., Parkhill, J., Crook, J., Horsnell, T., Rice, P., Rajandream, M. a and Barrell, B. (2000) ‘Artemis: sequence visualization and annotation.’, *Bioinformatics (Oxford, England)*, 16(10), pp. 944–945. doi: 10.1093/bioinformatics/16.10.944.

Sabidussi, G. (1966) ‘The centrality of a graph.’, *Psychometrika*, 31(4), pp. 581–603. doi: 10.1016/j.socnet.2005.11.005.

Saeys, Y., Inza, I. and Larranaga, P. (2007) ‘A review of feature selection techniques in bioinformatics’, *Bioinformatics*, pp. 2507–2517. doi: 10.1093/bioinformatics/btm344.

Saiki, R., Gelfand, D., Stoffel, S., Scharf, S., Higuchi, R., Horn, G., Mullis, K. and Erlich, H. (1988) ‘Primer-directed enzymatic amplification of DNA with a thermostable DNA polymerase’, *Science*, 239(4839), pp. 487–491. doi: 10.1126/science.2448875.

Saiki, R. K., Scharf, S., Faloona, F., Mullis, K. B., Horn, G. T., Erlich, H. A. and Arnheim, N. (1985) ‘Enzymatic amplification of beta-globin genomic sequences and restriction site analysis for diagnosis of sickle cell anemia’, *Science*, 230(4732), pp. 1350–1354. doi: 10.1126/science.2999980.

Sanger, F. and Coulson, A. R. (1975) ‘A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase’, *Journal of Molecular Biology*, 94(3). doi: 10.1016/0022-2836(75)90213-2.

Schanen, B. C. and Li, X. (2011) ‘Transcriptional regulation of mammalian miRNA genes’, *Genomics*, pp. 1–6. doi: 10.1016/j.ygeno.2010.10.005.

Schmid-Burgk, J. L. and Hornung, V. (2015) ‘BrowserGenome.org: web-based RNA-seq data analysis and visualization’, *Nat Meth*, 12(11), p. 1001. doi: 10.1038/nmeth.3615.

Sequencing, H. (2011) ‘CLC Genomics Workbench’, *Workbench*, pp. 1–4.

Shah, R. D. and Meinshausen, N. (2014) ‘Random Intersection Trees’, *J. Mach. Learn. Res.* JMLR.org, 15(1), pp. 629–654.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B. and Ideker, T. (2003) ‘Cytoscape: A software Environment for integrated models of biomolecular interaction networks’, *Genome Research*, 13(11), pp. 2498–2504. doi: 10.1101/gr.1239303.

Sharan, R., Ulitsky, I. and Shamir, R. (2007) ‘Network-based prediction of protein function’, *Molecular Systems Biology*, pp. 1–13. doi: 10.1038/msb4100129.

Sharp, P. M., Tuohy, T. M. F. and Mosurski, K. R. (1986) ‘Codon usage in yeast: Cluster analysis clearly differentiates highly and lowly expressed genes’, *Nucleic Acids Research*, 14(13), pp. 5125–5143. doi: 10.1093/nar/14.13.5125.

Shen, L., Shao, N., Liu, X. and Nestler, E. (2014) ‘ngs.plot: Quick mining and visualization of next-generation sequencing data by integrating genomic databases.’, *BMC genomics*, 15(1),

p. 284. doi: 10.1186/1471-2164-15-284.

Shen, S., Park, J. W., Huang, J., Dittmar, K. A., Lu, Z. X., Zhou, Q., Carstens, R. P. and Xing, Y. (2012) ‘MATS: A Bayesian framework for flexible detection of differential alternative splicing from RNA-Seq data’, *Nucleic Acids Research*, 40(8). doi: 10.1093/nar/gkr1291.

Shifman, A. R., Johnson, R. M. and Wilhelm, B. T. (2016) ‘Cascade: an RNA-seq visualization tool for cancer genomics’, *BMC Genomics*, 17(1), p. 75. doi: 10.1186/s12864-016-2389-8.

Signal, B., Gloss, B. S. and Dinger, M. E. (2016) ‘Computational Approaches for Functional Prediction and Characterisation of Long Noncoding RNAs’, *Trends in Genetics*. doi: 10.1016/j.tig.2016.08.004.

Simonelig, M. (2011) ‘Developmental functions of piRNAs and transposable elements: a Drosophila point-of-view.’, *RNA biology*, 8(5), pp. 754–759. doi: 10.4161/rna.8.5.16042.

Singh, U., Khemka, N., Rajkumar, M. S., Garg, R. and Jain, M. (2017) ‘PLncPRO for prediction of long non-coding RNAs (lncRNAs) in plants and its application for discovery of abiotic stress-responsive lncRNAs in rice and chickpea’, *Nucleic acids research*. doi: 10.1093/nar/gkx866.

Skinner, M. E., Uzilov, A. V., Stein, L. D., Mungall, C. J. and Holmes, I. H. (2009) ‘JBrowse: A next-generation genome browser’, *Genome Research*, 19(9), pp. 1630–1638. doi: 10.1101/gr.094607.109.

St.Laurent, G., Wahlestedt, C. and Kapranov, P. (2015) ‘The Landscape of long noncoding RNA classification’, *Trends in Genetics*, pp. 249–251. doi: 10.1016/j.tig.2015.03.007.

Stein, L. D. (2013) ‘Using GBrowse 2.0 to visualize and share next-generation sequence data’, *Briefings in Bioinformatics*, 14(2), pp. 162–171. doi: 10.1093/bib/bbt001.

Struhl, K. (2007) ‘Transcriptional noise and the fidelity of initiation by RNA polymerase II’, *Nature Structural and Molecular Biology*, pp. 103–105. doi: 10.1038/nsmb0207-103.

Sun, J., Ruan, Y., Wang, M., Chen, R., Yu, N., Sun, L., Liu, T. and Chen, H. (2016) ‘Differentially expressed circulating LncRNAs and mRNA identified by microarray analysis in obese patients’, *Scientific Reports*, 6. doi: 10.1038/srep35421.

Sun, L., Liu, H., Zhang, L. and Meng, J. (2015) ‘lncRScan-SVM: A Tool for Predicting Long Non-Coding RNAs Using Support Vector Machine’, *PLoS One*, 10(10), p. e0139654. doi: 10.1371/journal.pone.0139654.

Sun, L., Luo, H., Bu, D., Zhao, G., Yu, K., Zhang, C., Liu, Y., Chen, R. and Zhao, Y. (2013) ‘Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts’, *Nucleic Acids Research*, 41(17). doi: 10.1093/nar/gkt646.

Suresh, M. X., Gromiha, M. M. and Suwa, M. (2015a) ‘Development of a machine learning method to predict membrane protein-ligand binding residues using basic sequence information’, *Advances in Bioinformatics*, 2015. doi: 10.1155/2015/843030.

Suresh, V., Liu, L., Adjeroh, D. and Zhou, X. (2015b) ‘RPI-Pred: Predicting ncRNA-protein interaction using sequence and structural information’, *Nucleic Acids Research*, 43(3), pp. 1370–1379. doi: 10.1093/nar/gkv020.

Suzuki, H., Saito, R. and Tomita, M. (2004) ‘The “weighted sum of relative entropy”: A new index for synonymous codon usage bias', *Gene*, 335(1–2), pp. 19–23. doi: 10.1016/j.gene.2004.03.001.

Swiezewski, S., Liu, F., Magusin, A. and Dean, C. (2009) 'Cold-induced silencing by long antisense transcripts of an Arabidopsis Polycomb target', *Nature*, 462(7274), pp. 799–802. doi: 10.1038/nature08618.

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., Kuhn, M., Bork, P., Jensen, L. J. and Von Mering, C. (2015) 'STRING v10: Protein-protein interaction networks, integrated over the tree of life', *Nucleic Acids Research*, 43(D1), pp. D447–D452. doi: 10.1093/nar/gku1003.

Tarazona, S., García, F., Ferrer, A., Dopazo, J. and Conesa, A. (2012) 'NOIseq: a RNA-seq differential expression method robust for sequencing depth biases', *EMBnet.journal*, 17(B), p. 18. doi: 10.14806/ej.17.B.265.

Thorvaldsdóttir, H., Robinson, J. T. and Mesirov, J. P. (2013) 'Integrative Genomics Viewer (IGV): High-performance genomics data visualization and exploration', *Briefings in Bioinformatics*, 14(2), pp. 178–192. doi: 10.1093/bib/bbs017.

Tibshirani, R. (1996) 'Regression Selection and Shrinkage via the Lasso', *Journal of the Royal Statistical Society B*, pp. 267–288. doi: 10.2307/2346178.

Trapnell, C., Pachter, L. and Salzberg, S. L. (2009) 'TopHat: Discovering splice junctions with RNA-Seq', *Bioinformatics*, 25(9), pp. 1105–1111. doi: 10.1093/bioinformatics/btp120.

Trapnell, C., Roberts, A., Goff, L., Pertea, G., Kim, D., Kelley, D. R., Pimentel, H., Salzberg, S. L., Rinn, J. L. and Pachter, L. (2012) 'Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks.', *Nature protocols*, 7(3), pp. 562–78. doi: 10.1038/nprot.2012.016.

Tripathi, R., Patel, S., Kumari, V. and Chakraborty, P. (2016) 'DeepLNC , a long non-coding RNA prediction tool using deep neural network', *Network Modeling Analysis in Health Informatics and Bioinformatics*. Springer Vienna. doi: 10.1007/s13721-016-0129-2.

Tsai, M. C., Spitale, R. C. and Chang, H. Y. (2011) 'Long intergenic noncoding RNAs: New links in cancer progression', *Cancer Research*, pp. 3–7. doi: 10.1158/0008-5472.CAN-10-2483.

Ulitsky, I. and Bartel, D. P. (2013) 'LincRNAs: Genomics, evolution, and mechanisms', *Cell*, 154(1), pp. 26–46. doi: 10.1016/j.cell.2013.06.020.

Vanderkam, D., Aksoy, B. A., Hodes, I., Perrone, J. and Hammerbacher, J. (2016) 'Pileup.js: A JavaScript library for interactive and in-browser visualization of genomic data', *Bioinformatics*, 32(15), pp. 2378–2379. doi: 10.1093/bioinformatics/btw167.

Vitting-Seerup, K., Porse, B., Sandelin, A. and Waage, J. (2014) 'spliceR: an R package for classification of alternative splicing and prediction of coding potential from RNA-seq data', *BMC Bioinformatics*, 15(1), p. 81. doi: 10.1186/1471-2105-15-81.

Volders, P. J., Helsens, K., Wang, X., Menten, B., Martens, L., Gevaert, K., Vandesompele, J. and Mestdagh, P. (2013) 'LNCipedia: A database for annotated human lncRNA transcript sequences and structures', *Nucleic Acids Research*, 41(D1). doi: 10.1093/nar/gks915.

Wan, X.-F., Xu, D., Kleinhofs, A. and Zhou, J. (2004) 'Quantitative relationship between synonymous codon usage bias and GC composition across unicellular genomes.', *BMC* *evolutionary biology*, 4, p. 19. doi: 10.1186/1471-2148-4-19.

Wang, B. and Brendel, V. (2006) ‘Genomewide comparative analysis of alternative splicing in plants’, *Pnas*, 103(18), p. 602039103. doi: 10.1073/pnas.0602039103.

Wang, J., Ma, R., Ma, W., Chen, J., Yang, J., Xi, Y. and Cui, Q. (2016a) ‘LncDisease: A sequence based bioinformatics tool for predicting lncRNA-disease associations’, *Nucleic Acids Research*, 44(9). doi: 10.1093/nar/gkw093.

Wang, L., Park, H. J., Dasari, S., Wang, S., Kocher, J. P. and Li, W. (2013) ‘CPAT: Coding-potential assessment tool using an alignment-free logistic regression model’, *Nucleic Acids Research*, 41(6). doi: 10.1093/nar/gkt006.

Wang, P., Fu, H., Cui, J. and Chen, X. (2016b) ‘Differential lncRNA-mRNA co-expression network analysis revealing the potential regulatory roles of lncRNAs in myocardial infarction’, *Molecular Medicine Reports*, 13(2), pp. 1195–1203. doi: 10.3892/mmr.2015.4669.

Wang, T., Lin, Y., Chen, Y., Yeh, C., Huang, Y., Hsieh, T., Shieh, T., Hsueh, C. and Chen, T. (2015) ‘Long non-coding RNA AOC4P suppresses hepatocellular carcinoma metastasis by enhancing vimentin degradation and inhibiting epithelial-mesenchymal transition.’, *Oncotarget*. doi: 10.18632/oncotarget.4344.

Wang, W., Gao, F., Zhao, Z., Wang, H., Zhang, L., Zhang, D., Zhang, Y., Lan, Q., Wang, J. and Zhao, J. (2017) ‘Integrated Analysis of LncRNA-mRNA Co-Expression Profiles in Patients with Moyamoya Disease’, *Scientific Reports*, 7. doi: 10.1038/srep42421.

Wang, Y., Fan, X., Lin, F., He, G., Terzaghi, W., Zhu, D. and Deng, X. W. (2014) ‘Arabidopsis noncoding RNA mediates control of photomorphogenesis by red light’, *Proceedings of the National Academy of Sciences*, 111(28), pp. 10359–10364. doi: 10.1073/pnas.1409457111.

Warren, R. L., Sutton, G. G., Jones, S. J. M. and Holt, R. A. (2007) ‘Assembling millions of short DNA sequences using SSAKE’, *Bioinformatics*, 23(4), pp. 500–501. doi: 10.1093/bioinformatics/btl629.

Wasaki, J., Yonetani, R., Shinano, T., Kai, M. and Osaki, M. (2003) ‘Expression of the OsPI1 gene, cloned from rice roots using cDNA microarray, rapidly responds to phosphorus status’, *New Phytologist*, 158(2), pp. 239–248. doi: 10.1046/j.1469-8137.2003.00748.x.

Washietl, S., Kellis, M. and Garber, M. (2014) ‘Evolutionary dynamics and tissue specificity of human long noncoding RNAs in six mammals’, *Genome Research*. doi: 10.1101/gr.165035.113.

Williams, S. (1996) ‘Pearson’s correlation coefficient.’, *The New Zealand medical journal*, p. 38. doi: 10.1136/bmj.e4483.

Wu, T. D., Reeder, J., Lawrence, M., Becker, G. and Brauer, M. J. (2016) ‘GMAP and GSNAP for genomic sequence alignment: Enhancements to speed, accuracy, and functionality’, in *Methods in Molecular Biology*, pp. 283–334. doi: 10.1007/978-1-4939-3578-9_15.

Wu, T. T. and Lange, K. (2008) ‘Coordinate descent algorithms for lasso penalized regression’, *Annals of Applied Statistics*, 2(1), pp. 224–244. doi: 10.1214/07-AOAS147.

Wu, T., Wang, J., Liu, C., Zhang, Y., Shi, B., Zhu, X., Zhang, Z., Skogerbø, G., Chen, L., Lu, H., Zhao, Y. and Chen, R. (2006) ‘NPInter: the noncoding RNAs and protein related biomacromolecules interaction database.’, *Nucleic acids research*, 34(Database issue), pp.

D150–D152. doi: 10.1093/nar/gkj025.

Wucher, V., Legeai, F., Hédan, B., Rizk, G., Lagoutte, L., Leeb, T., Jagannathan, V., Cadieu, E., David, A., Lohi, H., Cirera, S., Fredholm, M., Botherel, N., Leegwater, P. A. J., Le Béguec, C., Fieten, H., Johnson, J., Alföldi, J., André, C., Lindblad-Toh, K., Hitte, C. and Derrien, T. (2017) 'FEELnc: A tool for long non-coding RNA annotation and its application to the dog transcriptome', *Nucleic Acids Research*, 45(8). doi: 10.1093/nar/gkw1306.

Wunderlich, M., Groß-Hardt, R. and Schöffl, F. (2014) 'Heat shock factor HSFB2a involved in gametophyte development of Arabidopsis thaliana and its expression is controlled by a heat-inducible long non-coding antisense RNA', *Plant Molecular Biology*, 85(6), pp. 541–550. doi: 10.1007/s11103-014-0202-0.

Xiao, Y., Lv, Y., Zhao, H., Gong, Y., Hu, J., Li, F., Xu, J., Bai, J., Yu, F. and Li, X. (2015) 'Predicting the Functions of Long Noncoding RNAs Using RNA-Seq Based on Bayesian Network', *Biomed Res Int*, 2015, p. 839590. doi: 10.1155/2015/839590.

Xie, C., Yuan, J., Li, H., Li, M., Zhao, G., Bu, D., Zhu, W., Wu, W., Chen, R. and Zhao, Y. (2014a) 'NONCODEv4: Exploring the world of long non-coding RNA genes', *Nucleic Acids Research*, 42(D1). doi: 10.1093/nar/gkt1222.

Xie, Y., Wu, G., Tang, J., Luo, R., Patterson, J., Liu, S., Huang, W., He, G., Gu, S., Li, S., Zhou, X., Lam, T. W., Li, Y., Xu, X., Wong, G. K. S. and Wang, J. (2014b) 'SOAPdenovo-Trans: De novo transcriptome assembly with short RNA-Seq reads', *Bioinformatics*, 30(12), pp. 1660–1666. doi: 10.1093/bioinformatics/btu077.

Xu, Q., Song, Z., Zhu, C., Tao, C., Kang, L., Liu, W., He, F., Yan, J. and Sang, T. (2017) 'Systematic comparison of lncRNAs with protein coding mRNAs in population expression and their response to environmental change', *BMC Plant Biology*. London. doi: 10.1186/s12870-017-0984-8.

Yang, Y., Shen, H. T., Ma, Z., Huang, Z. and Zhou, X. (2011) 'ℓ2,1-Norm regularized discriminative feature selection for unsupervised learning', in *IJCAI International Joint Conference on Artificial Intelligence*. doi: 10.5591/978-1-57735-516-8/IJCAI11-267.

Yu, B. (2013) 'Stability', *Bernoulli*, 19(4), pp. 1484–1500. doi: 10.3150/13-BEJSP14.

Zeng, Y., Luo, J. and Lin, S. (2009) 'Classification using Markov blanket for feature selection', *2009 IEEE International Conference on Granular Computing*, pp. 743–747. doi: 10.1109/GRC.2009.5255023.

Zerbino, D. R., Achuthan, P., Akanni, W., Amode, M. R., Barrell, D., Bhai, J., Billis, K., Cummins, C., Gall, A., Girón, C. G., Gil, L., Gordon, L., Haggerty, L., Haskell, E., Hourlier, T., Izuogu, O. G., Janacek, S. H., Juettemann, T., To, J. K., Laird, M. R., Lavidas, I., Liu, Z., Loveland, J. E., Maurel, T., McLaren, W., Moore, B., Mudge, J., Murphy, D. N., Newman, V., Nuhn, M., Ogeh, D., Ong, C. K., Parker, A., Patricio, M., Riat, H. S., Schuilenburg, H., Sheppard, D., Sparrow, H., Taylor, K., Thormann, A., Vullo, A., Walts, B., Zadissa, A., Frankish, A., Hunt, S. E., Kostadima, M., Langridge, N., Martin, F. J., Muffato, M., Perry, E., Ruffier, M., Staines, D. M., Trevanion, S. J., Aken, B. L., Cunningham, F., Yates, A. and Flicek, P. (2017) 'Ensembl 2018', *Nucleic Acids Research*. doi: 10.1093/nar/gkx1098.

Zhang, C., Fan, C., Gan, J., Zhu, P., Kong, L. and Li, C. (2018) 'iSeq: Web-Based RNA-seq Data Analysis and Visualization', in Huang, T. (ed.) *Computational Systems Biology: Methods and Protocols*. New York, NY: Springer New York, pp. 167–181. doi: 10.1007/978-1-4939-7717-8_10.

Zhang, C., Freddolino, P. L. and Zhang, Y. (2017) 'COFACTOR: Improved protein function prediction by combining structure, sequence and protein-protein interaction information', *Nucleic Acids Research*, 45(W1), pp. W291–W299. doi: 10.1093/nar/gkx366.

Zhang, L., Zhou, Y., Huang, T., Cheng, A. S. L., Yu, J., Kang, W. and To, K. F. (2017) 'The interplay of LncRNA-H19 and its binding partners in physiological process and gastric carcinogenesis', *International Journal of Molecular Sciences*. doi: 10.3390/ijms18020450.

Zhang, T. (2008) 'Adaptive Forward-Backward Greedy Algorithm for Sparse Learning with Linear Models.', *NIPS*, (1), pp. 1–8.

Zhang, Z. H., Jhaveri, D. J., Marshall, V. M., Bauer, D. C., Edson, J., Narayanan, R. K., Robinson, G. J., Lundberg, A. E., Bartlett, P. F., Wray, N. R. and Zhao, Q. Y. (2014) 'A comparative study of techniques for differential expression analysis on RNA-seq data', *PLoS ONE*, 9(8), p. e103207. doi: 10.1371/journal.pone.0103207.

Zhao, J., Song, X. and Wang, K. (2016a) 'lncScore: alignment-free identification of long noncoding\nRNA from assembled novel transcripts', *Sci. Rep.*, 6, p. 34838. doi: 10.1038/srep34838.

Zhao, Y., Li, H., Fang, S., Kang, Y., Wu, W., Hao, Y., Li, Z., Bu, D., Sun, N., Zhang, M. Q. and Chen, R. (2016b) 'NONCODE 2016: An informative and valuable data source of long non-coding RNAs', *Nucleic Acids Research*, 44(D1), pp. D203–D208. doi: 10.1093/nar/gkv1252.

Zheng, L. L., Li, J. H., Wu, J., Sun, W. J., Liu, S., Wang, Z. L., Zhou, H., Yang, J. H. and Qu, L. H. (2016) 'deepBase v2.0: Identification, expression, evolution and function of small RNAs, LncRNAs and circular RNAs from deep-sequencing data', *Nucleic Acids Research*. doi: 10.1093/nar/gkv1273.

Zhou, J., Huang, Y., Ding, Y., Yuan, J., Wang, H. and Sun, H. (2018) 'lncFunTK: a toolkit for functional annotation of long noncoding RNAs', *Bioinformatics*, p. bty339. doi: 10.1093/bioinformatics/bty339.

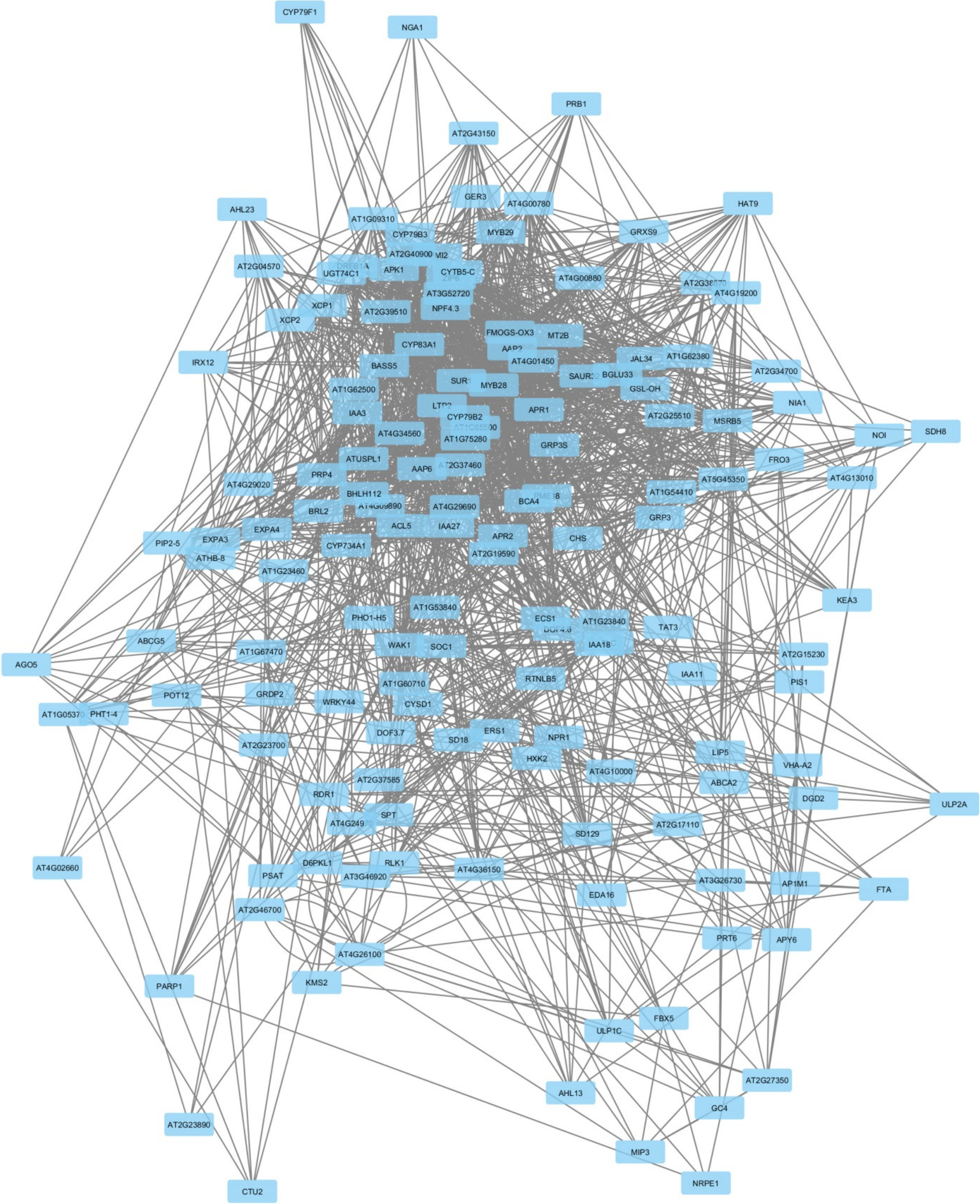
Zhou, T., Ren, J., Medo, M. and Zhang, Y. C. (2007) 'Bipartite network projection and personal recommendation', *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 76(4). doi: 10.1103/PhysRevE.76.046115.

Zhu, M., Zhang, M., Xing, L., Li, W., Jiang, H., Wang, L. and Xu, M. (2017) 'Transcriptomic analysis of long non-coding RNAs and coding genes uncovers a complex regulatory network that is involved in maize seed development', *Genes*, 8(10). doi: 10.3390/genes8100274.

Zou, H. and Hastie, T. (2005) 'Regularization and variable selection via the elastic net', *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 67(2), pp. 301–320. doi: 10.1111/j.1467-9868.2005.00503.x.

APPENDIX A

a



CYP79F1
NGA1
PRB1
AT2G43150
GER3
AT4G00780
HAT9
AHL23
AT1G09310
CYP79B3
MYB29
GRXS9
AT2G40900
MI2
AT2G04570
DREB1A
UGT74C1
APK1
CYTB5-C
ZIP6
AT4G00880
AT2G38870
AT4G19200
AT3G52720
XCP1
AT2G39510
NPF4.3
XCP2
FMOGS-OX3
MT2B
CYP83A1
AAP2
AT4G01450
JAL34
AT1G62380
IRX12
BASS5
AT2G34700
SAUR32
BGLU33
SUR1
MYB28
GSL-OH
AT1G62500
NIA1
LTP2
APR1
IAA3
CYP79B2
AT1G65500
AT2G25510
MSRB5
SDH8
AT4G34560
AT1G75280
NOI
GRP3S
ATUSPL1
FRO3
AAP6
AT2G37460
PRP4
AT5G45350
AT4G13010
AT4G29020
BHLH112
PME18
AT1G54410
BCA4
AT4G09890
AT4G29690
BRL2
GRP3
EXPA4
ACL5
IAA27
PIP2-5
EXPA3
APR2
CHS
CYP734A1
ATHB-8
AT2G19590
AT1G23460
KEA3
AT1G53840
PHO1-H5
ECS1
AT1G23840
DOF4.6
TAT3
ABCG5
IAA18
WAK1
SOC1
AT2G15230
AGO5
AT1G67470
IAA11
PIS1
RTNLB5
AT1G60710
POT12
GRDP2
WRKY44
CYSD1
AT1G05370
PHT1-4
ERS1
DOF3.7
NPR1
SD18
AT2G23700
LIP5
HXK2
VHA-A2
AT4G10000
ABCA2
AT2G37585
RDR1
DGD2
ULP2A
SPT
AT4G24970
AT2G17110
SD129
AT4G02660
D6PKL1
RLK1
AT4G36150
PSAT
AT3G46920
AT3G26730
AP1M1
FTA
EDA16
AT2G46700
PRT6
APY6
AT4G26100
KMS2
PARP1
FBX5
ULP1C
AT2G27350
AHL13
GC4
AT2G23890
MIP3
NRPE1
CTU2

b

Some materials have been removed due to 3rd party copyright. The unabridged version can be viewed in Lancester Library - Coventry University.

Figure A.1: Protein-protein interaction network and Functional Grouped Network (FGN) of CGenes (in *A.thaliana* apical-shoot dataset) obtained from Cuffdiff-DESeq-edgeR overlap (Anders and Huber, 2010; Robinson, McCarthy and Smyth, 2010; Trapnell *et al.*, 2012). (a) PPI network obtained from GeneMania (Montojo *et al.*, 2010) showing interconnection and regulation of genes displayed by nodes which are coloured in blue and edges coloured in grey, (b) FGN obtained from ClueGO (Bindea *et al.*, 2009) with GOTerms as nodes linked based on kappa score where node size represents enrichment significance.

Table A.1: BMRF function annotation results of 50 lncRNA sequences in *A. thaliana* apical-shoot dataset.

Genename	GO Term	probability	Function
AT1G10682.1	GO:0000502	0.999985	is subunit of proteasome complex
AT1G10682.1	GO:0000502	0.999985	located in proteasome complex
AT1G08592.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G08592.1	GO:0000502	0.99799	located in proteasome complex
AT1G26558.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G26558.1	GO:0000502	0.95731	located in proteasome complex
AT1G04425.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G04425.1	GO:0000502	0.99799	located in proteasome complex
AT1G25175.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G25175.1	GO:0000502	0.95731	located in proteasome complex
AT1G22403.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G22403.1	GO:0000502	0.99974	located in proteasome complex
AT1G26208.2	GO:0000502	0.923123	is subunit of proteasome complex
AT1G26208.2	GO:0000502	0.923123	located in proteasome complex
AT1G18735.1	GO:0000502	0.999985	is subunit of proteasome complex
AT1G18735.1	GO:0000502	0.999985	located in proteasome complex
AT1G07119.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G07119.1	GO:0000502	0.95731	located in proteasome complex
AT1G07128.1	GO:0000502	0.999681	is subunit of proteasome complex
AT1G07128.1	GO:0000502	0.999681	located in proteasome complex
AT1G17255.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G17255.1	GO:0000502	0.99799	located in proteasome complex
AT1G01448.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G01448.1	GO:0000502	0.99799	located in proteasome complex
AT1G07728.2	GO:0000502	0.95731	is subunit of proteasome complex
AT1G07728.2	GO:0000502	0.95731	located in proteasome complex
AT1G18415.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G18415.1	GO:0000502	0.99799	located in proteasome complex
AT1G05562.1	GO:0000502	0.947952	is subunit of proteasome complex
AT1G05562.1	GO:0000502	0.947952	located in proteasome complex
AT1G18382.1	GO:0000502	0.999966	is subunit of proteasome complex
AT1G18382.1	GO:0000502	0.999966	located in proteasome complex
AT1G25098.2	GO:0000502	0.923123	is subunit of proteasome complex
AT1G25098.2	GO:0000502	0.923123	located in proteasome complex
AT1G11592.2	GO:0000502	0.965041	is subunit of proteasome complex
AT1G11592.2	GO:0000502	0.965041	located in proteasome complex
AT1G21529.1	GO:0000502	0.999972	is subunit of proteasome complex
AT1G21529.1	GO:0000502	0.999972	located in proteasome complex

AT1G16635.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G16635.1	GO:0000502	0.99974	located in proteasome complex
AT1G03545.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G03545.1	GO:0000502	0.95731	located in proteasome complex
AT1G07728.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G07728.1	GO:0000502	0.95731	located in proteasome complex
AT1G02952.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G02952.1	GO:0000502	0.99974	located in proteasome complex
AT1G06265.1	GO:0000502	0.99799	is subunit of proteasome complex
AT1G06265.1	GO:0000502	0.99799	located in proteasome complex
AT1G11175.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G11175.1	GO:0000502	0.99974	located in proteasome complex
AT1G20515.1	GO:0000502	0.838972	is subunit of proteasome complex
AT1G20515.1	GO:0000502	0.838972	located in proteasome complex
AT1G06265.2	GO:0000502	0.947952	is subunit of proteasome complex
AT1G06265.2	GO:0000502	0.947952	located in proteasome complex
AT1G11185.1	GO:0000502	0.999985	is subunit of proteasome complex
AT1G11185.1	GO:0000502	0.999985	located in proteasome complex
AT1G16489.1	GO:0000502	0.947952	is subunit of proteasome complex
AT1G16489.1	GO:0000502	0.947952	located in proteasome complex
AT1G17232.1	GO:0000502	0.923123	is subunit of proteasome complex
AT1G17232.1	GO:0000502	0.923123	located in proteasome complex
AT1G13448.1	GO:0000502	0.999949	is subunit of proteasome complex
AT1G13448.1	GO:0000502	0.999949	located in proteasome complex
AT1G04295.1	GO:0000502	0.996972	is subunit of proteasome complex
AT1G04295.1	GO:0000502	0.996972	located in proteasome complex
AT1G20691.1	GO:0000502	0.95731	is subunit of proteasome complex
AT1G20691.1	GO:0000502	0.95731	located in proteasome complex
AT1G22403.2	GO:0000502	0.984439	is subunit of proteasome complex
AT1G22403.2	GO:0000502	0.984439	located in proteasome complex
AT1G18745.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G18745.1	GO:0000502	0.99974	located in proteasome complex
AT1G01448.3	GO:0000502	0.887787	is subunit of proteasome complex
AT1G01448.3	GO:0000502	0.887787	located in proteasome complex
AT1G26218.1	GO:0000502	0.947952	is subunit of proteasome complex
AT1G26218.1	GO:0000502	0.947952	located in proteasome complex
AT1G15002.1	GO:0000502	0.999985	is subunit of proteasome complex
AT1G15002.1	GO:0000502	0.999985	located in proteasome complex
AT1G24068.1	GO:0000502	0.999972	is subunit of proteasome complex
AT1G24068.1	GO:0000502	0.999972	located in proteasome complex
AT1G01448.2	GO:0000502	0.947952	is subunit of proteasome complex

AT1G01448.2	GO:0000502	0.947952	located in proteasome complex
AT1G26208.1	GO:0000502	0.965041	is subunit of proteasome complex
AT1G26208.1	GO:0000502	0.965041	located in proteasome complex
AT1G14518.1	GO:0000502	0.999949	is subunit of proteasome complex
AT1G14518.1	GO:0000502	0.999949	located in proteasome complex
AT1G06002.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G06002.1	GO:0000502	0.99974	located in proteasome complex
AT1G25098.1	GO:0000502	0.923123	is subunit of proteasome complex
AT1G25098.1	GO:0000502	0.923123	located in proteasome complex
AT1G23052.1	GO:0000502	0.99974	is subunit of proteasome complex
AT1G23052.1	GO:0000502	0.99974	located in proteasome complex
AT1G15175.1	GO:0000502	0.887787	is subunit of proteasome complex
AT1G15175.1	GO:0000502	0.887787	located in proteasome complex
AT1G09421.1	GO:0000502	0.999949	is subunit of proteasome complex
AT1G09421.1	GO:0000502	0.999949	located in proteasome complex
AT1G11592.1	GO:0000502	0.923123	is subunit of proteasome complex
AT1G11592.1	GO:0000502	0.923123	located in proteasome complex
AT1G15405.1	GO:0000502	0.996972	is subunit of proteasome complex
AT1G15405.1	GO:0000502	0.996972	located in proteasome complex
AT1G19968.1	GO:0000502	0.999972	is subunit of proteasome complex
AT1G19968.1	GO:0000502	0.999972	located in proteasome complex
AT1G18735.1	GO:0003677	0.762754	functions in DNA binding
AT1G18735.1	GO:0003677	0.762754	has DNA binding
AT1G11185.1	GO:0003677	0.762754	functions in DNA binding
AT1G11185.1	GO:0003677	0.762754	has DNA binding
AT1G10682.1	GO:0003677	0.762754	functions in DNA binding
AT1G10682.1	GO:0003677	0.762754	has DNA binding
AT1G15002.1	GO:0003677	0.762754	functions in DNA binding
AT1G15002.1	GO:0003677	0.762754	has DNA binding
AT1G18382.1	GO:0003700	0.98441	has sequence-specific DNA binding transcription factor activity
AT1G13448.1	GO:0003700	0.985994	has sequence-specific DNA binding transcription factor activity
AT1G24068.1	GO:0003700	0.990161	has sequence-specific DNA binding transcription factor activity
AT1G16635.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G18745.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G14518.1	GO:0003700	0.985994	has sequence-specific DNA binding transcription factor activity
AT1G02952.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity

AT1G11185.1	GO:0003700	0.995881	has sequence-specific DNA binding transcription factor activity
AT1G04425.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G07128.1	GO:0003700	0.959959	has sequence-specific DNA binding transcription factor activity
AT1G21529.1	GO:0003700	0.990161	has sequence-specific DNA binding transcription factor activity
AT1G01448.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G23052.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G17255.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G18735.1	GO:0003700	0.995881	has sequence-specific DNA binding transcription factor activity
AT1G06002.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G15002.1	GO:0003700	0.995881	has sequence-specific DNA binding transcription factor activity
AT1G10682.1	GO:0003700	0.995881	has sequence-specific DNA binding transcription factor activity
AT1G22403.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G19968.1	GO:0003700	0.990161	has sequence-specific DNA binding transcription factor activity
AT1G08592.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G06265.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G04295.1	GO:0003700	0.758852	has sequence-specific DNA binding transcription factor activity
AT1G18415.1	GO:0003700	0.827086	has sequence-specific DNA binding transcription factor activity
AT1G09421.1	GO:0003700	0.985994	has sequence-specific DNA binding transcription factor activity
AT1G11175.1	GO:0003700	0.957817	has sequence-specific DNA binding transcription factor activity
AT1G01448.1	GO:0005634	0.86913	located in nucleus
AT1G01448.1	GO:0005634	0.86913	expressed in nucleus
AT1G14518.1	GO:0005634	0.903581	located in nucleus
AT1G14518.1	GO:0005634	0.903581	expressed in nucleus
AT1G01448.3	GO:0005634	0.817821	located in nucleus
AT1G01448.3	GO:0005634	0.817821	expressed in nucleus
AT1G10682.1	GO:0005634	0.914013	located in nucleus
AT1G10682.1	GO:0005634	0.914013	expressed in nucleus
AT1G04425.1	GO:0005634	0.86913	located in nucleus
AT1G04425.1	GO:0005634	0.86913	expressed in nucleus

AT1G16635.1	GO:0005634	0.888665	located in nucleus
AT1G16635.1	GO:0005634	0.888665	expressed in nucleus
AT1G07128.1	GO:0005634	0.885801	located in nucleus
AT1G07128.1	GO:0005634	0.885801	expressed in nucleus
AT1G09421.1	GO:0005634	0.903581	located in nucleus
AT1G09421.1	GO:0005634	0.903581	expressed in nucleus
AT1G22403.2	GO:0005634	0.84814	located in nucleus
AT1G22403.2	GO:0005634	0.84814	expressed in nucleus
AT1G01448.2	GO:0005634	0.827943	located in nucleus
AT1G01448.2	GO:0005634	0.827943	expressed in nucleus
AT1G17232.1	GO:0005634	0.822941	located in nucleus
AT1G17232.1	GO:0005634	0.822941	expressed in nucleus
AT1G11592.1	GO:0005634	0.822941	located in nucleus
AT1G11592.1	GO:0005634	0.822941	expressed in nucleus
AT1G15002.1	GO:0005634	0.914013	located in nucleus
AT1G15002.1	GO:0005634	0.914013	expressed in nucleus
AT1G26208.2	GO:0005634	0.822941	located in nucleus
AT1G26208.2	GO:0005634	0.822941	expressed in nucleus
AT1G17255.1	GO:0005634	0.86913	located in nucleus
AT1G17255.1	GO:0005634	0.86913	expressed in nucleus
AT1G11592.2	GO:0005634	0.835929	located in nucleus
AT1G11592.2	GO:0005634	0.835929	expressed in nucleus
AT1G24068.1	GO:0005634	0.908932	located in nucleus
AT1G24068.1	GO:0005634	0.908932	expressed in nucleus
AT1G06002.1	GO:0005634	0.888665	located in nucleus
AT1G06002.1	GO:0005634	0.888665	expressed in nucleus
AT1G23052.1	GO:0005634	0.888665	located in nucleus
AT1G23052.1	GO:0005634	0.888665	expressed in nucleus
AT1G26218.1	GO:0005634	0.827943	located in nucleus
AT1G26218.1	GO:0005634	0.827943	expressed in nucleus
AT1G18735.1	GO:0005634	0.914013	located in nucleus
AT1G18735.1	GO:0005634	0.914013	expressed in nucleus
AT1G04295.1	GO:0005634	0.863822	located in nucleus
AT1G04295.1	GO:0005634	0.863822	expressed in nucleus
AT1G18745.1	GO:0005634	0.888665	located in nucleus
AT1G18745.1	GO:0005634	0.888665	expressed in nucleus
AT1G07728.2	GO:0005634	0.833534	located in nucleus
AT1G07728.2	GO:0005634	0.833534	expressed in nucleus
AT1G13448.1	GO:0005634	0.903581	located in nucleus
AT1G13448.1	GO:0005634	0.903581	expressed in nucleus
AT1G02952.1	GO:0005634	0.888665	located in nucleus

AT1G02952.1	GO:0005634	0.888665	expressed in nucleus
AT1G08592.1	GO:0005634	0.86913	located in nucleus
AT1G08592.1	GO:0005634	0.86913	expressed in nucleus
AT1G11175.1	GO:0005634	0.888665	located in nucleus
AT1G11175.1	GO:0005634	0.888665	expressed in nucleus
AT1G07119.1	GO:0005634	0.833534	located in nucleus
AT1G07119.1	GO:0005634	0.833534	expressed in nucleus
AT1G20691.1	GO:0005634	0.833534	located in nucleus
AT1G20691.1	GO:0005634	0.833534	expressed in nucleus
AT1G06265.1	GO:0005634	0.86913	located in nucleus
AT1G06265.1	GO:0005634	0.86913	expressed in nucleus
AT1G03545.1	GO:0005634	0.833534	located in nucleus
AT1G03545.1	GO:0005634	0.833534	expressed in nucleus
AT1G11185.1	GO:0005634	0.914013	located in nucleus
AT1G11185.1	GO:0005634	0.914013	expressed in nucleus
AT1G07728.1	GO:0005634	0.833534	located in nucleus
AT1G07728.1	GO:0005634	0.833534	expressed in nucleus
AT1G25175.1	GO:0005634	0.833534	located in nucleus
AT1G25175.1	GO:0005634	0.833534	expressed in nucleus
AT1G06265.2	GO:0005634	0.827943	located in nucleus
AT1G06265.2	GO:0005634	0.827943	expressed in nucleus
AT1G22403.1	GO:0005634	0.888665	located in nucleus
AT1G22403.1	GO:0005634	0.888665	expressed in nucleus
AT1G18415.1	GO:0005634	0.86913	located in nucleus
AT1G18415.1	GO:0005634	0.86913	expressed in nucleus
AT1G05562.1	GO:0005634	0.827943	located in nucleus
AT1G05562.1	GO:0005634	0.827943	expressed in nucleus
AT1G25098.1	GO:0005634	0.822941	located in nucleus
AT1G25098.1	GO:0005634	0.822941	expressed in nucleus
AT1G25098.2	GO:0005634	0.822941	located in nucleus
AT1G25098.2	GO:0005634	0.822941	expressed in nucleus
AT1G21529.1	GO:0005634	0.909878	located in nucleus
AT1G21529.1	GO:0005634	0.909878	expressed in nucleus
AT1G18382.1	GO:0005634	0.907501	located in nucleus
AT1G18382.1	GO:0005634	0.907501	expressed in nucleus
AT1G26558.1	GO:0005634	0.833534	located in nucleus
AT1G26558.1	GO:0005634	0.833534	expressed in nucleus
AT1G15405.1	GO:0005634	0.865146	located in nucleus
AT1G15405.1	GO:0005634	0.865146	expressed in nucleus
AT1G20515.1	GO:0005634	0.810878	located in nucleus
AT1G20515.1	GO:0005634	0.810878	expressed in nucleus

AT1G19968.1	GO:0005634	0.909878	located in nucleus
AT1G19968.1	GO:0005634	0.909878	expressed in nucleus
AT1G16489.1	GO:0005634	0.827943	located in nucleus
AT1G16489.1	GO:0005634	0.827943	expressed in nucleus
AT1G15175.1	GO:0005634	0.817821	located in nucleus
AT1G15175.1	GO:0005634	0.817821	expressed in nucleus
AT1G26208.1	GO:0005634	0.835929	located in nucleus
AT1G26208.1	GO:0005634	0.835929	expressed in nucleus
AT1G02952.1	GO:0006260	0.714657	involved in DNA replication
AT1G11175.1	GO:0006260	0.714657	involved in DNA replication
AT1G16635.1	GO:0006260	0.714657	involved in DNA replication
AT1G23052.1	GO:0006260	0.714657	involved in DNA replication
AT1G22403.1	GO:0006260	0.714657	involved in DNA replication
AT1G07128.1	GO:0006260	0.729471	involved in DNA replication
AT1G18745.1	GO:0006260	0.714657	involved in DNA replication
AT1G06002.1	GO:0006260	0.714657	involved in DNA replication
AT1G23052.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G16635.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G07128.1	GO:0006270	0.733036	involved in DNA replication initiation
AT1G22403.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G06002.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G02952.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G18745.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G11175.1	GO:0006270	0.719565	involved in DNA replication initiation
AT1G11175.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G14518.1	GO:0006275	0.858233	involved in regulation of DNA replication
AT1G11185.1	GO:0006275	0.835148	involved in regulation of DNA replication
AT1G23052.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G13448.1	GO:0006275	0.858233	involved in regulation of DNA replication
AT1G15002.1	GO:0006275	0.835148	involved in regulation of DNA replication
AT1G24068.1	GO:0006275	0.847047	involved in regulation of DNA replication
AT1G10682.1	GO:0006275	0.835148	involved in regulation of DNA replication
AT1G06002.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G07128.1	GO:0006275	0.799039	involved in regulation of DNA replication
AT1G18745.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G16635.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G18382.1	GO:0006275	0.850854	involved in regulation of DNA replication
AT1G18735.1	GO:0006275	0.835148	involved in regulation of DNA replication
AT1G22403.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G21529.1	GO:0006275	0.847047	involved in regulation of DNA replication
AT1G09421.1	GO:0006275	0.858233	involved in regulation of DNA replication

AT1G19968.1	GO:0006275	0.847047	involved in regulation of DNA replication
AT1G02952.1	GO:0006275	0.794229	involved in regulation of DNA replication
AT1G11185.1	GO:0006464	0.996563	involved in cellular protein modification process
AT1G06265.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G06002.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G18745.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G14518.1	GO:0006464	0.992556	involved in cellular protein modification process
AT1G15002.1	GO:0006464	0.996563	involved in cellular protein modification process
AT1G01448.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G10682.1	GO:0006464	0.996563	involved in cellular protein modification process
AT1G04425.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G16635.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G18415.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G07128.1	GO:0006464	0.973144	involved in cellular protein modification process
AT1G02952.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G19968.1	GO:0006464	0.994984	involved in cellular protein modification process
AT1G18735.1	GO:0006464	0.996563	involved in cellular protein modification process
AT1G09421.1	GO:0006464	0.992556	involved in cellular protein modification process
AT1G23052.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G13448.1	GO:0006464	0.992556	involved in cellular protein modification process
AT1G22403.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G17255.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G15405.1	GO:0006464	0.856902	involved in cellular protein modification process
AT1G24068.1	GO:0006464	0.994984	involved in cellular protein modification process
AT1G21529.1	GO:0006464	0.994984	involved in cellular protein modification process
AT1G11175.1	GO:0006464	0.976865	involved in cellular protein modification process
AT1G04295.1	GO:0006464	0.856902	involved in cellular protein modification process
AT1G18382.1	GO:0006464	0.994289	involved in cellular protein modification process
AT1G08592.1	GO:0006464	0.893899	involved in cellular protein modification process
AT1G10682.1	GO:0007094	0.851635	involved in mitotic spindle assembly checkpoint
AT1G24068.1	GO:0007094	0.808349	involved in mitotic spindle assembly checkpoint
AT1G14518.1	GO:0007094	0.756051	involved in mitotic spindle assembly checkpoint
AT1G09421.1	GO:0007094	0.756051	involved in mitotic spindle assembly checkpoint
AT1G15002.1	GO:0007094	0.851635	involved in mitotic spindle assembly checkpoint
AT1G18735.1	GO:0007094	0.851635	involved in mitotic spindle assembly checkpoint
AT1G19968.1	GO:0007094	0.808349	involved in mitotic spindle assembly checkpoint
AT1G18382.1	GO:0007094	0.791929	involved in mitotic spindle assembly checkpoint
AT1G11185.1	GO:0007094	0.851635	involved in mitotic spindle assembly checkpoint
AT1G13448.1	GO:0007094	0.756051	involved in mitotic spindle assembly checkpoint
AT1G21529.1	GO:0007094	0.808349	involved in mitotic spindle assembly checkpoint
AT1G18735.1	GO:0008233	0.829943	has peptidase activity

AT1G09421.1	GO:0008233	0.77917	has peptidase activity
AT1G21529.1	GO:0008233	0.806724	has peptidase activity
AT1G19968.1	GO:0008233	0.806724	has peptidase activity
AT1G13448.1	GO:0008233	0.77917	has peptidase activity
AT1G24068.1	GO:0008233	0.806724	has peptidase activity
AT1G15002.1	GO:0008233	0.829943	has peptidase activity
AT1G10682.1	GO:0008233	0.829943	has peptidase activity
AT1G11185.1	GO:0008233	0.829943	has peptidase activity
AT1G18382.1	GO:0008233	0.798055	has peptidase activity
AT1G14518.1	GO:0008233	0.77917	has peptidase activity
AT1G14518.1	GO:0008283	0.725926	involved in cell proliferation
AT1G14518.1	GO:0008283	0.725926	expressed only during cell proliferation
AT1G22403.1	GO:0008283	0.810703	involved in cell proliferation
AT1G22403.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G11175.1	GO:0008283	0.810703	involved in cell proliferation
AT1G11175.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G18745.1	GO:0008283	0.810703	involved in cell proliferation
AT1G18745.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G07128.1	GO:0008283	0.819741	involved in cell proliferation
AT1G07128.1	GO:0008283	0.819741	expressed only during cell proliferation
AT1G18382.1	GO:0008283	0.701366	involved in cell proliferation
AT1G18382.1	GO:0008283	0.701366	expressed only during cell proliferation
AT1G16635.1	GO:0008283	0.810703	involved in cell proliferation
AT1G16635.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G02952.1	GO:0008283	0.810703	involved in cell proliferation
AT1G02952.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G13448.1	GO:0008283	0.725926	involved in cell proliferation
AT1G13448.1	GO:0008283	0.725926	expressed only during cell proliferation
AT1G23052.1	GO:0008283	0.810703	involved in cell proliferation
AT1G23052.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G06002.1	GO:0008283	0.810703	involved in cell proliferation
AT1G06002.1	GO:0008283	0.810703	expressed only during cell proliferation
AT1G09421.1	GO:0008283	0.725926	involved in cell proliferation
AT1G09421.1	GO:0008283	0.725926	expressed only during cell proliferation
AT1G02952.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G23052.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G06002.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G01448.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex

AT1G11185.1	GO:0008540	0.994131	located in proteasome regulatory particle, base subcomplex
AT1G07128.1	GO:0008540	0.953708	located in proteasome regulatory particle, base subcomplex
AT1G19968.1	GO:0008540	0.991086	located in proteasome regulatory particle, base subcomplex
AT1G04295.1	GO:0008540	0.814531	located in proteasome regulatory particle, base subcomplex
AT1G06265.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex
AT1G08592.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex
AT1G18415.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex
AT1G14518.1	GO:0008540	0.98648	located in proteasome regulatory particle, base subcomplex
AT1G18382.1	GO:0008540	0.989756	located in proteasome regulatory particle, base subcomplex
AT1G15405.1	GO:0008540	0.814531	located in proteasome regulatory particle, base subcomplex
AT1G09421.1	GO:0008540	0.98648	located in proteasome regulatory particle, base subcomplex
AT1G04425.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex
AT1G17255.1	GO:0008540	0.853289	located in proteasome regulatory particle, base subcomplex
AT1G15002.1	GO:0008540	0.994131	located in proteasome regulatory particle, base subcomplex
AT1G21529.1	GO:0008540	0.991086	located in proteasome regulatory particle, base subcomplex
AT1G16635.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G10682.1	GO:0008540	0.994131	located in proteasome regulatory particle, base subcomplex
AT1G13448.1	GO:0008540	0.98648	located in proteasome regulatory particle, base subcomplex
AT1G18735.1	GO:0008540	0.994131	located in proteasome regulatory particle, base subcomplex
AT1G24068.1	GO:0008540	0.991086	located in proteasome regulatory particle, base subcomplex
AT1G22403.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G11175.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G18745.1	GO:0008540	0.959532	located in proteasome regulatory particle, base subcomplex
AT1G02952.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G15405.1	GO:0009570	0.851548	located in chloroplast stroma
AT1G24068.1	GO:0009570	0.759327	located in chloroplast stroma

AT1G22403.2	GO:0009570	0.901079	located in chloroplast stroma
AT1G23052.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G11175.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G06002.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G14518.1	GO:0009570	0.767346	located in chloroplast stroma
AT1G04425.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G07128.1	GO:0009570	0.797227	located in chloroplast stroma
AT1G21529.1	GO:0009570	0.884511	located in chloroplast stroma
AT1G04295.1	GO:0009570	0.851548	located in chloroplast stroma
AT1G01448.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G13448.1	GO:0009570	0.767346	located in chloroplast stroma
AT1G18735.1	GO:0009570	0.751868	located in chloroplast stroma
AT1G16635.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G08592.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G11185.1	GO:0009570	0.751868	located in chloroplast stroma
AT1G18382.1	GO:0009570	0.761926	located in chloroplast stroma
AT1G17255.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G06265.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G15002.1	GO:0009570	0.751868	located in chloroplast stroma
AT1G15175.1	GO:0009570	0.709498	located in chloroplast stroma
AT1G19968.1	GO:0009570	0.884511	located in chloroplast stroma
AT1G09421.1	GO:0009570	0.767346	located in chloroplast stroma
AT1G20515.1	GO:0009570	0.729036	located in chloroplast stroma
AT1G10682.1	GO:0009570	0.751868	located in chloroplast stroma
AT1G22403.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G01448.3	GO:0009570	0.709498	located in chloroplast stroma
AT1G18745.1	GO:0009570	0.793344	located in chloroplast stroma
AT1G18415.1	GO:0009570	0.840113	located in chloroplast stroma
AT1G19968.1	GO:0009965	0.737943	involved in leaf morphogenesis
AT1G21529.1	GO:0009965	0.737943	involved in leaf morphogenesis
AT1G11185.1	GO:0010048	0.798751	involved in vernalization response
AT1G11185.1	GO:0010048	0.798751	required for vernalization response
AT1G24068.1	GO:0010048	0.823808	involved in vernalization response
AT1G24068.1	GO:0010048	0.823808	required for vernalization response
AT1G10682.1	GO:0010048	0.798751	involved in vernalization response
AT1G10682.1	GO:0010048	0.798751	required for vernalization response
AT1G15002.1	GO:0010048	0.798751	involved in vernalization response
AT1G15002.1	GO:0010048	0.798751	required for vernalization response
AT1G18735.1	GO:0010048	0.798751	involved in vernalization response
AT1G18735.1	GO:0010048	0.798751	required for vernalization response
AT1G19968.1	GO:0016571	0.873479	involved in histone methylation

AT1G21529.1	GO:0016571	0.873479	involved in histone methylation
AT1G10682.1	GO:0016571	0.865645	involved in histone methylation
AT1G11185.1	GO:0016571	0.865645	involved in histone methylation
AT1G15002.1	GO:0016571	0.865645	involved in histone methylation
AT1G18735.1	GO:0016571	0.865645	involved in histone methylation
AT1G18382.1	GO:0016571	0.876039	involved in histone methylation
AT1G24068.1	GO:0030154	1	involved in cell differentiation
AT1G07728.2	GO:0030154	1	involved in cell differentiation
AT1G19968.1	GO:0030154	1	involved in cell differentiation
AT1G25098.2	GO:0030154	1	involved in cell differentiation
AT1G20691.1	GO:0030154	1	involved in cell differentiation
AT1G16635.1	GO:0030154	1	involved in cell differentiation
AT1G13448.1	GO:0030154	1	involved in cell differentiation
AT1G07728.1	GO:0030154	1	involved in cell differentiation
AT1G23052.1	GO:0030154	1	involved in cell differentiation
AT1G11175.1	GO:0030154	1	involved in cell differentiation
AT1G06265.1	GO:0030154	1	involved in cell differentiation
AT1G03545.1	GO:0030154	1	involved in cell differentiation
AT1G06002.1	GO:0030154	1	involved in cell differentiation
AT1G22403.2	GO:0030154	1	involved in cell differentiation
AT1G25175.1	GO:0030154	1	involved in cell differentiation
AT1G18745.1	GO:0030154	1	involved in cell differentiation
AT1G11592.2	GO:0030154	1	involved in cell differentiation
AT1G07119.1	GO:0030154	1	involved in cell differentiation
AT1G10682.1	GO:0030154	1	involved in cell differentiation
AT1G01448.2	GO:0030154	1	involved in cell differentiation
AT1G11185.1	GO:0030154	1	involved in cell differentiation
AT1G15405.1	GO:0030154	1	involved in cell differentiation
AT1G11592.1	GO:0030154	1	involved in cell differentiation
AT1G15002.1	GO:0030154	1	involved in cell differentiation
AT1G17255.1	GO:0030154	1	involved in cell differentiation
AT1G08592.1	GO:0030154	1	involved in cell differentiation
AT1G06265.2	GO:0030154	1	involved in cell differentiation
AT1G26218.1	GO:0030154	1	involved in cell differentiation
AT1G26558.1	GO:0030154	1	involved in cell differentiation
AT1G04425.1	GO:0030154	1	involved in cell differentiation
AT1G18415.1	GO:0030154	1	involved in cell differentiation
AT1G25098.1	GO:0030154	1	involved in cell differentiation
AT1G15175.1	GO:0030154	1	involved in cell differentiation
AT1G18382.1	GO:0030154	1	involved in cell differentiation
AT1G01448.1	GO:0030154	1	involved in cell differentiation

AT1G02952.1	GO:0030154	1	involved in cell differentiation
AT1G09421.1	GO:0030154	1	involved in cell differentiation
AT1G26208.2	GO:0030154	1	involved in cell differentiation
AT1G07128.1	GO:0030154	1	involved in cell differentiation
AT1G16489.1	GO:0030154	1	involved in cell differentiation
AT1G04295.1	GO:0030154	1	involved in cell differentiation
AT1G21529.1	GO:0030154	1	involved in cell differentiation
AT1G26208.1	GO:0030154	1	involved in cell differentiation
AT1G17232.1	GO:0030154	1	involved in cell differentiation
AT1G22403.1	GO:0030154	1	involved in cell differentiation
AT1G05562.1	GO:0030154	1	involved in cell differentiation
AT1G18735.1	GO:0030154	1	involved in cell differentiation
AT1G14518.1	GO:0030154	1	involved in cell differentiation
AT1G01448.3	GO:0030154	1	involved in cell differentiation
AT1G01448.2	GO:0031507	0.77795	involved in heterochromatin assembly
AT1G15175.1	GO:0031507	0.777874	involved in heterochromatin assembly
AT1G07128.1	GO:0031507	0.860208	involved in heterochromatin assembly
AT1G25175.1	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G11175.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G22403.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G23052.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G02952.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G07728.2	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G11185.1	GO:0031507	0.848912	involved in heterochromatin assembly
AT1G19968.1	GO:0031507	0.851069	involved in heterochromatin assembly
AT1G06265.2	GO:0031507	0.77795	involved in heterochromatin assembly
AT1G22403.2	GO:0031507	0.777902	involved in heterochromatin assembly
AT1G25098.1	GO:0031507	0.777938	involved in heterochromatin assembly
AT1G07728.1	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G04425.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G01448.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G26218.1	GO:0031507	0.77795	involved in heterochromatin assembly
AT1G18745.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G15405.1	GO:0031507	0.873934	involved in heterochromatin assembly
AT1G18735.1	GO:0031507	0.848912	involved in heterochromatin assembly
AT1G16489.1	GO:0031507	0.77795	involved in heterochromatin assembly
AT1G06002.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G14518.1	GO:0031507	0.85326	involved in heterochromatin assembly
AT1G17232.1	GO:0031507	0.777938	involved in heterochromatin assembly
AT1G18415.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G03545.1	GO:0031507	0.777947	involved in heterochromatin assembly

AT1G24068.1	GO:0031507	0.851069	involved in heterochromatin assembly
AT1G08592.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G09421.1	GO:0031507	0.85326	involved in heterochromatin assembly
AT1G06265.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G15002.1	GO:0031507	0.848912	involved in heterochromatin assembly
AT1G17255.1	GO:0031507	0.870102	involved in heterochromatin assembly
AT1G16635.1	GO:0031507	0.859386	involved in heterochromatin assembly
AT1G20691.1	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G20515.1	GO:0031507	0.777679	involved in heterochromatin assembly
AT1G13448.1	GO:0031507	0.85326	involved in heterochromatin assembly
AT1G26208.1	GO:0031507	0.77794	involved in heterochromatin assembly
AT1G10682.1	GO:0031507	0.848912	involved in heterochromatin assembly
AT1G18382.1	GO:0031507	0.851795	involved in heterochromatin assembly
AT1G26208.2	GO:0031507	0.777938	involved in heterochromatin assembly
AT1G04295.1	GO:0031507	0.873934	involved in heterochromatin assembly
AT1G25098.2	GO:0031507	0.777938	involved in heterochromatin assembly
AT1G26558.1	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G01448.3	GO:0031507	0.777874	involved in heterochromatin assembly
AT1G07119.1	GO:0031507	0.777947	involved in heterochromatin assembly
AT1G05562.1	GO:0031507	0.77795	involved in heterochromatin assembly
AT1G21529.1	GO:0031507	0.851069	involved in heterochromatin assembly
AT1G11592.2	GO:0031507	0.77794	involved in heterochromatin assembly
AT1G11592.1	GO:0031507	0.777938	involved in heterochromatin assembly
AT1G09421.1	GO:0051567	0.870138	involved in histone H3-K9 methylation
AT1G13448.1	GO:0051567	0.870138	involved in histone H3-K9 methylation
AT1G11185.1	GO:0051567	0.797527	involved in histone H3-K9 methylation
AT1G18382.1	GO:0051567	0.848776	involved in histone H3-K9 methylation
AT1G10682.1	GO:0051567	0.797527	involved in histone H3-K9 methylation
AT1G19968.1	GO:0051567	0.837054	involved in histone H3-K9 methylation
AT1G21529.1	GO:0051567	0.837054	involved in histone H3-K9 methylation
AT1G18735.1	GO:0051567	0.797527	involved in histone H3-K9 methylation
AT1G24068.1	GO:0051567	0.837054	involved in histone H3-K9 methylation
AT1G15002.1	GO:0051567	0.797527	involved in histone H3-K9 methylation
AT1G14518.1	GO:0051567	0.870138	involved in histone H3-K9 methylation
AT1G21529.1	GO:0051726	0.742142	involved in regulation of cell cycle
AT1G02952.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G22403.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G11185.1	GO:0051726	0.815916	involved in regulation of cell cycle
AT1G23052.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G15002.1	GO:0051726	0.815916	involved in regulation of cell cycle
AT1G16635.1	GO:0051726	0.773148	involved in regulation of cell cycle

AT1G13448.1	GO:0051726	0.75084	involved in regulation of cell cycle
AT1G06002.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G10682.1	GO:0051726	0.815916	involved in regulation of cell cycle
AT1G24068.1	GO:0051726	0.822598	involved in regulation of cell cycle
AT1G18745.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G11175.1	GO:0051726	0.773148	involved in regulation of cell cycle
AT1G09421.1	GO:0051726	0.75084	involved in regulation of cell cycle
AT1G18382.1	GO:0051726	0.824782	involved in regulation of cell cycle
AT1G19968.1	GO:0051726	0.742142	involved in regulation of cell cycle
AT1G07128.1	GO:0051726	0.77584	involved in regulation of cell cycle
AT1G14518.1	GO:0051726	0.75084	involved in regulation of cell cycle
AT1G18735.1	GO:0051726	0.815916	involved in regulation of cell cycle