Prediction of Mammalian Essential Genes based on Sequence and Functional Features

A thesis submitted to The University of Manchester for the degree of Doctor of Philosophy (PhD) in the Faculty of Biology, Medicine and Health

2016

MITRA KABIR

School of Biological Sciences

Contents

List of Figures

List of Tables

Abbreviations

AI	**A**rtifical **I**ntelligence
ANN	**A**rtifical **N**eural **N**etwork
ASP	**A**verage **S**hortest **P**ath
BC	**B**etweenness **C**entrality
BLAST	**B**asic **L**ocal **A**lignment **S**earch **T**ool
BN	**B**ottle**N**eck
CC	**C**loseness **C**entrality
CCo	**C**lustering **Co**efficient
DES	**D**omain **E**nrichment **S**core
DCA	**D**uplicate **C**ommon **A**ncestor
DMNC	**D**ensity of **M**aximum **N**eighbourhood **C**omponent
EPC	**E**dge **P**ercolation **C**omponent
FN	**F**alse **N**egative
FP	**F**alse **P**ositive
FPR	**F**alse **P**ositive **R**ate
GO	**G**ene **O**ntology
***k*–NN**	***k*-N**earest **N**eighbour
K–S	**K**olmogorov **S**mirnov
LASSO	**L**east **A**bsolute **S**hrinkage and **S**election **O**perator
MGI	**M**ouse **G**enome **I**nformatics
MP	**M**ammalian **P**henotype
MRD	**M**ost **R**ecent **D**uplication

MNC	**M**aximum **N**eighbourhood **C**omponent
MYA	**M**illions of **Y**ears **A**ge
ORF	**O**pen **R**eading **F**rame
RF	**R**andom **F**orest
SSD	**S**mall **S**cale **D**uplicate
SVM	**S**upport **V**ector **M**achine
TAI	**T**ranscription **A**ge **I**ndex
TN	**T**rue **N**egative
TP	**T**rue **P**ositive
TPM	**T**ranscripts **P**er **M**illion
TPR	**T**rue **P**ositive **R**ate
WGD	**W**hole **G**enome **D**uplicate

Abstract

Essential genes are those whose presence is imperative for an organism's survival, whereas the functions of non–essential genes may be useful but not critical. Abnormal functionality of essential genes may lead to defects or death at an early stage of life. Knowledge of essential genes is therefore key to understanding development, maintenance of major cellular processes and tissue–specific functions that are crucial for life. Existing experimental techniques for identifying essential genes are accurate, but most of them are time consuming and expensive. Predicting essential genes using computational methods, therefore, would be of great value as they circumvent experimental constraints. Our research is based on the hypothesis that mammalian essential (lethal) and non–essential (viable) genes are distinguishable by various properties. We examined a wide range of features of *Mus musculus* genes, including sequence, protein–protein interactions, gene expression and function, and found 75 features that were statistically discriminative between lethal and viable genes. These features were used as inputs to create a novel machine learning classifier, allowing the prediction of a mouse gene as lethal or viable with the cross–validation and blind test accuracies of ∼91% and ∼93%, respectively. The prediction results are promising, indicating that our classifier is an effective mammalian essential gene prediction method.

We further developed the mouse gene essentiality study by analysing the association between essentiality and gene duplication. Mouse genes were labelled as singletons or duplicates, and their expression patterns over 13 developmental stages were examined. We found that lethal genes originating from duplicates are considerably lower in proportion than singletons. At all developmental stages a significantly higher proportion of singletons and lethal genes are expressed than duplicates and viable genes. Lethal genes were also found to be more ancient than viable genes. In addition, we observed that duplicate pairs with similar patterns of developmental co–expression are more likely to be viable; lethal gene duplicate pairs do not have such a trend. Overall, these results suggest that duplicate genes in mouse are less likely to be essential than singletons.

Finally, we investigated the evolutionary age of mouse genes across development to see if the morphological hourglass pattern exists in the mouse. We found that in mouse embryos, genes expressed in early and late stages are evolutionarily younger than those expressed in mid–embryogenesis, thus yielding an hourglass pattern. However, the oldest genes are not expressed at the phylotypic stage stated in prior studies, but instead at an earlier time point – the egg cylinder stage. These results question the application of the hourglass model to mouse development.

Declaration of Authorship

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

Acknowledgements

All praise and thanks are due to Allah SWT. I would like to thank my PhD supervisors Dr Kathryn Hentges and Professor Andrew Doig for their constant supervision and guidance. Their invaluable advice, suggestions and support have helped me to sharpen up my ideas and to complete this research. I am indebted to their valuable time that they have dedicated to proofread my thesis. I would also like to express my gratitude to my PhD advisor Dr Sam Griffiths–Jones for his insightful questions and general advice.

I would like to express my profound regard to my parents for their encouragement, support and blessing. They were always there in the hard days of my research. My deepest gratitude goes to my sisters and other family members for their love and support.

A special thanks goes to my dearest husband A S Md Mukarram Hossain who has carried the burden of commuting between Cambridge and Manchester even with his extreme PhD workload. With his continuous support and encouragement, I was able to accomplish my goal.

I would also like to thank Professor David Robertson, Dr Simon Bull and Ana Barradas for their support.

I would like to thank the PDS awarding body and the Charles Wallace Trust for selecting me as a recipient of those awards.

Last, but not the least, I would like to show my sincere gratitude to the Commonwealth Scholarship Commission, UK. I could not have completed this research without the funding provided to me by them.

Chapter 1

Introduction

1.1 Embryonic Development and Birth Defects

Development is a key process in the initiation of life for multi–cellular organisms. Development attains two fundamental objectives: it produces cellular diversity and arranges order within each generation, and it guarantees the continuity of life from one generation to the next (Gilbert, 1994). Hence, developmental biology addresses a fundamental question of life: how does a complex multicellular organism develop from a single fertilised egg? It addresses the processes by which a single cell gives rise to the different types of cells that the body contains and ultimately generates the specialised tissues, organs and anatomy of a mature organism. Moreover, developmental biology highlights different changes in development and how these changes create new body forms. Overall, it addresses the events of biological development, *i.e.*, the progressive transformations in size, shape, and functions

during the life of an organism by which its genotypes are decoded into corresponding functioning phenotypes. Developmental biology research mostly concentrates on the prenatal development in mammals, the period during which an embryo or fetus progressively develops after fertilisation and gestates until birth. As most of the changes within major tissues and organs of mammals take place during the prenatal period, understanding the underlying molecular and cellular processes is crucial to comprehend the growth of a new life.

Embryogenesis is the most crucial phase of the prenatal development of mammals. During this stage, an embryo is formed and progressively developed until it becomes a fetus. This progression always commences with the fertilisation of an ovum (or egg) by a sperm in one of the two fallopian tubes several hours after ovulation (Oppenheimer and Lefevre, 1984). The product of the fertilisation process is a zona pellucida bounded single-celled fertilised egg which is termed as the **zygote**. A zygote is the first cell of a new individual that ultimately leads to the growth of an embryo. The development of a zygote into an embryo proceeds through four specific stages: cleavage, implantation, gastrulation and organogenesis. The subsequent development involves the growth of the embryo, thereby, matures it for the life outside the womb. In the following, the brief description of each of these stages is given. Though these developmental stages are very similar among mammals, the duration of the gestation varies. For example, the gestation period for humans is approximately 40 weeks, whereas it lasts for 19 to 21 days for the mouse.

Cleavage

The zygote undergoes a series of rapid mitotic cell divisions called cleavage while moving towards the uterus through the fallopian tube few hours after fertilisation. This generates a cluster of cells (blastomeres) where the combined cell size is identical to the size of the zygote. Cleavage at first divides the zygote into 2 identical cells which eventually progresses through 4–cell, 8–cell, and 16–cell stages and so on (Gilbert, 1994). Cleavage is very important for two specific reasons: it generates a large number of cells which ultimately gives rise to different tissues and organs; it increases the nucleus–cytoplasmic ratio to support embryogenesis. Repeated cleavage allows blastomeres to tightly bind themselves to each other, thereby forming the **morula**. This early embryo looks like a mulberry and consists of 16 to 32 cells. Further cell division occurs once the morula enters into the uterus.

Blastocyst Formation and Implantation

The morula starts to accumulate uterine fluid shortly after it enters the uterine cavity. This results in the formation of a hollow, fluid–filled cavity called blastocoele in the centre of the morula. This embryo structure with blastocoele is called the **blastocyst** (Figure 1.1). A blastocyst is characterised by two cell types: inner cell mass and trophoblast. Inner cell mass gives rise to the embryo proper as well as to the extra-embryonic tissues including the yolk sac, allantois and amnion that support the developing embryo, whereas the trophoblast forms the outer layer of

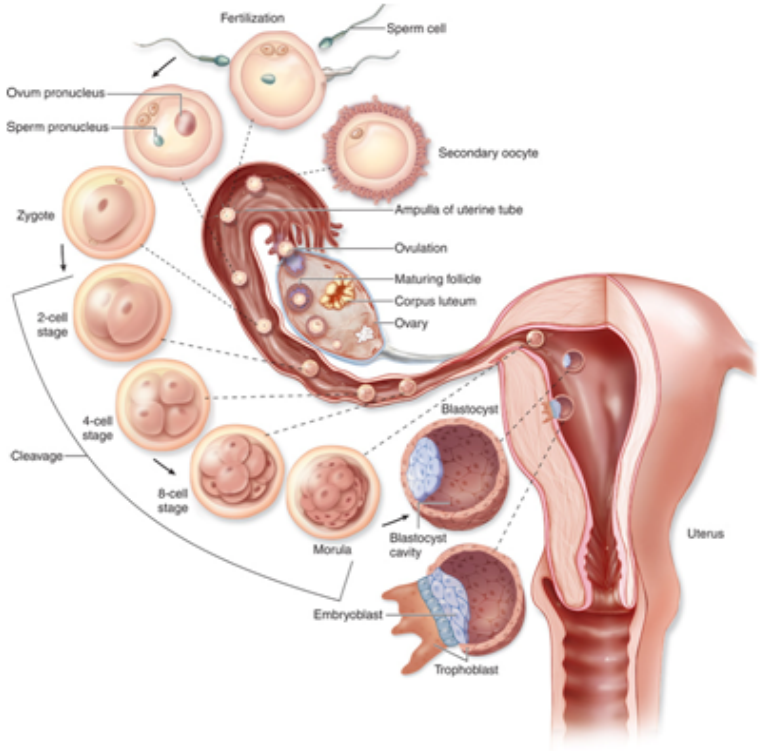


FIGURE 1.1: Development of the blastocyst in humans (Information, 2001).

the blastocyst and offers nutrients to the embryo and eventually form the invading placenta and membranes.

The zona pellucida surrounding the blastocyst begins to degenerate and decomposes completely to be replaced by the underlying trophoblast cells when the blastocyst reaches the uterus. The blastocyst adheres to the endometrium (innermost epithelial lining) of the uterus, inducing the rapid proliferation and differentiation of the trophoblast and invades the endometrium and its constituent blood vessels. This process is known as **implantation** of the embryo that facilitates nutrient exchange between the mother and the embryo. Usually, the embryo implants between eight to nine days post–fertilization in most of the successful human pregnancies (Wilcox et al., 1999).

Gastrulation

Once the blastocyst becomes completely embedded in the endometrium, morphological changes occur in the inner cell mass and it subsequently splits into two layers. One cell layer is called hypoblast which gives rise to the primary yolk sac (Gardner, 2001) and the other layer is called the epiblast which gives rise to the cells of the embryo proper. The epiblast further forms a three-layered structure by differentiating into the three primary germ layers ectoderm, mesoderm, and endoderm. This three-layer embryo structure is called the **gastrula**.

Organogenesis

Following gastrulation is the **organogenesis** stage that involves the molecular and cellular interactions between the newly formed germ layers, which ultimately constitute specific tissues and organs in the developing embryo. The ectoderm (outer layer of the embryo) differentiates to form the epidermis, and the neural crest and other tissues that give rise to the nervous system. It also develops the epithelium of the mouth and nose, sweat glands, lens of the eye, hair, and nails. The endoderm differentiates to form the epithelium of the respiratory system and digestive system (Zaret, 2001). It also develops into organs associated with the digestive system including liver and pancreas. The mesoderm that lies between the ectoderm and the endoderm gives rise to muscles, connective tissue, cartilage, notochord, kidney, blood, blood vessels, reproductive system organs and bone.

In human, an embryo develops all organ systems that an adult has by the end of the eighth week when cell differentiation is mostly completed. The embryo is known as a **fetus** at this stage. The fetal development lasts until birth. It involves the growth of all the tissues and organs of the organism which is accomplished through cell proliferation or accumulation of extracellular material (Wolpert et al., 2015). From seven months until birth, the majority of the physical growth of the fetus in human occurs which prepares it for the life outside the womb.

The early developmental events such as the formation of the body axes, gastrulation, and organogenesis are critical for normal embryogenesis (Gardner, 2001). These events greatly influence the development of different organ systems as well as cell fate. Significant disruptions in these dynamic *in utero* processes frequently cause embryonic lethality. Accordingly, understanding the underlying genetic basis of these processes is much needed. At present, birth defects or congenital malformation are severe and common threats to human life. Birth defects are defined as structural (*e.g.* congenital heart defects, neural tube defects, cleft lip and palate) or functional (*e.g.* Downs syndrome and autism) abnormalities that are recognisable at birth, and which can lead to physical or mental disability or even to death (Queisser-Luft et al., 2002). The American College of Obstetricians and Gynecologists (ACOG) reports that about approximately out of every 33 infants is born with a major birth defect. Developing effective strategies to prevent these birth defects is feasible only with a thorough understanding of the underlying cause(s), patterns and pathogenesis of these abnormalities. Although

most congenital malformations have a genetic basis, the specific gene(s) behind these developmental abnormalities remain unknown. A greater understanding of the causes of many birth defects can come from the identity of the genes involved and also from the phenotypic effects that come from disrupting their functions.

1.2 Gene Essentiality

All organisms have a number of genes encoded in their DNA. These genes possess diverse functionality and together regulate the overall characteristics of an individual. Some genes are absolutely required for the survival and development of an organism. These genes are termed as **essential** genes. Essential genes produce lethal phenotypes when mutated and these may lead to defects or death even at an early developmental phase. In contrast, **non–essential** genes are those that may be beneficial but not absolutely required for survival. A greater understanding of basic cellular processes, tissue–specific functions and development which are vital for life can come from identifying essential genes. Moreover, knowledge of essential genes promotes discovering potential drug targets in pathogens (Haselbeck et al., 2002) and helps to recognise human diseases (Huang et al., 2004). Thus, identification of essential genes is very important. However, the set of genes that are absolutely critical for sustaining life are still unknown for most organisms (Juhas et al., 2011).

Determining essential genes in human is a major progression from the sequencing of the human genome. We have already gained knowledge of the complex

genomic structure and the functional diversity encoded by approximately 21,000 protein–coding genes located in the human genome. But we still have not identified specifically which genes are absolutely required for our survival. Identification of human essential genes will be indicative of the importance of a specific gene in regulating our development and survival. Moreover, it will help us to learn how different developmental abnormalities including congenital birth defects and miscarriages, occur while normal development is perturbed.

1.3 Use of Mouse as a Model Organism

In developmental biology, non–human species are used as model organisms to understand biological phenomena, as human experimentation is impractical, unethical and expensive. This strategy is made possible due to the common descent of all living organisms and the conservation of genetic material over the course of evolution. The mouse (*Mus musculus*) is one of the preferred models to study human development since mouse and human demonstrate a high level of similarity in their genomes. Comparative analyses of mouse and human genomes revealed that they are organised in a very similar manner with a similar number of genes and regions of conserved linkage (synteny). Approximately 99% of human genes have a mouse orthologue (Guénet, 2005). Genome similarities may apply to most mammals, but the mouse has some additional properties which make it an ideal model for studying humans. Mice are small and they are easy to maintain in the laboratory because a mouse takes short time to reach sexual maturity (about 8

weeks). A mouse can have a litter every 3 weeks. They can produce approximately one litter of about 8 to 10 offspring every month which ensures their suitability for genetic analysis.

Through mouse knockout methodologies, researchers have already revealed biological functions of many human genes by examining their functional counterparts in mouse (Rangarajan and Weinberg, 2003; López-Bigas and Ouzounis, 2004; Oliver et al., 2007). A knockout mouse refers to a genetically modified mouse in which an existing gene is purposely inactivated or "knocked out" in a precise manner (Wolfer et al., 2002). A knockout is achieved by a technique called "gene targeting" where the functional copy of a gene is deleted or exchanged with a non–functional mutated version in mouse embryonic stem cells (ES cells). Mice are generated from the modified ES cells and the mutation remains present in every cell. This artificially introduced mutation abolishes the activity of the pre-selected gene that might cause changes in a mouse's phenotype including appearance, biochemical characteristics, behaviour *etc.* The resulting mutant phenotypes provide valuable information about the probable role of individual genes in maintaining normal physiological functions in mice, and by extrapolation, in humans (Huang et al., 2004). Prior experiments on knockout mice estimated approximately 40% of genes being essential (White et al., 2013). As it is impossible to experimentally investigate each of the human genes for essentiality, these experimental data from mice, therefore, can be used to infer mammalian gene essentiality.

1.4 Overview of Essential Gene Prediction

Identification of essential genes has already been accomplished for various organisms through various experimental methods including single gene knockouts (Crawley, 1999; Giaever et al., 2002; Kobayashi et al., 2003), conditional knockouts (Liu et al., 2000; Roemer et al., 2003), RNA interference (Cullen and Arndt, 2005; Kamath et al., 2003), and transposon mutagenesis (Gallagher et al., 2007). Experimental studies found approximately 40% genes in *S. cerevisiae* (Steinmetz et al., 2002), 30% genes in *C. elegans* (Simmer et al., 2003) and 40% genes in mouse (White et al., 2013) as essential. A previous study also found mouse essential genes in Chromosome 4, 7 and 11 using high–efficiency ENU mutagenesis screen (Hentges et al., 2007). All these experimental methods evaluating gene essentiality are very time–consuming, resource intensive and laborious. In addition, these techniques are restricted to few species and are not always feasible. Alternatives to these costly and challenging experimental techniques are machine learning methods (computational methods) which have received great attention in reliably predicting essential genes at a reduced cost and efforts in recent years. Prior studies reported different machine learning models integrating various sequence–derived gene features to aid essential genes identification (Chen and Xu, 2005; Gustafson et al., 2006; Seringhaus et al., 2006; Hwang et al., 2009; Acencio and Lemke, 2009; Deng et al., 2010; Yuan et al., 2012; Yang et al., 2014). These studies deciphered the relationships between many gene characteristics and the experimentally determined essential genes, and also established the viability of machine learning

approaches to reliably predict essential genes from these features.

Chen and Xu (2005) used neural networks (Hagan et al., 1996) and support vector machines (SVMs) (Cortes and Vapnik, 1995) to predict essential genes in *S. cerevisiae* from various features derived from high–throughput data sources including sequence, protein–protein interaction (PPI), gene expression and comparative genomic data. The most important features determining gene essentiality were found to be protein evolutionary rate, protein-protein interaction connectivity, gene duplication rate and gene expression. The authors suggest that their approach could also be applicable to other organisms with the availability of high-throughput data.

Saha and Heber (2006) employed the dependencies between gene essentiality and various attributes derived from sequence, PPI and comparative genomics data to distinguish essential and non–essential genes in *S. cerevisiae*. An improved simulated annealing optimisation algorithm (Kirkpatrick, 1984) was used to determine the weight of different attributes which signified their potential influence on essentiality prediction. Protein–protein interactions connectivity, paralogy score and phylogenetic conservation score were found to have more importance for the prediction of essential genes. The authors used the weighted k–nearest neighbour (KNN) algorithm (Cover and Hart, 1967) and SVMs to extract relationships between essentiality and gene attributes in a *S. cerevisiae* training dataset. Predicting genes in a separate test set by the trained classifiers also report the efficiency of these machine learning methods.

Gustafson et al. (2006) built a Naive Bayes classifier (Russell et al., 2003; Rish, 2001) using different genomic and protein sequence features to classify essential genes in *S. cerevisiae* and *E. coli*. A feature selection algorithm was used to decide which subset of features is optimal for predicting essential genes. Phyletic retention, which measures the number of closely related organisms that have an orthologue, was found to be most informative for essentiality prediction. The authors are the first to report the importance of using this feature in predicting essential genes. Other significant attributes were number of paralogues (number of gene duplicates), gene upstream size (distance to the closest gene), protein length and codon bias. The efficacy of the Naive Bayes classifier was assessed by cross–validation.

Seringhaus et al. (2006) proposed an ensemble classifier to predict gene essentiality in *S. cerevisiae* from sequence-derived features only, arguing that, since functional genomics information are unavailable for most organisms, training a classifier using them is not rational. The ensemble classifier was developed combining the output of seven different machine learning methods which include Random Forest (Breiman, 2001), logistic regression (Le Cessie and Van Houwelingen, 1992), Naive Bayes, C4.5 decision tree (Quinlan, 1993), decision stump boosted through Adaboost (Freund and Schapire, 1996), alternating decision tree (Freund and Mason, 1999) and zeroR rule. The importance of each feature was measured using the Naive Bayes method. Protein hydrophobicity, cellular localisation, GC

content, rare amino acid ratio and codon adaptation were reported as more informative features of essential genes. Assaying gene essentiality in an unstudied and closely related yeast species *S. mikatae* evaluated the effectiveness of this classifier.

Hwang et al. (2009) explored a number of topological features in the PPI networks with an attempt to reveal their relationships with essential genes in *S. cerevisiae* and *E. coli*. Many topological properties were found to statistically differentiate essential and non–essential genes. The topological properties together with a number of sequence properties were used in building an SVM classifier to infer which genes are essential and which genes are non–essential. The authors suggest that topological properties of interaction degrees, betweenness and neighbours' intra-degree along with phyletic retention and the length of the open reading frame (ORF) are capable of making reliable predictions of essential genes in both of organisms through machine learning approach.

Acencio and Lemke (2009) demonstrated that integration of protein–protein interaction information along with cellular localisation information and biological processes can aid in the reliable prediction of gene essentiality. Using these features, a decision tree based meta classifier was developed which showed it effectiveness in determining gene essentiality in S. cerevisiae. This meta classifier was generated averaging the prediction results of eight different machine learning methods which include Naive Bayes tree, REPtree (Witten and Frank, 2005), Random Forest, random tree (Witten and Frank, 2005), J48 decision tree (Quinlan, 1993), logistic model tree (Landwehr et al., 2005), best–first decision tree (Shi,

2007), and alternating decision tree. The number of protein–protein interactions, the number of regulating transcription factors and nuclear localisation were found to be most important properties for essential genes prediction. The number of protein-protein interactions was found as the most significant feature among all.

Deng et al. (2010) built an ensemble classifier combining logistical regression model, C4.5 decision tree, CN2 rule and Naive Bayes method to predict essential genes from sequence and functional genomic features in four distantly related bacterial species namely *E. coli*, *P. aeruginosa*, *A. baylyi* and *B. subtilis*. The Naive Bayes technique was used to measure the influence that a feature has on predicting gene essentiality. Protein domain enrichment score (DES), which represents the ratio of the occurrence frequency of each protein domain between an essential gene and the total genes of the target organism, was found as the most predictive feature among all. Other dominant features were cellular localisation, protein aromaticity and gene conservation rate. For each pair AB of organisms, the classifier was trained to learn traits related to essential genes in A, and further was applied to make predictions in the other organism (B) and vice versa. Overall, this study suggests that essential genes prediction could be transferable between distantly related organisms through a machine learning approach trained on sequence features.

The above-mentioned studies predicted essential genes of either bacteria or unicellular yeast organisms taking the information of the experimentally identified essential genes in those species into consideration. The large genome size and

developmental complexity of mammals has obstructed the ample experimental analysis of essentiality in these organisms. Though mouse knockout experiments have already proved useful in identifying a subset of mammalian essential genes, the entirety of the mouse genome has not yet been experimentally examined. Yuan et al. (2012) analysed a subset of mouse essential and non-essential genes defined by the knockout experiments and showed the feasibility of predicting mouse essential genes through three machine learning classifiers (logistic regression, SVM and random forests) trained on a large number of gene features. The predictive power of these features was examined using the least absolute shrinkage and selection operator technique (LASSO) (Tibshirani, 1996). Among all, gene evolutionary age, protein connectivity and paralogue sequence identity were found to have more influence on defining essentiality. However, the classifier gave poor prediction accuracy and it failed to justify why these features are vital. Though this study used 491 features to predict gene essentiality in mouse, a bulk of these features essentially represent the similar information and could have been reduced into smaller number. In addition, the miRNA target site data was handled strangely – the authors counted each miRNA sequence as a separate feature. This would be analogous to including all gene lengths in the training set as individual features rather than stating that gene length is one feature.

A recent study by Yang et al. (2014) explored a large number of topological and biological properties to discern essential and non–essential genes in humans. Most of these properties were statistically discriminative. The authors conducted

a Gene Ontology (GO) enrichment analysis to infer the functional uniqueness of essential and non–essential genes. The SVM classifier (with radial basis kernel function) was developed based on these topological and biological properties to learn the characteristics underlying gene essentiality. With a prediction accuracy of 72.87%, this classifier indicates the usefulness of machine learning approaches in predicting human essential genes.

A bioinformatics study also discovered that sequence features could determine disease genes (Kondrashov et al., 2004). The authors examined 18 different sequence features between disease and generic human gene sets which were used to develop a multilayer neural network model for classifying disease and non-disease genes reliably. In addition, (Dickerson et al., 2011) presented a computational framework to test the association of the human orthologues of mouse essential genes with human diseases by means of a number of sequence features. This study found the connections of disease genes with protein–protein interactions, nuclear localisation and autosomal dominant mode of inheritance.

Currently, two databases – DEG (Zhang and Lin, 2009) and OGEE (Chen et al., 2012a) are available that accumulated the experimentally validated essential genes for various organisms. The DEG database includes known essential genes in *M. genitalium*, *V. cholerae*, *H. influenzae*, *S. aureus*, *E. coli*, and *S. cerevisiae*. The OGEE database organises experimentally tested essential and non–essential genes in *D. melanogaster*, *A. fumigatus*, *B. subtilis*, *M. genitalium*, *S. cerevisiae* and mouse along with a number of features that prior studies identified or assumed

to have an influence on gene essentiality. These features include gene expression profiles, conservation across species, duplication status, involvement in embryonic development, and evolutionary origins.

1.5 Machine Learning Methods

Machine learning is a branch of computer science originating from the artificial intelligence (AI) that has become one of the most accepted approaches for data processing and data analysis in recent years. It is concerned with developing systems or models that can learn from data (Witten and Frank, 2005). These models or systems are not explicitly programmed; rather machine learning allows the models to automatically learn hidden knowledge from training data and to apply this knowledge in making reliable decisions when exposed to new data. Machine learning approaches have been applied to a broad range of areas including spam e–mail filtering, image and speech recognition, effective web search, and genetics and genomics.

One of the major tasks a machine learning system can be trained to perform is classification. The input to a classification problem is a training dataset comprised of a number of observations. Each observation is characterised by a number of features. A classification problem seeks to determine to which group (or class) an observation belongs based on its properties. For example, a system could be trained to classify patients, and then be used to decide whether a new patient should be assigned a diagnosis. Machine learning classification methods are mainly

segregated into two categories: supervised learning methods and unsupervised learning methods.

Unsupervised learning methods are used when no class labels are assigned to the observations in the training dataset. The classifier training, therefore, cannot use the class information to map each observation into a unique class. Instead, *unsupervised learning* methods search for hidden relationships in the data. A well-known unsupervised classification method is clustering. Clustering divide the observations in the training dataset into a set of disjoint groups or clusters with each cluster having observations that are similar to each other (share similar properties). In contrast, *supervised learning* methods (Figure 1.2) are used when each observation in the training dataset is labelled with a known class. The goal of these methods is to build a classifier for the training dataset based on the class information. The classifier learns traits of each class and applies this knowledge to predict the class of each observation in the test dataset.

Supervised learning methods are further divided into two groups. Methods in the first category are used to build a single best classifier. Examples of these methods are Naive Bayes, Artificial Neural Network (ANN), SVM and decision tree. Methods in the second category are called *ensemble* methods, which build multiple classifiers and then combine their outputs to make a prediction about an observation. Random Forest is the most common example of ensemble methods. The following subsections present brief descriptions of some of the well–known supervised learning techniques used for classification.

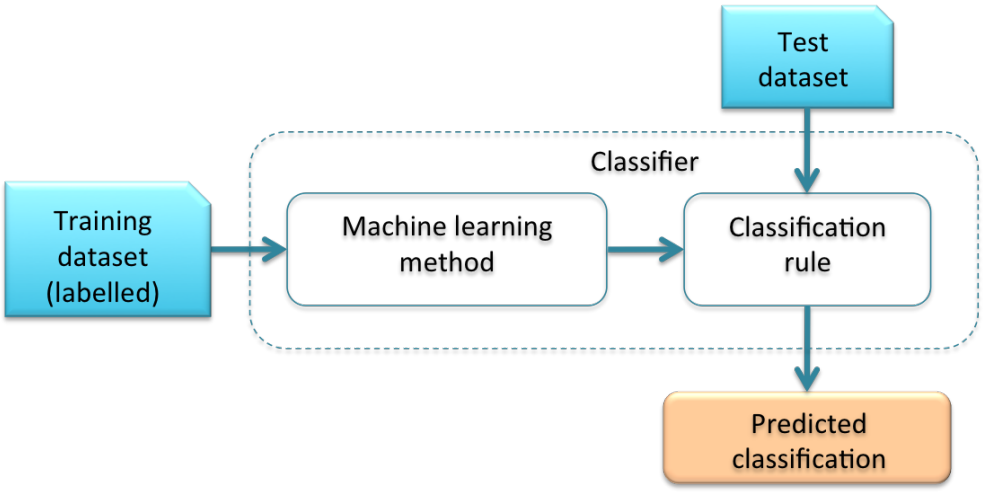


FIGURE 1.2: Workflow showing classification with supervised learning methods.

1.5.1 Naive Bayes

The Naive Bayes classifier (Rish, 2001) is a probabilistic learning method. It classifies an observation on the basis of an assumption that each input feature is conditionally independent of every other input feature. The input features are dependent on the class variable. The Naive Bayes method uses Bayes rule to recognise the most probable class for the observation. Given a class variable C_k and a dependent feature vector $x = (x_1, x_2, ..., x_n)$ with n features, Equation 1.1 (Bayes rule) is used to compute a probability that the observation x belongs to class C_k.

$$p(C_k|\mathbf{x}) = \frac{p(C_k)p(x_1, x_2, ..., x_n|C_k)}{p(x_1, x_2, ..., x_n)} = \frac{p(C_k)\prod_{i=1}^{n} p(x_i|C_k)}{p(x_1, x_2, ..., x_n)} \tag{1.1}$$

Here, $p(x_1, x_2, ..., x_n)$ is the probability of the observation $\mathbf{x}$; $p(C_k)$ is the probability of the class C_k; $p(x_i|C_k)$ is the probability of x_i given that the class is C_k that

the classifier estimates from the training dataset. The observation $\mathbf{x}$ is predicted belonging to class C_k with the highest $p(C_k|\mathbf{x})$ value.

Advantages – The advantages of Naive Bayes classifiers are: i) simple, easy to implement and scalable, ii) require considerably small training dataset to obtain good results, iii) converge quickly if the assumption of conditionally independent properties holds.

Disadvantages – The disadvantages are: i) conditional independence assumption does not always hold, ii) cannot model dependencies among various properties, iii) accuracy reduces with the increased size of the training dataset.

1.5.2 Artificial Neural Network (ANN)

Artificial neural network (ANN) (Hagan et al., 1996) is a machine learning approach that mimics the structure and function of a biological neural network. A typical ANN system is comprised of several artificial neurons or nodes that are interconnected by edges or signal connectors. The nodes are the computational units that receive data and process them to get an output, whereas the connectors carry processed results to and from nodes like synapses. The nodes in the ANN often have a layered structure and work collectively as a group rather than individuals. ANN receives data from a set of nodes usually known as input layer and the final output is expressed via nodes in the output layer. The third set of nodes is also often used between the input and output layers for data processing.

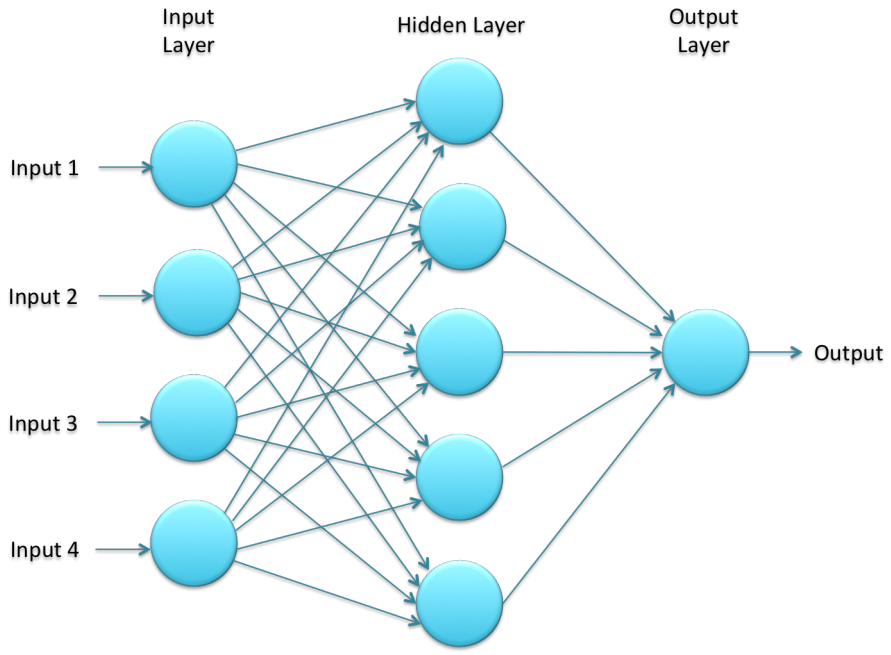


FIGURE 1.3: Structure of an ANN with single hidden layer. Here, circles represent the artificial neurons and the edges represent the connections between these neurons.

These nodes comprise the hidden layer. Figure 1.3 demonstrates the structure of an ANN with one hidden layer.

All nodes of the ANN have fixed computation associated with them. The connectors on the other hand store parameters (weights) that are adjusted during the learning process. The structure of the ANN thus is defined by the connection pattern of the neurons of different layers, the weights of the connector set up by the learning process and the activation function that converts the weighted input of a neuron into the output.

ANN can be trained to separate samples into different classes. The training process involves finding the common features that observations of a known class exhibit. These features and the class information are then used to train the ANN

by adjusting weights of the connectors. The weights of the connectors are adjusted or optimised until the error in predictions is minimised and a specific level of accuracy is achieved. The performance of the classification is further confirmed by feeding the ANN with new observations belonging to a known class and monitoring its response.

Neural networks with a single hidden layer are limited to the problems where there exist complex relationships between input features and desired outputs. Alternative to these conventional techniques are the deep neural networks (DNNs), which attempt to model high level of abstractions in data (Figure 1.4). In recent years, various deep neural network architectures have become popular tools for machine learning including deep feed–forward neural networks, convolutional deep neural networks and recurrent neural networks. These deep learning architectures are neural networks with multiple hidden or processing layers (usually more than 2), composed of multiple linear and non–linear transformations, which can efficiently learn complex mappings between input features and outputs. DNNs excel at those problems where large amounts of training data are available. DNNs have shown state–of–the–art performance in speech recognition, natural language processing, image recognition, recognising protein folds, structures and understanding their functions (Jo et al., 2015; Spencer et al., 2015), and also in genomics (Chen et al., 2016).

Advantages – The advantages of ANNs are: i) flexible and adaptive, ii) good at perceiving hidden and complex non–linear relationships between dependent and

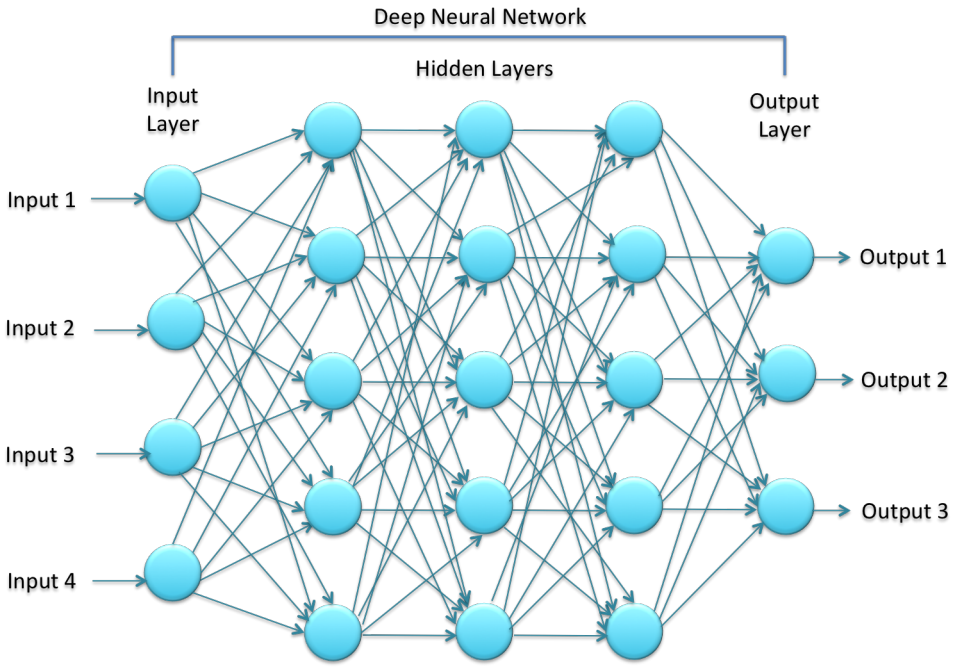


FIGURE 1.4: Structure of a deep neural network with three hidden layers. Here, circles represent the artificial neurons and the edges represent the connections between these neurons.

independent variables.

Disadvantages – The disadvantages are: i) training dataset has to be very large to achieve good performance, ii) long training time, iii) provide multiple solutions linked to local minima, iv) prone to overfitting.

1.5.3 *k*–Nearest Neighbours (*k*–NN)

The k–Nearest Neighbours (k–NN) algorithm is an instance–based or lazy learning method which has no training phase (Aha et al., 1991). This method attempts to predict the class of a new observation by locating k nearest observations or neighbours in the training dataset. k–NN classification follows three steps:

1. Determining the value of k.

2. Determining nearest neighbours of an observation in the test dataset. The nearest neighbourhood is determined based on a Euclidean distance measure.
3. Determining the class using the class information of neighbours. The test observation is assigned a class to which most of its nearest neighbours belong.

Advantages – The advantages of k–NN classifiers are: i) simple to implement, ii) analytically tractable, iii) highly adaptive to local information.

Disadvantages – The disadvantages are: i) requires large data storage, ii) runs slowly, iii) biased by the value of k, iv) sensitive to outliers and noise for small values of k, iv) larger values of k increase the computational complexity, v) highly vulnerable to the curse of dimensionality, *i.e.*, prediction accuracy degrades for large–scale features as they dominate the distance metrics.

1.5.4 Support Vector Machine (SVM)

The support vector machine (SVM) is a supervised machine learning technique used for classification and was first coined by (Cortes and Vapnik, 1995). Basic SVMs work as binary classifiers separating inputs in only two classes, but the design of multi–class SVM models are not rare. As a binary classifier, this method attempts to classify observations by dividing the feature space into two subspaces. This is done by finding an optimal hyperplane that separates the observations in the training dataset in such a way that as many observations as possible belonging to a class remain in the same subspace. SVM classifies an observation in the test dataset based on its position relative to the optimal hyperplane.

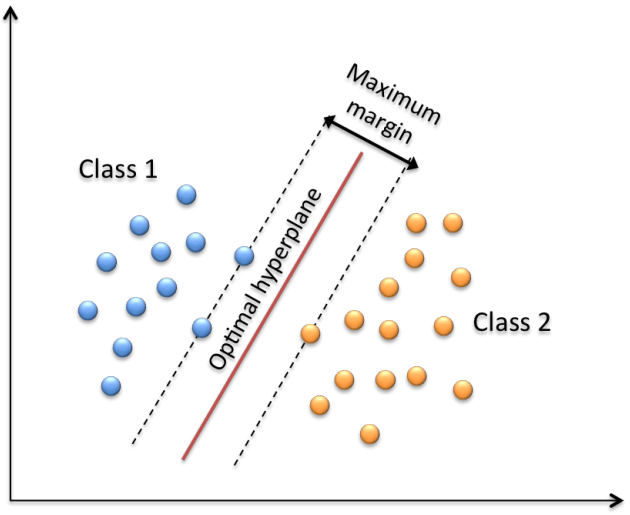


FIGURE 1.5: Depiction of an optimal hyperplane in SVM dividing a two-dimensional feature space into two subspaces.

There will be multiple hyperplanes that could offer perfect separation of the observations (Ben-Hur et al., 2008). SVM finds the optimal hyperplane based on the concept of a margin (Figure 1.5). Given a binary classification problem with classes a and b, the margin of a separating hyperplane H is the Euclidean distance between the closest observation of class a and the closest observation of class b to H. SVM attempts to choose an hyperplane with the maximum margin (hard margin). The hyperplane with maximum margin is obtained by finding two hyperplane H_a and H_b, such that:

1. H_a passes through at least one observation from class a.
2. H_b passes through at least one observation from class b.
3. There exist no observations between H_a and H_b in the training dataset, *i.e.* they must be parallel.
4. The distance between H_a and H_b is the highest among all possible hyperplane pairs.

The training observations that lie on H_a and H_b are the only ones that are required to define H and are called support vectors. Choosing the hyperplane in this way lowers the expected generalisation error of the classification (Kotsiantis et al., 2007). To classify an observation correctly, the above-mentioned hard margin approach of SVM requires the observations in the training dataset (original feature space) to be linearly separable (correctly classified). In practice, data often have noise and may not always be perfectly linearly separable. SVM employs soft margin and kernel functions to resolve this issue. The idea of the soft margin is to incorporate a penalty term to allow misclassification. The penalty term is used to penalise the training observations that fall inside the margin. However, this still needs the separating hyperplane to be linear, and often a better classification accuracy is achieved by considering a non–linear boundary between classes.

To implement non–linear class boundaries, SVM utilises kernel functions that map the training observations into a high dimensional feature space where the training observations are linearly separable. The linear SVM classifier relies on the computation of dot product $f(x).f(y)$ between all (x, y) pairs of training observations. A kernel function k circumvents the increased computational complexity and data overfitting of the explicit mapping of the training data into the high dimensional feature space by replacing every dot product such that $k(x, y) = f(x).f(y)$, therefore allowing the maximum margin hyperplane to fit into the high dimensional feature space. Common kernel functions include linear, polynomial and sigmoid

kernels and the radial basis function (RBF). This works because a linear hyperplane constructed in the high dimensional feature space represents a non–linear boundary in the original feature space (Cord and Cunningham, 2008).

Advantages – The advantages of SVMs are: i) high prediction accuracy, ii) robust to noise and outliers, iii) less prone to overfitting, iv) good generalisation ability even with smaller training data.

Disadvantages – The disadvantages are: i) memory intensive, ii) runs slowly for both training and testing, iii) not suitable for larger datasets, iv) likely to offer poor performance if the number of attributes is much larger than the number of observations, v) difficult to extend for multiclass classification, vi) need to select a suitable kernel function, vii) it is also hard to determine which features are most useful in the classification.

1.5.5 Decision Trees

A decision tree is a machine learning technique where a tree–like structure is used as a model to classify an observation by learning simple decision rules inferred from representative features (Breiman et al., 1984). A decision tree has three types of nodes, namely root node, internal nodes and leaf nodes with each node representing a feature. A root node is the one that has no incoming edges. Each internal node consists of one incoming edge and two or more outgoing edges. However, a leaf or terminal node, which has one incoming edge, but no outgoing edges, represents a class label. All internal nodes along with the root node represent a set of disjoint

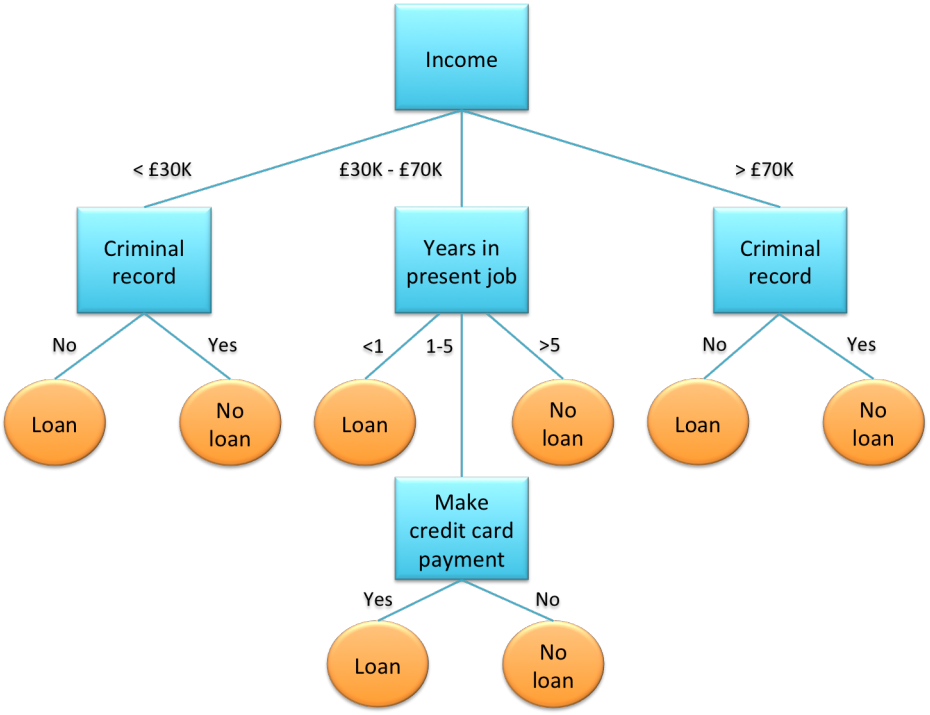


FIGURE 1.6: Example of a decision tree. Here, squares represent the root as well as internal nodes and circles represent leaf nodes. The text inside each internal node denotes the feature that is used as a best cutpoint and the edges denotes the feature values that are used to partition the observations in the subspace of a node.

rules inferred from the values of features to separate the training observations and each branch of the tree represents the decision (the outcome of the rules). A path from the root node to a leaf node denotes a classification rule. Decision trees classify an observation by sorting them down the tree starting from the root to a leaf node. At first, it applies the decision rule linked to the root node to the observation. The outcome of this rule selects the appropriate branch to follow. This leads either to an internal node to employ another decision rule or to a leaf node. The observation is assigned the class information associated with the leaf node. Figure 1.6 shows an example of a decision tree.

A decision tree is built in a top–down manner starting with a root node. It

involves repeated partitioning of the feature space into disjoint smaller subspace containing training observations with similar feature values until a leaf node is formed. In constructing the decision tree, for each node n, a decision must be made to choose the best cutpoint for n. A cutpoint is the values of a feature f which are used to split the subspace (characterised by n) into classes. The superiority of a cutpoint is evaluated in terms of the purity of the induced subspace. A subspace is called pure when all training observations in it are from the same class. The best cutpoint is the one which gives the maximum purity or alternately minimum impurity. A common approach to estimate impurity is the gain ratio. Gain ratio (Equation 1.2) measures the amount of information gained from a feature in classifying the training observations. It is computed on the basis of intrinsic split information (Equation 1.3) and information gain (Equation 1.4). Information gain is measured by entropy (Equation 1.5). The feature with the maximum gain ratio is selected as the root node. Values of this feature are then used to for subsequent partitions.

$$GainRatio(S, f) = \frac{InfoGain(S, f)}{SplitInfo(S, f)} \tag{1.2}$$

$$SplitInfo(S, f) = \sum_{i=1}^{C} \frac{|S_i|}{|S|} log_2 \frac{|S_i|}{|S|} \tag{1.3}$$

$$InfoGainR(S, f) = Entropy(S) - \sum_{v \in values(F)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{1.4}$$

$$Entropy(S \text{ or } S_v) = \sum_{i=1}^{C} - p(i) log_2 p(i) \tag{1.5}$$

Here, S is the training observations in node n; S_i is a subset of S containing observations belonging to class i; S_v is a subset of S with value v of the feature f; $|S|$ and $|S_v|$ are the number of observations in S and S_v; $p(i)$ is the proportion of observations belonging to class i and C is the set of classes.

Another common estimate of the impurity is called the Gini index. This computes the impurity of the training observations S in the node n using Equation 1.6. The impurity of a cutpoint is computed as the weighted sum of the impurities of the two child nodes L and R (Equation 1.7).

$$Gini(n) = 1 - \sum_{i=1}^{C} p(i)^2 \tag{1.6}$$

$$Gini_split(n) = \frac{|S_L|}{|S|} Gini(L) + \frac{|S_R|}{|S|} Gini(R) \tag{1.7}$$

Here, $S_L \subseteq S$ and $S_R \subseteq S$ are the training observations in child nodes L and R, respectively; $p(i)$ is the proportion of the training observations in S that belong to class i.

The overall algorithm for constructing a decision tree is as follows:

1. Set the root node as the parent node.
2. Split the parent node at the feature f into child nodes based on the minimum impurity.
3. Assign training observations to new child nodes.
4. Stop if all leaf nodes are pure. Else, repeat step 1 to 3 for each new child node.

Due to the repeated partitioning, larger decision trees are likely to overfit and give poor generalisation accuracy. One way to lessen overfitting is not to grow the decision tree to its full size. This can be achieved by pruning the tree which decreases the size of the tree by removing tree nodes that do not provide additional information (Kotsiantis et al., 2007). Moreover, this decreases the complexity of the final classifier as well as increases the accuracy of the classification.

Advantages – The advantages of decision trees are: i) simple, ii) easy to understand and interpret, ii) runs fast, iii) suitable for large–scale analysis, iv) can handle irrelevant features, v) can handle missing values, vi) can handle non–linear relationships.

Disadvantages – The main disadvantage of decision trees is they are prone to overfitting without proper tree pruning. Also, finding the optimal decision tree is a challenge.

1.5.6 Random Forest

An alternative to the single optimal classifier is the ensemble classifier which includes multiple suboptimal classifier or base learners. Rather than using the classification result produced by a single classifier, an ensemble classifies an observation by aggregating the classification results of all base learners. This approach is particularly beneficial in situations where finding and choosing an optimal classifier is infeasible (Ditterrich, 1997). In those classification problems, the ability of an

ensemble classifier to approximate an optimal solution by combining a set of approximations can make it superior to that of a single best classifier. One of the popular ensemble classifiers is the Random Forest (Breiman, 2001) which operates by constructing multiple decision tree models at the training time. It is one of the most accurate supervised learning methods in recent times. Each decision tree in a Random Forest represents one class of observations that are being considered. Decision trees are constructed during the learning process with the training data.

Random Forests mainly rely upon two parameters to control their growth: $numTrees$, the number of decision trees to be built and $numFeatures$, the number of random subset of features to assess at each tree node. Let $numTrees = T$ and $numFeatures = m$. Each of the T decision trees is constructed in a top–down manner starting with a root node by selecting a set of N observations of size n at random with replacement from the training dataset and selecting the most significant features of these samples as the tree nodes. At each node a, the m number of features is selected at random from n features to grow the tree and the most significant feature that provides the best binary split on that node is selected among all according to an objective function. Feature significance is generally estimated using the Gini index (mentioned in section 1.4.6). This significant feature splits node a into left (L) and right (R) child nodes with a set of $N_L \subseteq N$ and $N_R \subseteq N$ samples in L and R nodes, respectively. This process continues at each node until the decision tree cannot grow further.

To classify a new gene, the features of the gene are tested with each of the

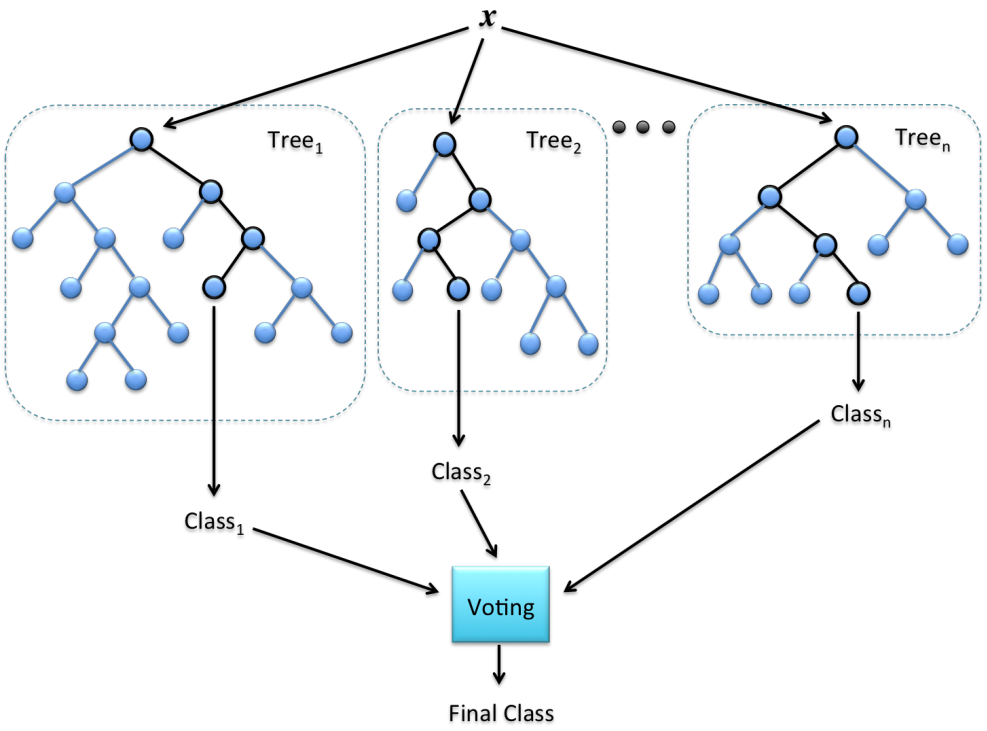


FIGURE 1.7: Example of a Random Forest classification. Here, x denotes a test observation, n denotes the number of decision trees in the Random Forest, and $Class_z$ denotes the classification given by the decision tree z. The final class information of x is determined by counting the most votes.

decision trees present in the random forest. Each tree gives a classification score or "vote" and the class with the most votes is selected as the class to which the gene belongs (Figure 1.7).

Advantages – The advantages of Random Forests are: i) high level of accuracy, ii) runs faster, iii) scale well with large datasets, iv) robust to outliers, v) maintains accuracy while a large number of features are missing in the training dataset, vi) easy to interpret class predictions, vii) resistant to overfitting.

Disadvantages – Random forests can occasionally suffer from overfitting for noisy datasets.

1.6 Are Gene Duplicates as Essential as Singletons?

Gene duplication may have an important role in determining gene essentiality. Gene duplication is a frequent event in the evolution of organisms, especially for multicellular eukaryotes, through which new genes and/or biological functions are added to increase the genomic complexity (Zhang, 2003; Lynch and Conery, 2003). It results in a single gene having multiple copies (paralogues) in the genome. Duplicated genes further undergo mutations over time that either cause the functional loss in one copy or functional differences between them (Ohno, 1970). Understanding the role of gene duplicates and their associations with phenotypic effects of gene deletion, therefore, have received much attention.

Early studies on mouse found mild or even no phenotypes by knocking out one copy of a duplicate gene (Joyner et al., 1991; Saga et al., 1992). This prompted the hypothesis that discerning the function of each duplicate gene would be difficult by knocking out individual genes because many duplicate genes are functionally redundant (Thomas, 1993; Cooke et al., 1997). This opinion was reinforced by some subsequent studies which revealed that duplicate genes could compensate for the functional loss of their duplicate copies due to their overlapping function and expression and thus, deleting a duplicate gene has little phenotypic effect (Gu, 2003; Gu et al., 2003; Kamath et al., 2003; Conant and Wagner, 2004; Guan et al., 2007; DeLuna et al., 2008). At least 25% of gene deletions showed no noticeable phenotypic effect due to the presence of their duplicate gene copies in *S.*

cerevisiae (Gu, 2003). Moreover, genome–wide single gene deletion experiments demonstrated that 29% of singletons, compared to 12.4% of duplicates, are essential to the viability of *S. cerevisiae* (Gu et al., 2003). Likewise, only 2.3% of duplicate genes showed lethal phenotypes in *C. elegans* compared to 7.6% of singletons using a genome–wide gene knockdown experiment (Conant and Wagner, 2004), thereby confirming the tendency of duplicate genes to be less essential than singletons.

However, further studies with mouse knockout phenotypes did not support the expected trend of duplicate genes being less essential (Liao and Zhang, 2007; Liang and Li, 2007). With proportions of ∼55% (Liao and Zhang, 2007) and ∼48% (Liang and Li, 2007) essential genes in both duplicates and singletons, it was reported that mouse duplicate genes are just as critical as singletons. These conflicting results in the mouse are further disputed in a recent study which found duplicate genes as an important factor in the genetic robustness of human (Hsiao and Vitkup, 2008).

The outcomes of previous studies question what factors govern essentiality in singletons and duplicates. Xun Gu (2003) showed that in *S. cerevisiae* the protein products of singleton genes are more likely to have high protein connectivity, whereas duplicate proteins are less connected. The author rationalised this observation claiming that since high protein connectivity means high functional importance, the functionally critical genes are more likely to have low gene duplicability. Two follow–up studies on *S. cerevisiae* (Gu et al., 2003) and *C. elegans*

(Conant and Wagner, 2004) showed that the proportion of essential genes rises as the protein sequence divergence increases between a duplicate gene and its closest paralogue. In contrast, a separate study (Liang and Li, 2007) detected that mouse gene essentiality is positively correlated with the existence of duplicate genes; also, highly connected protein are more likely to have the high degree of gene duplicability. However, these contradictory results could be the result of biases in the mouse knockout dataset since mouse geneticists usually choose to report those genes that show evident phenotypes in the knockout experiments.

By analysing the divergence in protein sequence, expression and evolutionary conservation for approximately 3,900 targeted mouse genes, Liao and Zhang (2007) rejected any potential biases exist in the knockout dataset, which led them to affirm that mouse duplicate genes rarely compensate for each other. Although both of the above-mentioned studies agreed on the essentiality of mouse duplicate genes, these failed to rationalise why essential genes tend to be more duplicated in mouse, while genes in yeast and nematode show the opposite trend.

Makino et al. (2009) discovered that the knockout dataset for mouse and Drosophilla is greatly biased towards the presence of developmental genes and genes derived from old duplications. They showed that developmental genes are more likely to be essential than the non–developmental genes, irrespective of gene duplication. In particular, the essentiality of developmental duplicates was higher than the developmental singletons for both organisms. A subsequent study gave a system–level explanation of this finding (Liang and Li, 2009). In addition, Makino

et al. (2009) revealed that genes duplicated by a WGD event are more essential than the ones from single gene duplication (SGD) events.

However, Su and Gu (2008) showed that the current mouse knockout dataset is highly enriched in genes derived from old duplications, giving overestimated essentiality in mouse duplicates. This study found that ancient duplicate genes are more likely to be essential than singletons. In addition, Chen et al. (2012b) reported the propensity of older genes to be more essential in mouse, regardless of duplication status. It was also found that singletons are more likely to be essential than duplicate genes of the same age. A very recent study further confirmed that the effect of duplicate genes on the genetic robustness of mouse is duplication–age dependent (Su et al., 2014). After ruling out the confounding effect of WGD duplicates, protein–protein connectivity, coding–sequence conservation, and functional bias, this study found that mouse singletons have a higher percentage of essential genes than duplicates. Even though the bias–corrected data confirmed the critical role of mouse duplicates in genetic robustness, it could not explain why some duplicate gene copies are both essential and some are non–essential. Further investigation is required to resolve this issue.

1.7 The Developmental Hourglass Model

One of the central concerns in developmental biology is how to formulate the relationships between evolution and embryonic development. The biogenetic law of

Ernst Haeckel (published in 1866) stated that embryogenesis recapitulates evolution. This is because, despite their divergent appearances, all animals develops through an analogous number of developmental stages, starting from a single fertilised egg, proceeding through cleavage, blastula, morula, gastrula and organogenesis stages, before the later developmental stages give the complex body structure which leads to the adult form (Richardson and Keuck, 2002). Though it has been accepted that embryogenesis cannot just be a recapitulation of evolution, no definite consensus regarding the relationships of evolution with embryogenesis has achieved yet.

The relationships between embryonic development and animal evolution can be revealed from four laws of animal development proposed by Karl von Baer (Baer, 1828). These laws state that: (1) common characteristics of the phylum to which an embryo belongs develop earlier than the specialised characteristics; (2) the special characteristics develop from the most common characteristics; (3) animal embryos progressively become dissimilar from each other as development proceeds; (4) the embryo of one animal does not resemble the adult form of another animal. Overall, the third law suggested that embryos of different species from the same phylum pass through a developmental phase during which they closely resemble each other morphologically. They become different from each other progressively due to the advent of distinguishing features. This law remained unchallenged until it was found unfit for early embryogenesis (Hazkani-Covo et al., 2005) as the earlier stages often vary widely even among the closest species. Duboule

(1994) and Raff (1996) individually expanded this law by proposing a morphological model during embryogenesis in terms of the morphological divergences in the early embryogenesis, which has since become much acknowledged (Prud'Homme and Gompel, 2010; Kalinka and Tomancak, 2012). It asserts that animal embryos are divergent in morphology to other embryos of the same phylum at early and late embryogenesis but they are morphologically similar at mid–gestation. This morphological pattern during embryogenesis is called the developmental hourglass model. The stage during mid–embryogenesis where embryos are morphologically conserved is known as the phylotypic stage (Sander et al., 1983; Elinson, 1987) or phylotypic period (Richardson, 1995). Molecular interpretations of the developmental hourglass model imply that embryos have most similar gene expression patterns at the phylotypic stage.

Recently much attention has been paid to the hourglass model of development in animal embryos. Initially, different studies have asserted different gene features rationalising the developmental hourglass model (Davis et al., 2005; Irie and Sehara-Fujisawa, 2007). Genes that are expressed in the phylotypic stage were found to have highly similar protein sequences in mouse and human (Hazkani-Covo et al., 2005). Recent studies succeeded in observing the presence of an hourglass pattern also in transcriptomes for several vertebrates. A study on zebrafish reported that genes expressed in mid–embryogenesis are evolutionarily older than genes expressed early or late embryogenesis (Domazet-Lošo and Tautz, 2010) when plotting the the average of the phylogenetic age of genes (quantified by their

"phylostrata"), weighted by their microarray signal intensity values across developmental time point. Moreover, a study on Drosophila (Kalinka et al., 2010) demonstrated that genes that are expressed in mid–embryogenesis have the highest expression similarities. Likewise, Irie and Kuratani (2011) reported the highest gene expression conservation between zebrafish, frog, chicken, and mouse, for genes expressed in the presumptive phylotypic stage.

In contrast to the above-mentioned studies, a study on zebrafish did not observe the hourglass pattern with respect to gene age and expression (Piasecka et al., 2013). The authors claimed that applying a standard log-transformation to the microarray signal intensity values changes the overall pattern that Domazet-Lošo and Tautz (2010) found and indicates that older genes are expressed preferentially in early development. This early conservation model was supported with respect to gene duplication and new gene introduction that are the most rare for genes expressed in early development. However, the authors found the hourglass pattern at the regulatory level, with sequence of regulatory regions being most conserved for genes expressed in mid–embryogenesis among all other properties like gene sequence conservation, gene age, gene orthology relationships and gene expression conservation, in contrast to earlier studies. The authors claimed that both hourglass and early conservation models are valid for embryogenesis, but with respect to different genomic features.

Furthermore, the hourglass model was also found to be hold for plant embryogenesis in terms of gene age and sequence conservation (Quint et al., 2012).

Similarly, a recent study of Drost et al. (2015) systematically investigated the transcriptome of zebrafish, Drosophila and one plant species *Arabidopsis thaliana*. The authors introduced three permutation tests, the reductive hourglass test, the flat line test, and the reductive early conservation test, to assess then potential hourglass patterns. The authors confirmed that the oldest genes are always expressed during the mid–development stages for each animal, even for plants, thus recapitulating an hourglass pattern when considering the average evolutionary age of transcriptome across developmental time–points.

In a number of recent studies, the novel evidence for the presence of the developmental hourglass pattern has been favoured at the transcriptome level while transcriptomic and evolutionary information were combined. These studies were done for Drosophila (Domazet-Lošo and Tautz, 2010; Kalinka et al., 2010; Ninova et al., 2014), zebrafish (Domazet-Lošo and Tautz, 2010) and plants (Quint et al., 2012; Drost et al., 2015). Only one study (Irie and Kuratani, 2011) has observed the hourglass pattern during mammalian development by examining the transcript levels. However, more in depth studies are needed to confirm the existence of the hourglass pattern in animal transcriptomes.

1.8 Research Aims and Objectives

Genomes of various organisms have been sequenced completely over the past decade, but the roles and importance of a large number of genes present in those genomes are still unknown. Some genes have a key role in organismal survival and

development, whereas the functions of other genes may be useful but not critical for an organism. Genes are called essential if an organism cannot survive and develop to maturity unless these genes are functionally active. When mutated, essential genes yield lethal phenotypes at an early stage of life. Identifying all the essential genes, therefore, can reveal critical functions that are required during the development of an organism.

Human essential gene identification is very important as it answers questions about key cellular processes, tissue–specific functions and development which are crucial for sustaining life. A greater understanding of the causes behind different developmental abnormalities, birth defects and human diseases can come from identifying those genes that are essential for normal development. Based on a subset of experimentally validated essential and non–essential genes, researchers have already established the plausibility of determining human essential genes from the corresponding mouse orthologues because human and mouse have great similarity in their genomes (Hughes, 2003). The task that remains is to recognise all those genes that are absolutely indispensable for human development and survival.

Mammalian essential genes could be determined by either experimental techniques or computational approaches. Mouse knockout experiments have already evidenced useful in identifying mammalian essential genes (White et al., 2013), however, the entirety of the mouse genome has not yet been experimentally examined. Computational methodologies offer a more rapid and low–cost means to complement the laborious and time–consuming mouse knockout experiments in

predicting mammalian essential genes. Previous studies have reported a number of machine learning classifiers (computational models) that can make a reliable prediction of essential genes in worm, bacteria and yeast by learning gene/protein features underlying gene essentiality (discussed in section 1.3). These studies established that essential genes could differ from non–essential genes in a range of gene/protein properties. Though two recent studies attempted to predict essential genes in mouse (Yuan et al., 2012) and human (Yang et al., 2014) using their sequence features, they had limited success. Except these, to date, no other study has employed gene features to predict mammalian gene essentiality using computational procedures. The outcome of prior studies gives us an indication that mouse gene characteristics could also serve as effective measures to elucidate mammalian gene essentiality, which is the main motivation towards the current study. In particular, we investigated the following hypothesis:

Mammalian essential (lethal) and non–essential (viable) genes are differentiable by their features.

To test this hypothesis, we seek to exploit a number of sequence-derived and functional features of *Mus musculus* genes. In this research, a mouse gene is considered as essential or lethal if the knockout mice cause lethality within 3 days of birth when the gene is deleted. Most mouse mutants will have a time period in which lethality occurs, so using a time frame of up to postnatal day 3 allows us to capture those genes where the mutants are dying over a couple of days. All other genes with non–lethal phenotypes are considered as non–essential or viable. We

aim to address which features are most correlated with mouse essential genes. As we are concerned with determining essential and non–essential genes in the mouse by learning their properties, it characterises a supervised classification problem. Therefore, the main research question to be addressed in this study is:

Are machine learning classifiers able to classify and predict mouse essential genes using sequence and functional features of mouse genes?

To answer the above research question, the objectives of this study are set as follows:

- Collect mouse lethal and viable genes from the existing data sources.
- Investigate a wide range of gene/protein sequence and functional properties of mouse lethal and viable genes, which are easily obtainable from the existing databases and web–based tools.
- Determine if some features significantly vary between lethal and viable genes. If all features have a similar distribution between lethal and viable genes, then they would not be able to characterise gene essentiality.
- Investigate the effect of culling by removing redundant proteins from our datasets and check whether the observed features differing lethal and viable genes are over–represented or not. Redundancy in the dataset might have the potential to bias our analyses outcome.

- Develop a machine learning classifier using sequence and functional properties to predict mouse essential genes. In this case, the classification performance of different machine learning methods will be compared and the method displaying the best performance among all will be chosen.

- Apply a feature selection method to select a subset of all relevant sequence properties that can significantly predict essentiality without compromising the prediction accuracy.

- Predict essentiality for mouse genes that have known status from experimental results, but not been included in training the classifier.

In addition, we aim to explore mouse genes to study the role of gene duplication on mammalian gene essentiality. Since duplicates with similar developmental expression patterns tend to functionally compensate for each other, we will use the expression profile of duplicate genes across mouse development to examine their developmental co–expression similarities. Our hypothesis is:

Duplicate genes with similar co–expression across development are more likely to be viable, and those with divergent expression patterns tend to be lethal.

Finally, we aim to investigate whether the morphological hourglass pattern is also observed during mammalian development. For that, we will examine the evolutionary age of mouse genes expressed at the early, phylotypic and late stages across development. To our best knowledge, this is the first study where functional and sequence–based gene properties are systematically investigated and used in

developing a computational model to predict mammalian essential genes. In addition, the relationship between mouse gene essentiality and developmental co–expression, and hourglass model as applied to mouse development are studied within the context of gene essentiality. This research will ultimately aid in the identification of candidate genes for genetic diseases and different developmental abnormalities in human. We expect and believe that our study will serve as a valuable resource for the mouse genetics research community to complement the time-consuming and technically challenging mouse knockout experiments. Moreover, we hope to reveal new insights into the relationship of gene essentiality, developmental expression, and gene duplication.

1.9 Thesis Outline

This thesis is organised into six chapters. Outlines of these chapters are given below:

Chapter 1 introduces the background of this research, different machine learning methodologies and an overview of research aims and objectives.

Chapter 2 covers all relevant methods to be followed in this study. Methods for assembling mouse genes, removing redundant proteins, retrieving sequence–based and functional (protein–protein interactions, GO terms) features, analysing gene age index and statistical analysis are discussed in this chapter.

Chapter 3 reports a number of sequence and functional features that differ significantly between lethal and viable genes in mouse; therefore, it provides a strong evidence of our research hypothesis. The correlations between these significant features and gene essentiality are also explained here.

Chapter 4 gives a description of the development of a Random Forest classifier based on the significant mouse gene features stated in Chapter 2. The construction of training and test datasets is addressed here. In addition, a feature selection method, which can select a subset of most important features among all to improve the performance of the classifier, is discussed. Overall, this chapter presents mouse essential genes prediction results achieved using the machine learning classifier.

Chapter 5 deals with determining the relationships between mouse gene essentiality and gene duplication. Results of the gene co–expression analysis across 13 stages of mouse development are discussed here. Moreover, the morphological hourglass model in mouse development is addressed.

Chapter 6 synthesises the overall findings of this research and discusses their implications and limitations. This chapter concludes with remarks on future research directions.

Chapter 2

Materials and Methods

2.1 Dataset Preparation

2.1.1 Essential and Non–essential Mouse Gene Datasets

To construct the datasets for the current research, the phenotype information of knockout mice was collected from the Mouse Genome Informatics (MGI) database (Bult et al., 2008) (http://www.informatics.jax.org/phenotypes.shtml, accessed on 1 November 2013). We considered only those mouse genes that have known phenotype resulting from targeted (knockout) deletions. The phenotype of a mouse gene was marked as **essential** or **lethal** if it is associated with any lethality annotation in the MGI (including prenatal, perinatal and postnatal annotations) (Table 2.1). The term 'prenatal lethality' is a valid Mammalian Phenotype Ontology term which is defined in MGI as death of the mice anytime between fertilization and birth, whereas, 'perinatal lethality' is defined as death anytime between embryonic

Table 2.1: Mammalian Phenotype (MP) annotations that were used for defining genes as either lethal or viable

Gene Type	MP Term	MP ID
Lethal	prenatal lethality	MP:0002080
Lethal	perinatal lethality	MP:0002081
Lethal	postnatal lethality	MP:0002082
Viable	adipose tissue phenotype	MP:0005375
Viable	behaviour/neurological phenotype	MP:0005386
Viable	abnormal behaviour	MP:0004924
Viable	abnormal postnatal growth/weight/body size	MP:0002089
Viable	hearing/vestibular/ear phenotype	MP:0005377
Viable	homeostasis/metabolism phenotype	MP:0005376
Viable	abnormal immune system physiology	MP:0001790
Viable	abnormal skin morphology	MP:0002060
Viable	abnormal skin physiology	MP:0005501
Viable	abnormal touch/nociception	MP:0001968
Viable	premature aging	MP:0003786
Viable	slow ageing	MP:0011614
Viable	normal phenotype	MP:0002873
Viable	pigmentation phenotype	MP:0001186
Viable	taste/olfaction phenotype	MP:0005394
Viable	altered tumour pathology	MP:0010639
Viable	altered tumour susceptibility	MP:0002166
Viable	abnormal eye physiology	MP:0005253

day E18.5 and postnatal day 1. The phenotype term 'postnatal lethality' refers to premature death anytime between postnatal day 1 and three weeks of age. In this study, any gene that produces lethality within 3 days of birth was considered as lethal.

We used 18 phenotypic annotations to classify a single-gene knockout phenotype as **non–essential** or **viable** (Table 2.1). These are: adipose tissue phenotype, behaviour/neurological phenotype, abnormal behaviour, abnormal postnatal growth/weight/body size, hearing/vestibular/ear phenotype, abnormal immune system physiology, homeostasis/metabolism phenotype, abnormal skin morphology, abnormal skin physiology, abnormal touch/nociception, premature aging, slow

aging, normal phenotype, pigmentation phenotype, taste/olfaction phenotype, altered tumor pathology, altered tumor susceptibility and abnormal eye physiology. Since the majority of these terms refer to processes or tissues present only after birth, homozygous knockouts of these genes are evidence of a viable phenotype. We manually checked genes for viability that were linked to the "adipose tissue", "abnormal skin morphology" and "abnormal skin physiology" terms as these could be applied to embryos. Our knockout datasets contained some ambiguous entries that have been annotated as both lethal and viable in the MGI database. We manually checked phenotypes of these overlapped entries against the published literature and labelled them either as lethal or viable.

Each MGI gene symbol and identifier was further mapped to their corresponding Ensembl gene ID (http://www.ensembl.org) (Hubbard et al., 2002), UniGene expression clusters ID (http://www.ncbi.nlm.nih.gov/unigene) (Stanton et al., 2003) and UniProt protein ID (http://www.uniprot.org/uniprot/) (Apweiler et al., 2004). For some instances there were multiple UniProt protein IDs that correspond to one gene. For some of these cases, only one protein had the longest length and we included that in our dataset. For others, two or more protein IDs were found to have longest length. In these cases, to avoid bias due to annotation quality we included the longest length protein ID in our dataset that was marked as 'reviewed' in the UniProt annotations. Mouse protein sequences in FASTA format were also downloaded from UniProt for further investigation.

2.1.2 Non–redundant or Culled Datasets

A protein sequence dataset is considered as redundant if it includes a pair of proteins that are highly similar or homologous. A large number of techniques are currently available for removing redundancy from the protein dataset. These systems utilize different methods to determine redundancy and to select proteins to remove.

PISCES

One of the widely used approaches to remove data redundancy is the PISCES protein culling server (Wang and Dunbrack, 2005). This list–based approach uses a combination of structure-based alignments (Shindyalov and Bourne, 1998) and PSI–BLAST (Altschul et al., 1997) to compute sequence identities between protein pairs. PSI–BLAST is used to calculate pairwise sequence identities when protein pairs are very similar, whereas the structure-based alignment is applied to estimate sequence identities at longer evolutionary distances for which PSI–BLAST shows a sequence identity of 50% or less. PISCES generates non–redundant protein datasets by removing proteins with at least 20% sequence identity from the entire Protein Data Bank (PDB) (Berman et al., 2000) or from a list of protein sequences provided by the user. PISCES first sorts the user provided protein list according to the sequence length in descending order and removes redundancy from this list using the following steps:

1. Find x, the topmost protein in the list that is not marked as 'included' or 'excluded'.

2. Mark x as being 'included'.
3. Mark each subsequent sequence y in the list as 'excluded' if the sequence identity between x and y is higher than the chosen cut–off value.
4. Repeat step 1 to step 3 until all proteins sequences in the list are checked.
5. Return the non–redundant list c with all proteins being flagged as 'included'.

The non–redundant dataset generated by PISCES is biased towards keeping longer length proteins since proteins sequences are sorted from largest to shortest by their length.

LEAF

A recent alternative to PISCES is a graph–based approach called Leaf (Bull et al., 2013) which relies on representing the similarity relationships between protein pairs using an undirected graph. As an example, let $G = (V, E)$ be an undirected graph representing the redundant protein dataset. In the graph G, nodes $V = \{1, 2, ..., n\}$ represent proteins and edges $E = \{(V_i, V_j)|i, j \in V\}$ represent connections between protein V_i and V_j those have the sequence identity above a chosen cut–off. The goal of the Leaf algorithm is to maximise the size of the non–redundant dataset. This is done by approximating a largest possible (maximal) independent set within the graph G, where an independent set $I \subseteq V$ constitutes a list of nodes provided that no two nodes are connected in the graph G. To generate a non–redundant dataset, the Leaf algorithm repeatedly looks for cliques of different sizes in graph G. A clique $C \subseteq V$ of size n in G contains a total of n nodes all adjacent to each other and at least one of these nodes does not have any edge connected to

any other nodes outside of C. The algorithm starts with searching for a clique of size 2 ($n = 2$) and the size is incremented by 1 in the subsequent iterations. This process of searching cliques with increased size is continued until a clique is found satisfying a certain threshold value or if no possible clique is found in G. The threshold value is set to be the degree of the node with the highest number of connections in G. Each time a clique is found, one node from the clique is chosen arbitrarily and is added in the independent set. All other nodes of the clique are then removed from G. If no clique is found, the node with the highest degree is removed from G and the process of searching for clique continues with the newly formed graph G_{new}. According to Bull et al. (2013), the resultant non–redundant datasets from Leaf are $\sim$10% bigger than those generated by PISCES without compromising the quality.

In this study, redundancy was removed from our original lethal and viable datasets by submitting the lethal and viable protein sequences in FASTA format to both Leaf and PISCES that were locally installed. We used these programs to get four sets of non–redundant lethal and viable proteins with protein pairs showing the maximum sequence similarities of 20%, 40%, 60% and 80% respectively. Protein sequences with $< 20\%$ identity are structurally very different implying functional differences (Rost, 1999; Wood and Pearson, 1999) and therefore, we did not generate non–redundant datasets by removing proteins with $< 20\%$ sequence identities.

2.1.3 Singletons and Duplicates

Mouse lethal and viable genes were labelled as singletons and duplicates from prior annotations (Makino and McLysaght, 2010), from Ensemble (release 75) gene trees of mouse gene families (Vilella et al., 2009) and from our own protein sequence similarity measure. Makino and McLysaght (2010) listed a set of 9,059 duplicated gene pairs in the human genome. We retrieved mouse orthologues of these human genes using the Ensembl BioMart data–mining tool (http://www.ensembl.org/biomart/martview/) (Smedley et al., 2009) with the Ensembl release 75 dataset of the *Homo sapiens* genes (GRCh37.p13). Comparing our lethal and viable mouse genes with these human orthologues of mouse duplicates, we made three groups of duplicated gene pairs: lethal–lethal, lethal–viable and viable–viable duplicates.

Furthermore, we downloaded BLAST+ software package from the NCBI website and performed an all–against–all Blast search (Altschul et al., 1990) on our local machine to detect mouse duplicates within our own datasets. We set up three target databases comprising lethal, viable and lethal+viable (combined) protein sequences to conduct the Blastp search. In accordance with prior studies (Chen et al., 2012b; Makino and McLysaght, 2010), we inferred a mouse gene as a duplicate if it has hits to other mouse genes within our datasets with E–values $< 10^{-7}$. We considered the best hit to be the closest paralogue of a duplicate gene. If a gene had no Blastp hit within an E–value $< 10^{-7}$ within a dataset, it was classified as singleton. A mouse duplicate gene was further labelled as either a small–scale

duplicate (SSD) or a whole–genome duplicate (WGD) gene. We labelled a mouse duplicate gene in our dataset as a whole–genome duplicate if its human orthologue is found within the 9,059 human duplicate pairs listed in Makino and McLysaght (2010). All these human duplicate gene pairs are duplicates generated by WGD mechanism. The rest of the mouse duplicate genes in our datasets were labelled as small–scale duplicates.

2.1.4 All Mouse Genes Dataset

We used the MouseMine data warehouse (http://www.mousemine.org/mousemine) (Motenko et al., 2015) to further compile a dataset comprising all mouse genes. The MouseMine system integrates a major portion of mouse data from the MGI database. This dataset also include all lethal and viable genes that we collected (section 2.1.1). Mouse genes that were not labelled as either lethal or viable were categorized as genes with unknown essentiality status. This dataset was used in investigating the hourglass pattern in mouse (section 5.6).

2.2 Features Collection

We collected a number of gene and protein sequence based features to distinguish lethal and viable phenotypes. Functional features like protein–protein interactions (PPI) and gene ontology (GO) annotations were also considered as quantifiable parameters to check whether they could offer differences between lethal and viable

TABLE 2.2: List of sequence and functional features collected and various bioinformatics tools used for their retrieval

Features	Bioinformatics Tools
Genomic features: gene length, % of GC content, number of transcripts, number of exons, length of exon and intron	Ensembl BioMart (Smedley et al., 2009)
Gene expression	UniGene (Stanton et al., 2003)
Evolutionary age	Ensembl gene trees (Vilella et al., 2009)
Protein sequence features: protein length, molecular weight, protein charge, isoelectric point, amino acid composition	Pepstats (Rice et al., 2000)
Enzyme class	UniProt (Apweiler et al., 2004)
Keywords: Glycoprotien, Phosphoprotein, Acetylation, Transcription	UniProt
Transmembrane domains	UniProt
Subcellular locations	UniProt, WoLF PSORT (Horton et al., 2007)
Signal peptide	SignalP 4.1 (Petersen et al., 2011), UniProt
Protein–protein interaction (PPI) network features	I2D database (version 2.3) (Brown and Jurisica, 2007), Cytoscape (version 3.1.1) (Shannon et al., 2003)
Gene Ontology terms: biological process, cellular component, molecular function	DAVID (version 6.8) (Huang et al., 2007)
Protein Domain	DAVID (version 6.8)

datasets. Analysis of these features ultimately helps us to highlight the characteristics of essential genes and also reveal novel features for further study. Table 2.2 summarizes the sequence and functional attributes collected and the corresponding tools that were used to extract them. The following subsections describe how different features were obtained using existing tools and web services.

2.2.1 Genomic Properties

2.2.1.1 Gene sequence properties

Features including gene length (in base pair), % of GC content, number of transcripts, number of exons, lengths of exons and introns were retrieved from the

Ensembl release 75 database of Mus musculus genes, using the Ensembl BioMart (Smedley et al., 2009) data mining tool. We used Ensembl gene IDs to get these features. For genes with multiple transcripts, the longest length transcript was assessed. A gene's exon number and the total exon length were calculated considering its longest transcript. The intron length of a gene was calculated by subtracting its total exon length from the corresponding gene length.

2.2.1.2 Gene expression

Raw expression data of mouse lethal and viable genes were obtained from the NCBI UniGene database (Stanton et al., 2003) as expressed sequence tag (EST) clusters using UniGene IDs. We retrieved EST clusters from 13 developmental stages: oocyte, unfertilized ovum, zygote, cleavage, morula, blastocyst, egg cylinder, gastrula, organogenesis, fetus, neonate, juvenile and adult. Since the total number of ESTs for a particular gene varies greatly between different developmental stages, we corrected the raw data to get gene expression in the form of transcripts per million (TPM). Equation 2.1 was used to estimate a TPM for the i^{th} gene at j^{th} developmental stage.

$$TPM_i^j = (Number\ of\ ESTs\ for\ i^{th}\ gene\ /\ Total\ ESTs\ in\ j^{th}\ stage) \times 10^6 \quad (2.1)$$

TPMs were also transformed to their corresponding log values using Equation 2.2 to measure co–expression between every gene pair.

$$L_{TPM_i^j} = log_e(TPM_i^j + 1) \quad (2.2)$$

TPMs were also normalised within the range (0, 1) using Equation 2.3 dividing each TPM data by the maximum TPM value. Since the value of these normalised TPMs were too small, we further multiplied them by 10.

$$N_{TPM_i^j} = (TPM_i^j + 1)/max(TPM) \tag{2.3}$$

We used the Euclidean and the Manhattan distance methods to calculate numerical scores representing gene co–expression. This numerical distance value is used to compare gene expression (L_{TPM}) between every gene pair during development. If $\mathbf{a} = (a_1, a_2, ..., a_{13})$ and $\mathbf{b} = (b_1, b_2, ..., b_{13})$ are two mouse genes having expression across 13 developmental stages, then the Euclidean and Manhattan distance between them are calculated by Equation 2.4 and 2.5, respectively. Small scores (distances) reflect higher co–expression between genes. We considered both log and normalised TPM data to calculate the Euclidean distance.

$$EucDis(a, b) = \sqrt{\sum_{i=1}^{13}(a_i - b_i)^2} \tag{2.4}$$

$$ManDis(a, b) = \sum_{i=1}^{13}|(a_i - b_i)| \tag{2.5}$$

2.2.1.3 Evolutionary age

Evolutionary ages of mouse protein coding genes were determined by analysing the Ensemble (release 75) gene trees of mouse gene families (Vilella et al., 2009). These gene trees represent the evolutionary lineage of genes with their common ancestors. Ensembl runs a orthology and paralogy gene prediction pipeline that

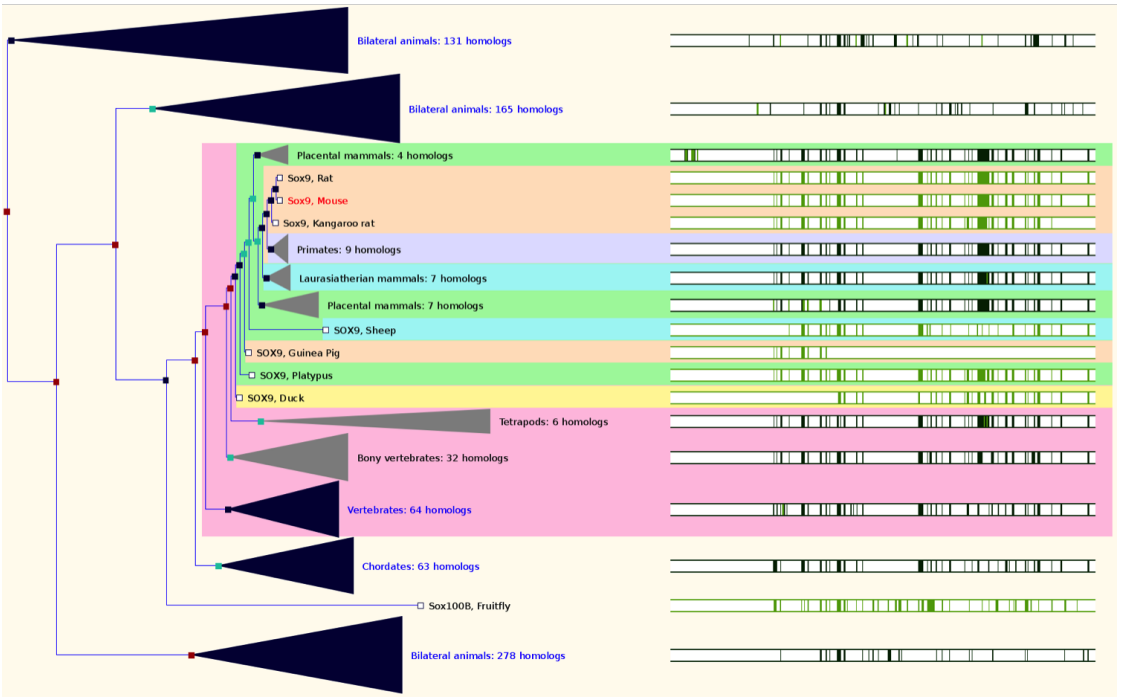


FIGURE 2.1: The Ensembl gene tree for mouse gene Sox9 (highlighted in red). Red squares symbolise duplication (paralogues) nodes, whereas blue squares symbolize speciation (orthologues) nodes.

uses the TreeBeST method from the TreeFam methodology (Li et al., 2006) to generate rooted phylogenetic trees. This pipeline merges tree topologies with the corresponding species trees inferred from the NCBI taxonomy and generates Ensembl genes trees with the tree internal nodes being annotated for duplication or speciation events. Figure 2.1 shows a gene tree generated by Ensembl for the mouse gene **Sox9**.

A program was developed and used to determine gene evolutionary ages from these gene trees. We assigned two evolutionary ages to a mouse duplicate gene of our datasets: the age of the most recent duplication (MRD) event and the age of the evolutionarily most distantly related species, *i.e.*, the age of the duplicate common ancestor (DCA) that has an identified homologue to that gene. For

singletons, we considered the age their single common ancestor (SCA).

2.2.2 Protein Sequence Properties

2.2.2.1 Simple sequence properties

We retrieved the length of our lethal and viable protein sequences by querying the UniProtKB database with their UniProt IDs. A script in Python was developed to compute the percentage frequencies of each of the 20 amino acid residues within protein sequences. We at first counted the number of occurrences of each amino acid residues in a protein sequence and further divided this count by the corresponding sequence length to obtain the frequency with which the amino acid occurs in the sequence.

Pepstats (http://emboss.bioinformatics.nl/cgibin/emboss/pepstats) is a EMBOSS suite program (Rice et al., 2000) which outputs a report comprising statistics of a number of properties about a FASTA formatted protein sequence. These attributes include molecular weight, number of residues, charge, isoelectric point, and amino acid composition. This program groups amino acids into nine categories: Tiny (A, C, G, S and T), Small (A, B, C, D, G, N, P, S, T and V), Aliphatic (I, L and V), Aromatic (F, H, W and Y), Non–polar (A, C, F, G, I, L, M, P, V, W and Y), Polar (D, E, H, K, N, Q, R, S, T and Z), Charged (B, D, E, H, K, R and Z), Basic (H, K and R) and Acidic (B, D, E, Z). Pepstats was used to evaluate these sequence properties for lethal and viable protein sequences.

The program was run with the default parameters setting. A Python script was written to extract features values from the output file generated by Pepstats.

2.2.2.2 Enzyme class and post–translational modifications

The Enzyme Commission (EC) numbering scheme is the most widely used method for the numerical classification of enzymes. According to this scheme, enzymes are categorized into six main classes based on enzyme–catalysed reactions.

- Class 1: **Oxidoreductase** – These enzymes catalyse oxidation or reduction reactions
- Class 2: **Transferase** – These enzymes catalyse transfer of a functional group, such as methyl, acyl and others, from one substance to another.
- Class 3: **Hydrolase** – These enzymes catalyse bond cleavage using water (hydrolysis)
- Class 4: **Lyase** – These enzymes catalyse splitting bonds to be cleaved
- Class 5: **Isomerase** – These enzymes catalyse intramolecule change
- Class 6: **Ligase** – These enzymes catalyse the formation of new bonds

Each enzyme number is represented by a numerical format: *a.b.c.d.* Here, '*a*' refers to any of the six classes the enzyme belongs; '*b*' and '*c*' refers to the subclass and sub–subclass respectively; and '*d*' refers to the rank of the enzyme in its sub-subclass. In this study, primary EC numbers of mouse proteins were obtained from the definition lines (DE) of UniProtKB annotations by submitting UniProt IDs.

UniProt entries are also labelled with different keywords that are classified into ten categories. These categories of keywords include domain, post–translational modification (PTM), biological process, coding sequence diversity, ligand, molecular function, cellular component, developmental stage, disease and technical terms. Post–translational modifications are the covalent and enzymatic modifications of the protein after the completion of its translation which resultant in mature protein products. A protein undergoes a PTM when a functional group (hydroxyl, phosphate, alkyl and others) is covalently added to it. Phosphorylation and glycosylation are the two most important post-translational modifications. In Uniprot, 'glycoprotein' and 'phosphoprotein' are the synonyms of glycosylation and phosphorylation processes. Within a phosphoprotein, phosphorylation occurs mainly on serine (S), threonine (T) or tyrosine (Y).

Glycosylation is a major post-translational modification in which glycans covalently attach to proteins. Preassembled gylcans can attach to the nitrogen of asparagine sidechain (N-linked glycoprotein) or to the hydroxyl oxygen on the sidechains of serine or threonine (O-linked glycoprotein).

Acetylation is another common PTM in which proteins are post-transnationally modified while an acetyl group is added a primary amine. For most of the proteins, acetylation occurs on lysine residues.

Three post–translational modification (PTM) keywords 'Glycoprotein', 'Phosphoprotein' and 'Acetylation' were collected from the UniProtKB database for each protein of our datasets. We included the N–linked glycoprotein only due to

the fact that UniProt annotates glycoproteins with N-glycosylation sites, not with O–glycosylation sites.

We also collected information about the keyword 'Transcription' from the UniProtKB database. It is a keyword in the biological process category representing proteins involved in regulating the process of transcription.

2.2.2.3 Signal peptides

Protein signal peptides were predicted using the SignalP program version 4.1 (http://www.cbs.dtu.dk/services/SignaP/) (Petersen et al., 2011). This program uses artificial neural network (ANN) and hidden Markov model (HMM) algorithms to predict the amino acid composition and the cleavage site position of the signal peptide. A script in Python was written to extract the HMM probabilities generated by SignalP which is considered as the measure for signal peptide prediction.

2.2.2.4 Transmembrane domain

We extracted the total number of transmembrane domains in each mouse protein by querying the UniProtKB database. Transmembrane helices are annotated in the UniProt feature table line (FT) as TRANSMEM. UniProt also outputs the information about the transmembrane domain locations in a protein sequence.

2.2.2.5 Subcellular location

Protein subcellular locations were predicted from sequence data using the WoLF PSORT program (http://wolfpsort.org/) (Horton et al., 2007). This program was chosen here as it can make prediction on any protein sequence. WoLF PSORT predicts subcellular locations on the basis of known sorting signals, functional motifs and sequence features like amino acid composition. It outputs a report covering predicted locations with different confidence levels. We submitted our FASTA formatted lethal and viable protein sequences to WoLF PSORT and extracted the confidence score for potential subcellular locations from the output report it generated. We found prediction scores for six subcellular locations: nucleus, cytosol, plasma membrane, mitochondria, Golgi apparatus, peroxisome, and extracellular. We assigned a score of zero to a subcellular location if no prediction is made for that. We further collected information about all these six subcellular locations from the UniProtKB database. This feature is annotated as SUBCELLULAR LOCATION in the UniProt data file and is found in the comment lines (CC). Value of a subcellular location was set to 1 if found; otherwise, it was set to 0.

2.2.3 Gene Ontology Terms and Protein Domains

GO terms were obtained by using the 'Functional Annotation' tool of the web based application DAVID version 6.8 (https://david.ncifcrf.gov/home.jsp) (Huang et al., 2007). It integrates gene functional annotations with intuitive graphical displays to facilitate biological interpretations of any list of genes encoded by

human, rat, mouse, or fly genomes. This program systematically associates a query gene list to their corresponding GO terms and highlights only the most pertinent terms among all along with their statistics. We downloaded the output files that DAVID generated for our lethal and viable mouse gene lists and extracted all possible GO terms for which the statistical test supported in DAVID has a p–value < 0.05. The Pfam domain (Bateman et al., 2004) information for lethal and viable proteins was also obtained from the DAVID functional annotation tool.

2.2.4 Protein–Protein Interactions

Mouse PPI data was downloaded from the Interologous Interaction Database (I2D) version 2.3 (Brown and Jurisica, 2007) which is an integrated repository of known, experimental and predicted PPIs for human, mouse, rat, fly, yeast and worm genomes. To obtain high quality PPI data, we analysed all known and predicted mouse PPIs. From these interactions two PPI networks were generated – *Known*(**K**) and *Known–Predicted*(**KP**). The network **K** contained experimentally verified mouse PPIs that I2D extracted from known PPI databases including BioGrid_Mouse, I2D–c_Fiona_MOUSE, BIND_Mouse, Chen_PiwiScreen, DIP_Mouse, I2D–c_Mouse, IntAct_Mouse, INNATEDB_Mouse, KIM_MYC, MGI, MINT_Mouse, WangEScmplx, WangEScmplxlow, and WangEScoIP. The network **KP** contained all known interactions as well as predicted mouse PPIs based on orthologous interactions in rat, human, yeast, worm, and fly.

We further divided **K** and **KP** networks as lethal–**K**, viable–**K**, lethal–**KP**

and viable–**KP** networks to analyse lethal and viable proteins separately. Lethal networks covered all interactions where one (or both) of the interacting partners is a lethal protein, whereas viable networks included all interactions linked to viable proteins. All self-interactions and duplicate interactions were removed from these networks. Cytoscape (version 3.1.1) (Shannon et al., 2003) was further used to visualize and analyse these PPI networks as a graph. The 'network analyser' plugin of Cytoscape was used to determine network properties including degree, the length of average shortest path (ASP), betweenness centrality, clustering coefficient, and closeness centrality. We further determined four other network properties including BottleNeck (BN), Edge Percolation Component (EPC), Maximum Neighbourhood Component (MNC) and Density of Maximum Neighbourhood Component (DMNC) by using a web-based service called Hub Object Analyser (Hubba) (http://hub.iis.sinica.edu.tw/Hubba/) (Lin et al., 2008). This system deciphers and visualizes hubs from the user-provided PPI networks. Query proteins are ranked in Hubba based on their topological features. Hubba also generates a subgraph for the top n ranked ($n \leq 100$) hub along with their identifier.

PPI networks are usually characterized as undirected graphs (Figure 2.2). As an example, let $G = (V, E)$ be an undirected graph representing a PPI network. In the graph G, nodes V represent proteins and edges $E = \{(a, b)|a, b \in V\}$ correspond to observed interactions between protein a and protein b. In the following paragraphs the definitions of each topological feature are given:

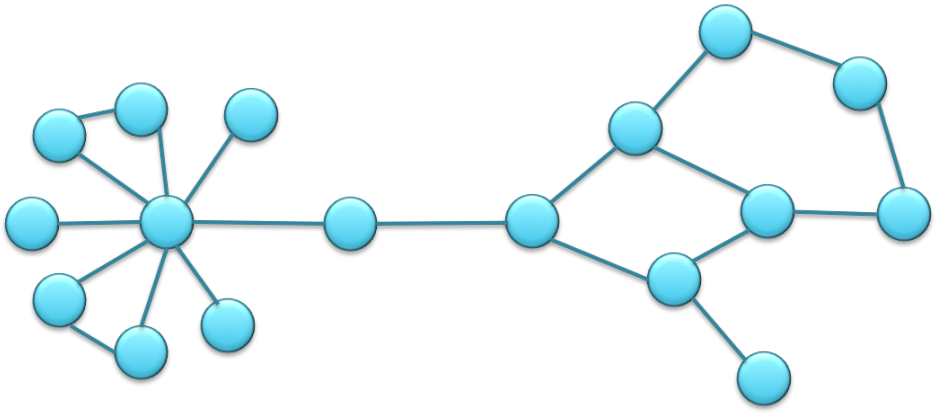


FIGURE 2.2: A simple graph model of a protein–protein interaction network. Here, the nodes represent proteins and edges represent the interaction between two proteins.

Degree

The most elementary property of a protein a is its degree or connectivity, which is the number of observed interactions a has to the other proteins in the network.

Average shortest path length (ASP)

The shortest path measures the path with the minimum number of edges between proteins a and b. The average shortest path (ASP) length therefore refers to the average over all shortest path length between all protein pairs.

Betweenness centrality (BC)

The betweenness centrality (BC) of a protein node a corresponds to the ratio of shortest paths passing through a (Brandes, 2001; Joy et al., 2005) and is computed as follows:

$$BC(a) = \sum_{b \neq c \neq a \in V} \frac{\sigma_{bc}(a)}{\sigma_{bc}} \tag{2.6}$$

Here, σ_{bc} denotes the number of shortest paths between proteins b and c; and $\sigma_{bc}(a)$ denotes the number of shortest paths between b and c that goes through protein node a.

Clustering coefficient (CCo)

The clustering coefficient (CCo) of protein a (Equation 2.7) measures the ratio of the number of connections between all nodes within the neighbourhood of a to the maximum number connections that could possibly present between them (Barabasi and Oltvai, 2004).

$$CCo(a) = \frac{2e_{bc}}{k_a(k_a - 1)} \tag{2.7}$$

Here, e_{bc} denotes the number of connections between all neighbours b and c of a; and k_a denotes the degree of a.

Closeness centrality (CC)

The closeness centrality (CC) of the protein a corresponds to the reciprocal of the sum of average shortest path length between a and all the other nodes within the network (Newman, 2005) (Equation 2.8). It measures how close a protein node is to all the other nodes in the PPI network.

$$CCo(a) = \frac{1}{\sum_{b \neq a} d(a, b)} \tag{2.8}$$

Here, $d(a, b)$ is the length of the average shortest path between proteins a and b.

Betweenness centrality, clustering coefficient and closeness centrality of each protein node are represented by a value between 0 and 1 where an isolated protein node has a value of 0 for these properties.

BottleNeck (BN)

Let, T_r be the shortest path tree derived from G considering protein node $r \in V$ as the root node. Protein $b \in V$ is a bottleneck node if at least $n/4$ nodes have its shortest path to r through b in T_r. The BottleNeck (BN) score of the protein node b is defined to be the number of nodes r for which b is a bottleneck node in T_r (Pržulj et al., 2004).

Edge Percolation Component (EPC)

Let G' is a graph which is constructed n times from G by randomly removing a subset of edges. It is possible that proteins a and b are connected in G but not in G'. The Edge Percolation Component (EPC) score (Chin and Samanta, 2003) of the protein node a is computed using the following equation:

$$EPC(a) = \sum_{b \neq a \in V} \frac{\sum_{for\ each\ G'} \begin{cases} a\ and\ b\ are\ connected\ in\ G' & 1 \\ else & 0 \end{cases}}{n} \tag{2.9}$$

Maximum Neighbourhood Component (MNC)

The Maximum Neighbourhood Component (MNC) of a protein a refers to the size of the maximum connected component of the subnetwork induced by the

neighbourhood of a (Lin et al., 2008).

Density of Maximum Neighbourhood Component (DMNC)

The Density of Maximum Neighbourhood Component (DMNC) for the protein a is calculated using the following equation:

$$DMNC(a) = \frac{E_M}{N^e} \tag{2.10}$$

Here, E_M denotes the number of edges and N denotes the number of protein nodes of $MNC(a)$; e is a constant which is equal to 1.7.

2.3 Calculation of Transcriptional Age Index

The transcriptional age index (TAI) is a weighted mean of evolutionary ages for a mouse developmental stage. Following the definition of TAI in (Domazet-Lošo and Tautz, 2010), Equation 2.11 was used to calculate TAI at the j^{th} mouse developmental stage.

$$TAI_j = \frac{\sum_{i=1}^{n}(age_i \times TPM_i)}{\sum_{i=1}^{n} TPM_i} \tag{2.11}$$

Here, n represents the total number of mouse genes expressed at the j^{th} stage. For the gene i in stage j, age_i represents either the MRD or the DCA age and TPM_i is the gene expression value.

Equation 2.11 can be alternatively written as:

$$\begin{aligned} TAI_j &= age_1 \times \frac{TPM_1}{TPM_1 + TPM_2 + ... + TPM_n} \\ &+ age_2 \times \frac{TPM_2}{TPM_1 + TPM_2 + ... + TPM_n} \\ &+ ... + age_n \times \frac{TPM_n}{TPM_1 + TPM_2 + ... + TPM_n} \\ &= (age_1 \times exp_1) + (age_2 \times exp_2) + ... + (age_n \times exp_n) \\ &= \sum_{i=1}^{n} (age_i \times exp_i) \end{aligned} \tag{2.12}$$

Here, $(TPM_1 + TPM_2 + ... + TPM_n)$ represents the total gene expression, whereas the ratio $TPM_i/(TPM_1 + TPM_2 + ... + TPM_n)$ as symbolised by exp_i denotes the expression frequency of gene i in the total gene expression at the stage j.

2.4 Statistical Analysis

To infer the relationship between gene features and essentiality, the first step is to compare the distribution of these features between lethal and viable datasets. Properties in both datasets were investigated in two ways. At first all features were tested for normality using a one sample Kolmogorov–Smirnov Test (K–S test) (Massey Jr, 1951). This non–parametric statistical method compares sample values with a standard normal distribution. If this test gives the p–value (statistical significance value) < 0.05, it indicates the distribution of a feature is not significantly different from the normal distribution. If a sequence property showed a normal distribution, a two–sample t–test with unequal variance analysis was

further applied to assess the feature–essentiality relationship. The t–test demonstrates whether the mean value of one observation (lethal genes) is significantly different from the mean value of the other observation (viable genes). The t–test calculates a t–statistic, which is compared with a standard table of t–values at 95% confidence level.

The statistical significance of each feature was tested using the two–tailed non–parametric Mann–Whitney U test (Mann and Whitney, 1947) when the data distribution was not normal. It shows whether the median of lethal dataset is significantly different from the median of viable datasets. This test assesses the null hypothesis that two independent samples of observations have identical continuous distributions with equal medians. This test calculates the probability of observing the value of a statistic called U to accept or reject the null hypothesis. A p–value < 0.05 rejects the null hypothesis. We also used the non–parametric Kruskal–Wallis method (Kruskal and Wallis, 1952) to test the null hypothesis that more than two independent variables come from identical continuous distributions with equal medians.

We used the Chi–squared (χ^2) test for features like GO annotations to check whether the frequencies of a GO term in lethal and viable gene differ from each other. This test compares the observed frequency of a particular feature with the frequency that would be expected. Furthermore, we applied the Bonferroni correction (Dunn, 1961) to calculate corrected p–values. It lowers the chances of getting false positive results that are derived from multiple pairwise comparisons

performed on a single dataset. All these statistical tests were carried out using the statistics software package SPSS (Norusis, 1985) version 20.

The ANOVA test with Bonferroni correction was applied to evaluate statistical significant difference in TAI and mean age values between different mouse developmental stages. We applied a bootstrap approach (Efron, 1981) to establish the confidence for TAI values at each mouse developmental stage. A population of 1,000 resamples was constructed at each stage by randomly sampling (Vitter, 1985) the MRD and DCA age values of expressed genes. At each stage, we computed 1,000 TAI values from corresponding resamples and estimated the standard deviation of them to measure the degree to which these TAI values differ from the TAI value of original sample.

2.5 Machine Learning

2.5.1 The Mammalian Essential Gene Prediction Classifier

In this study, the mammalian essential gene prediction problem was formulated as a supervised binary classification problem. Given a mouse gene p, we intended to predict the corresponding class $y \in \{lethal, viable\}$. We used Weka (version 3.6), a publicly available Java based machine learning software (Hall et al., 2009) to implement the predictive classifier. Weka offers a collection of machine learning algorithms and visualisation tools for data mining and predictive modelling tasks. We used the Naive Bayes, J48 decision tree, Support Vector Machine (SVM) and

Random Forest methods in Weka as classifiers. Classifiers were trained on fixed number of mouse genes labelled as lethal or viable, each consisting of m features. Each classifier generates a probability score representing gene essentiality. Separate test datasets were also created with gene features and known class labels that have not been included in the training dataset. Calculating the proportion of correctly predicted genes in these test datasets further validated the performance of classifiers.

We chose Random Forest in Weka as our model of choice based on its prediction accuracy. Random Forest (Breiman, 2001) is an ensemble classifier comprising multiple decision tree models (section 1.5.6). These decision trees are constructed during the learning process with the training data. Random Forests mainly rely upon two parameters to control their growth: $numTrees$, the number of decision trees in the forest to be built and $numFeatures$, the number of random subsets of features to assess at each tree node. Let $numTrees = T$ and $numFeatures = m$. Each of the T decision trees is constructed in a top down manner starting with a root node by selecting a set of N samples of size n at random with replacement from the training dataset and selecting the most significant features of these samples as the tree nodes. At each node a, m number of features is selected at random from n features to grow the tree and the most significant feature that provides the best binary split on that node is selected among all according to a objective function. This significant feature splits node a into left (L) and right (R) child nodes with a set of $N_L \subseteq N$ and $N_R \subseteq N$ samples in L and R nodes, respectively. This

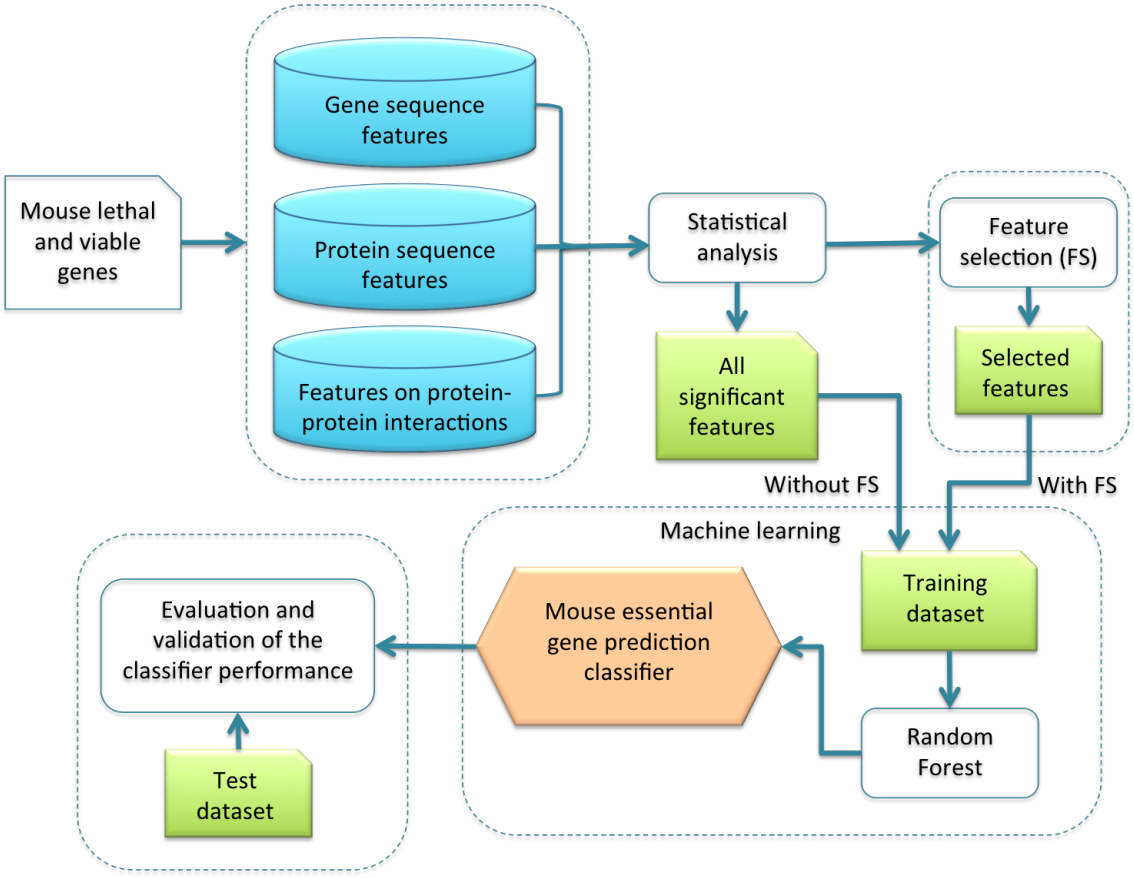


FIGURE 2.3: The flowchart for predicting mammalian essential genes from sequence and functional properties of mouse genes using a Random Forest classification model.

process continues at each node until the decision tree cannot grow further.

To classify a new gene, the features of the gene are tested with each of the decision trees present in the Random Forest. Each tree gives a classification score or "vote" and the class with the most votes is selected as the class to which the gene belongs. We set different values for $numTrees$ and $numFeatures$ parameters to obtain the best–fit Random Forest classification model for our mouse genes. Figure 2.3 shows the overall workflow of predicting mammalian essential genes using Random Forest classifier.

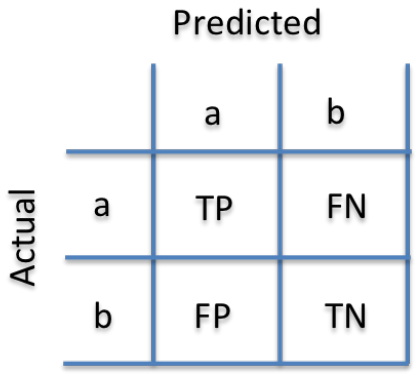


FIGURE 2.4: Confusion matrix in Weka, where a refers to positive class and b refers to negative class.

2.5.2 Performance Measures

The performance of our classifier was evaluated by a 10–fold cross-validation analysis, where each training dataset was randomly partitioned into 10 equal parts with 9 parts being used for model training (learning) and the remaining part being used for testing (validation). We used the cross–validation method to limit overfitting of the classifier. A classifier overfits if its prediction accuracy is high on the training dataset but is poor on the test dataset.

A prediction in a classification problem can either be a true positive (TP) or false positive (FP), or true negative (TN) or false negative (FN). The performance of the Random Forest classifier relied upon the total number of lethal genes predicted correctly (TP), lethal genes predicted incorrectly (FN), viable genes predicted correctly (TN), viable genes predicted incorrectly (FP). A confusion matrix predominantly represents it (Figure 2.4). Model performance was then evaluated by the true positive rate (recall or sensitivity) – TPR, false positive rate – FPR,

precision, F–measure, and the overall classification accuracy, as defined by the following equations:

$$TPR(Recall\ or\ Sensitivity) = \frac{TP}{TP + FN} \tag{2.13}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2.14}$$

$$FPR = 1 - Specificity = \frac{FP}{FP + TN} \tag{2.15}$$

$$Precision = \frac{TP}{TP + FP} \tag{2.16}$$

$$F - measure = \frac{2 \times TPR \times Precision}{TPR + Precision} \tag{2.17}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.18}$$

The high classification accuracy (the proportion of the correctly predicted instances) of a binary classifier is often a misleading performance measure for imbalanced dataset with the number of instances in one class being larger than the other class. A high accuracy could be achieved by predicting all instances belonging to the majority class. Therefore, despite this classifier potentially showing high accuracy, its performance is highly deceiving due to its disregard for the instances belonging to the minority class. TPR, FPR, Precision and F-measure give unbiased and accurate measures of the classifier performance.

To further demonstrate the performance of our classifier, we generated the receiver operating characteristics (ROC) curves by plotting the TPRs against the

FPRs at various threshold settings. These present the probability of predicting true positives as a function of the probability of predicting false positives (Huang and Ling, 2005). The area under curve (AUC) of these ROC curves offers a good estimate of the overall prediction performance of the classifier. The AUC measures how well a binary classifier could accurately classify two groups. An AUC of 1 represents a perfect prediction; an AUC of 0.5 represents a random guess.

2.5.3 Feature Selection

A feature can be irrelevant, strongly relevant (removal of this reduces the overall prediction accuracy), or weakly relevant (not sufficient alone for prediction). Feature selection, therefore, is a very important stage for the classification problem to deal with datasets comprised of a large number features and to select informative features. A number of feature selection algorithms are currently available whose goal is to choose a much smaller set of features relevant for classification from the larger datasets. We used the Information Gain feature selection filter (*InfoGainAttributeEval*) in Weka, which select a subset of features from the pool of all features (Han et al., 2011). This method estimates the worth (rank) of a feature by measuring its information gain with respect to a classification target and outputs only the top ranked features on the basis of a predetermined threshold. The information gain of a feature measures the amount of information obtained for a class prediction by knowing the presence or absence of the feature in the dataset. We further trained and test our classifier with these selected features (Figure 2.3).

2.5.4 Discretisation

Discretisation is a process of transforming the numeric value of a continuous attribute into nominal values or intervals. Many studies showed that some classifying algorithms work better when the continuous features are discretised (Liu et al., 2002). A prior study reported that discretisation makes learning faster and more accurate (Dougherty et al., 1995). Accordingly, we used a supervised discretise attribute filter implemented in Weka to discretise the continuous attributes of mouse genes within the training and test datasets. We further developed our Random Forest classifier using these discretised properties.

Chapter 3

Analysis of Mouse Essential Genes based on Sequence and Functional Features

3.1 Introduction

Essential genes are those whose presence is imperative for organism's survival. However, the set of genes that are absolutely vital to sustain life are still unknown for most organisms (Juhas et al., 2011). In mammals, knowledge of essential genes is required to understand development, maintenance of major cellular processes and tissue–specific functions that are crucial for life. Mammalian essential genes could be identified using existing experimental techniques, which include single gene knockouts (Crawley, 1999; Giaever et al., 2002; Kobayashi et al., 2003), conditional knockouts (Liu et al., 2000; Roemer et al., 2003), RNA interference (Cullen and Arndt, 2005; Kamath et al., 2003), and transposon mutagenesis (Gallagher

et al., 2007). Most of these conventional experimental techniques are time consuming and expensive. Computational prediction, which relies on sequence properties of a gene to evaluate essentiality, offers a fast and low cost alternative. In this research, we hypothesised that mammalian essential (lethal) and non–essential (viable) genes are distinguishable by various attributes.

We explored a wide range of sequence and functional features of mouse genes in order to characterise lethal and viable genes in mammals. Some of these features were previously found to be associated with essentiality in *E. coli* (Gustafson et al., 2006; Deng et al., 2010) and *S. cerevisiae* (Jeong et al., 2003; Seringhaus et al., 2006; Zhang et al., 2013). To be able to predict gene essentiality from these sequence and functional properties, we must first confirm that some properties are significantly different between lethal and viable groups. If all features are same within the lethal and viable groups, then they could not be predictive of essentiality. There should, therefore, be a number of features that differ between lethal and viable genes. A feature might still be useful for prediction in conjunction with other significant features even if it is not significantly different between the lethal and viable groups. In this chapter, we report a number of gene and protein features that vary significantly between lethal and viable genes in mouse. An explanation of the relationship between these highly correlated features and gene essentiality is also discussed here.

3.2 Datasets

The Mouse Genome Informatics (MGI) database (Bult et al., 2008) incorporates published gene data on mouse knockout phenotypes. We collected a total of 1,271 lethal and 4,378 viable mouse genes from MGI (accessed on 1 November, 2013) based on phenotype annotations of knockout mice. A total of 1,335 genes had both 'lethal' and 'viable' annotations in the MGI database which we considered ambiguous entries. To ensure that gene classifications were in agreement with our criteria of lethality and viability, we manually studied each of these ambiguous genes with the use of published experimental evidence. We further manually checked each gene to ensure that our datasets contained only protein–coding genes. This gives a total of 1,301 and 3,451 lethal and viable mouse genes, respectively.

However, the proteins encoded by these lethal and viable genes share significant levels of sequence identity. The presence of multiple similar protein sequences is a barrier in using a dataset effectively as it increases the size of the dataset; also it could potentially create a bias towards any conclusions drawn from using the dataset. We, therefore, used Leaf (Bull et al., 2013) and PISCES (Wang and Dunbrack, 2005) (discussed in Chapter 2) to remove redundant proteins from our datasets. Leaf uses a recent version of PSI–BLAST (version 2.2.25) to calculate pair-wise sequence identities between all protein pairs. On the other hand, an older version of PSI–BLAST (Altschul et al., 1997) method (version 2.2.10) is used in PISCES. This version of BLAST in PISCES does not have the same level of accuracy in finding sequence similarities as compared to Leaf. Thus, non-redundant

TABLE 3.1: Numbers of lethal and viable mouse genes in culled datasets

Sequence Identity	Lethal	Viable
20%	479	1017
40%	961	2302
60%	1215	3106
80%	1291	3391

datasets from PISCES may contain some proteins that the newer BLAST methods consider to be too similar. So, we only considered the non–redundant datasets provided by Leaf. Leaf also generates datasets that are 10% larger, even using the same PSI–BLAST (Bull et al., 2013). We generated four culled or non–redundant lethal and viable datasets from our original dataset where all proteins share sequence similarity less than a threshold of 20%, 40%, 60% or 80%. The numbers of lethal and viable proteins retrieved for each identity threshold from Leaf are summarised in Table 3.1.

3.3 Analysis of Genomic Features

The functionality of a gene may rely on its inherent sequence features at the genomic level. Analysing these gene sequence based features may provide valuable insights into their contributions to phenotypic fitness. This section covers results of the analysis done with several genomic features that we examined to assess their associations with gene essentiality.

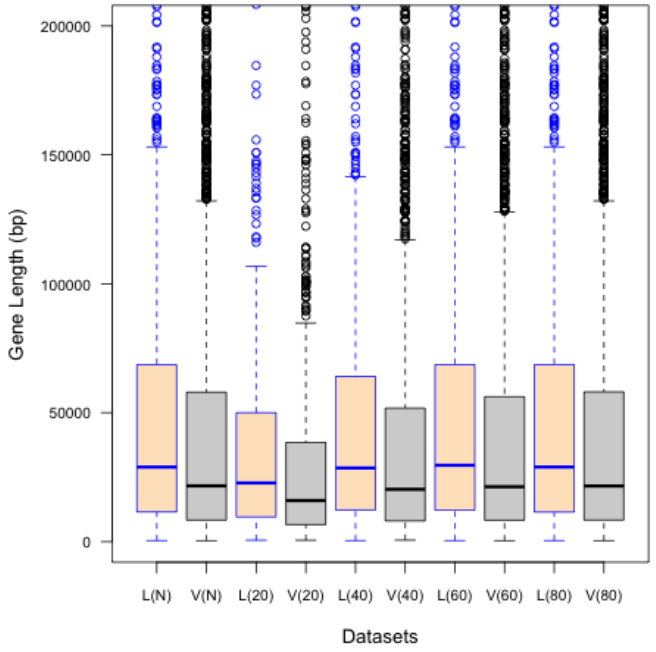


FIGURE 3.1: Distributions of total length of lethal and viable genes. Here, L(N) and V(N) refer to lethal and viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset where all coded proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the median; and individual points denote the outliers.

3.3.1 Gene Length and GC Content

Studies showed that gene selection during evolution can often be determined from the nucleic acid composition of a genome (Knight et al., 2001). Also, proteins show a trend of becoming longer in length throughout evolution (Lipman et al., 2002). We, therefore, anticipated that genomic features like GC content and gene length could be indicative of gene essentiality.

Our analysis showed that lethal genes tend to be longer in length compared to viable genes while considering the entire (non-culled) dataset. This result was also consistent for lethal and viable genes in the culled datasets. Figure 3.1 shows

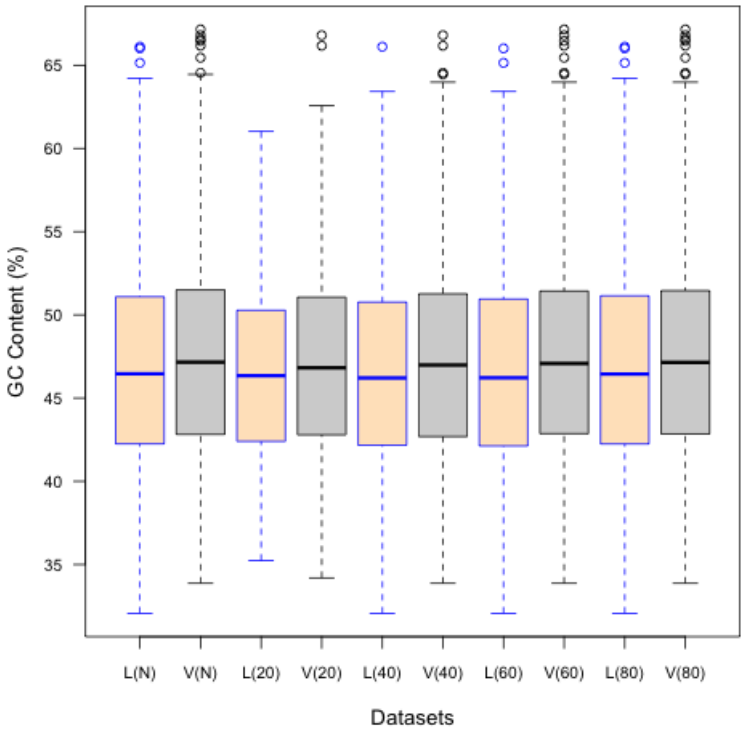


FIGURE 3.2: Distributions of the percentage of GC content in lethal and viable genes. Here, L(N) and V(N) refer to lethal and viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset where all coded proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the median; and individual points denote the outliers.

the comparison of the gene length distributions between lethal and viable genes for non–culled and culled datasets. The Mann–Whitney U test (Mann and Whitney, 1947) had also supported this trend by showing that the differences in gene length between lethal and viable genes are statistically significant. For the non–culled dataset, Mann–Whitney U test gave a p–value of 7.9×10^{-8}, which is < 0.0083 (Bonferroni–corrected p–value). For 20%, 40%, 60%, and 80% culled datasets, p–values were 5.0×10^{-6}, 1.4×10^{-8}, 1.4×10^{-9} and 1.8×10^{-7}, respectively.

When the distributions of GC content in lethal and viable genes were examined, we observed that viable genes have a higher percentage of GC content compared to lethal genes in both non–culled and culled datasets (Figure 3.2). After being adjusted for multiple testing by the Bonferroni correction method, the resulting p–value from the Mann–Whitney U test did not show the statistical significance of this result for the non–culled dataset (Table 3.2). However, differences in GC content distributions were statistically significant for the culled dataset where all coded proteins have a sequence identity $< 60\%$; this observation was not statistically significant for other culled datasets (Table 3.2).

TABLE 3.2: Results from the analysis of several genomic features of lethal and viable genes and the corresponding p-values from Mann–Whitney U test. The median value of each feature was considered to remove the effect of outliers. Highlighted cells in yellow represent statistically significant results based on the Bonferroni corrected p–value of 0.0083; blue cells refer to either lethal or viable genes where a feature shows a higher value.

Datasets		**Gene Sequence Features**				
		GC content (%)	**Transcript count**	**Exon count**	**Exon length** (bp)	**Intron length** (bp)
Non–culled	Lethal	46.46	4	11	3398	25341
	Viable	47.16	2	8	2780	18563
	p–value	0.009	4.74×10^{-16}	9.45×10^{-16}	1.18×10^{-22}	2.00×10^{-06}
culled(20%)	Lethal	46.34	4	10	2831	19226
	Viable	46.82	2	7	2263	13761
	p–value	0.091	1.27×10^{-10}	6.34×10^{-09}	2.18×10^{-10}	5.80×10^{-05}
culled(40%)	Lethal	46.21	4	11	3368	24928
	Viable	46.98	2	8	2631.5	17667
	p–value	0.009	1.81×10^{-15}	1.25×10^{-20}	9.10×10^{-23}	3.26×10^{-07}
culled(60%)	Lethal	46.22	4	11	3408.5	25964
	Viable	47.08	2	8	2745.5	18309.5
	p–value	0.001	4.47×10^{-17}	2.20×10^{-18}	2.72×10^{-24}	4.08×10^{-08}
culled(80%)	Lethal	46.445	4	11	3397.5	25332.5
	Viable	47.145	2	8	2767	18534
	p–value	0.013	5.50×10^{-16}	1.23×10^{-15}	1.22×10^{-22}	3.00×10^{-06}

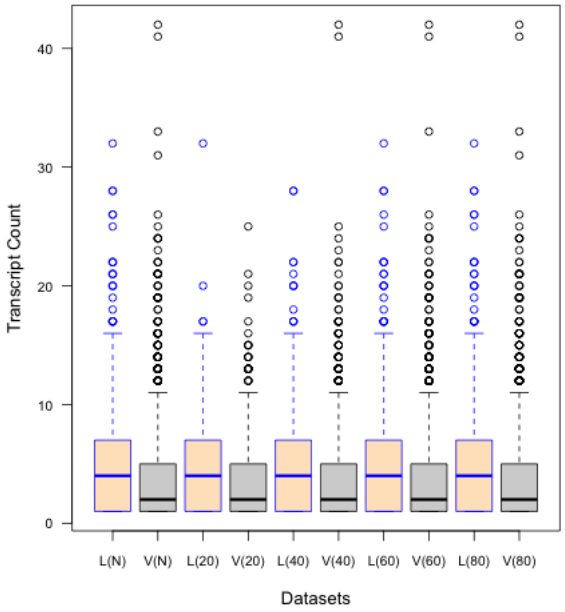


FIGURE 3.3: Distributions of the number of transcripts in lethal and viable genes. Here, L(N) and V(N) refer to lethal and viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset where all coded proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

3.3.2 Number of Gene Transcripts and Exons

Exons (coding regions) and introns (non–coding regions) are the two key sequence elements that build up gene architecture. Alternative splicing produces multiple transcripts, the translation of which contributes to multiple proteins encoded by a single gene. Initially introns were thought to be 'junk' DNA. This notion has been ruled out by several pieces of experimental evidence, which revealed that some introns could be expressed in the form of non-coding RNAs (*e.g.* microRNAs and small nucleolar RNAs). They may also play important functional roles in the transcriptional activity of cells (Comeron, 2001). Hence, we expected that these genomic properties could serve as key characteristics for essential genes.

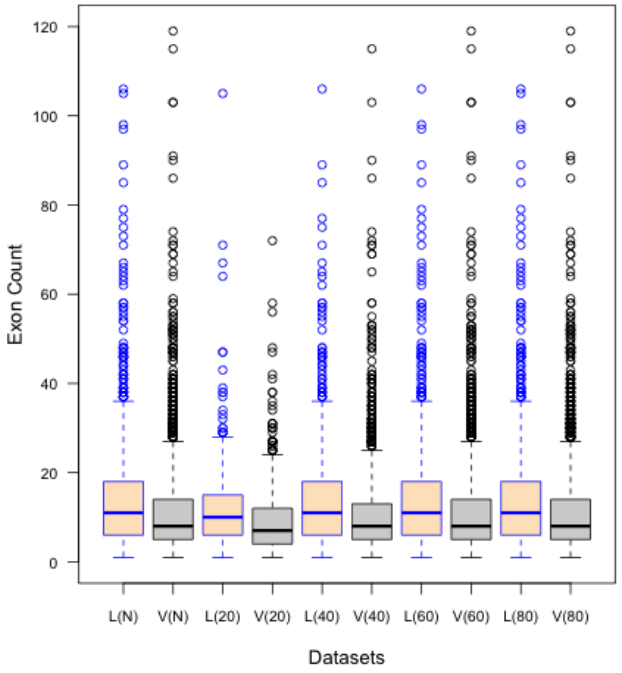


FIGURE 3.4: Distributions of the number of exons in lethal and viable genes. Here, L(N) refers to lethal and V(N) refers viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset, respectively, where all coded proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

Examining the number of transcripts of genes from the non–culled and culled datasets revealed that lethal genes tend to have more transcripts than viable genes (Figure 3.3). Differences in transcript number between lethal and viable genes were also statistically significant (Table 3.2). To quantify whether or not the number of exons could differentiate between lethal and viable genes, exon rank from the longest transcript of genes was analysed here. We identified that lethal genes have more exons than viable genes (Figure 3.4). This finding was also statistically significant for non–culled and culled datasets (Table 3.2).

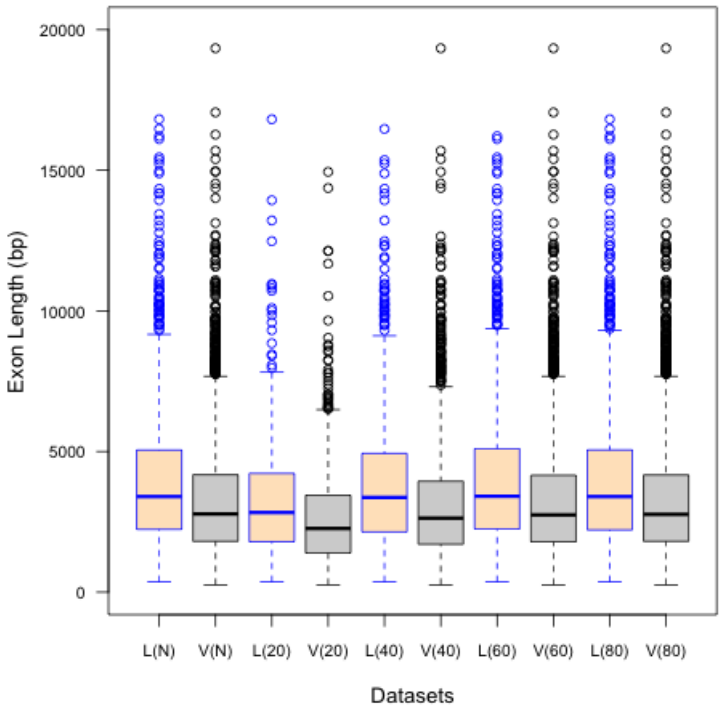


FIGURE 3.5: Distributions of the total length of exons in lethal and viable genes. Here, L(N) refers to lethal and V(N) refers viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset, respectively, where all coded proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

3.3.3 Lengths of Exons and Introns

When the distributions of the length of exons of lethal and viable genes were examined, we observed that lethal genes tend to have longer exon length than viable genes (Figure 3.5). Our analysis also identified that introns are significantly longer in lethal genes (Figure 3.6). The p–values in Table 3.2 also show that these differences in exon and intron length between lethal and viable genes of the non–culled and culled datasets are statistically significant.

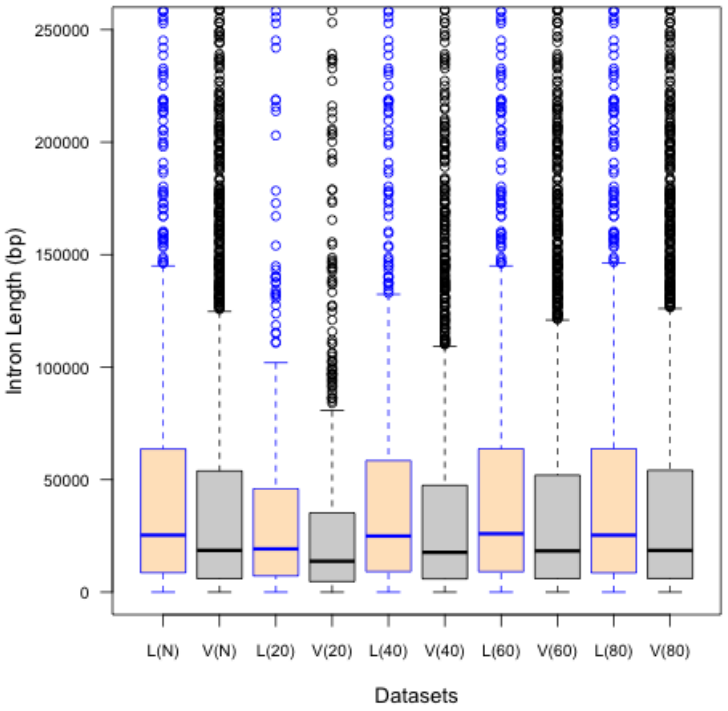


FIGURE 3.6: Distributions of the total intron length in lethal and viable genes. Here, L(N) refers to lethal and V(N) refers viable genes in the non–culled dataset. L(xx) and V(xx) define lethal and viable genes in the culled dataset, respectively, where all coded proteins share sequence similarity less than xx%. Top 5% lethal and viable genes with longest introns were excluded from the datasets to make plots more readable. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

3.3.4 Gene Expression

Gene expression is a way of quantifying if a particular gene is active in a biological process. We, therefore, expected that genes with embryonic expression would be more likely to be essential.

We obtained mouse gene expression data from the UniGene database (Stanton et al., 2003) for 13 developmental stages including oocyte, unfertilized ovum, zygote, cleavage, morula, blastocyst, egg cylinder, gastrula, organogenesis, fetus,

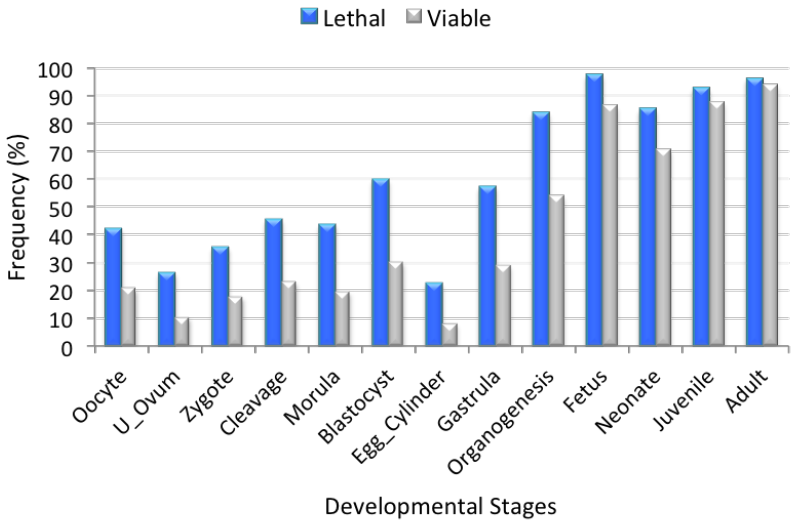


FIGURE 3.7: Frequencies (%) of lethal and viable mouse genes in the non–culled datasets that are expressed at 13 embryonic developmental stages.

neonate, juvenile and adult. Developmental expression data were found for 1,301 lethal and 3,409 viable genes. While comparing expressions of lethal and viable genes in the non–culled dataset, we observed that lethal genes are expressed in higher proportions compared to viable genes at almost every stage of mouse development (Figure 3.7). However, the χ^2 tests with the Bonferroni correction analysis showed that these differences are not statistically significant at later stages of development as all genes are nearly always expressed at those stages (Table 3.3). Lethal genes were further found being highly expressed, whereas viable genes are more likely to be found in the group with zero transcripts, while comparing differences in gene expression distributions (Figure 3.8). These differences in developmental expression were also observed for culled datasets.

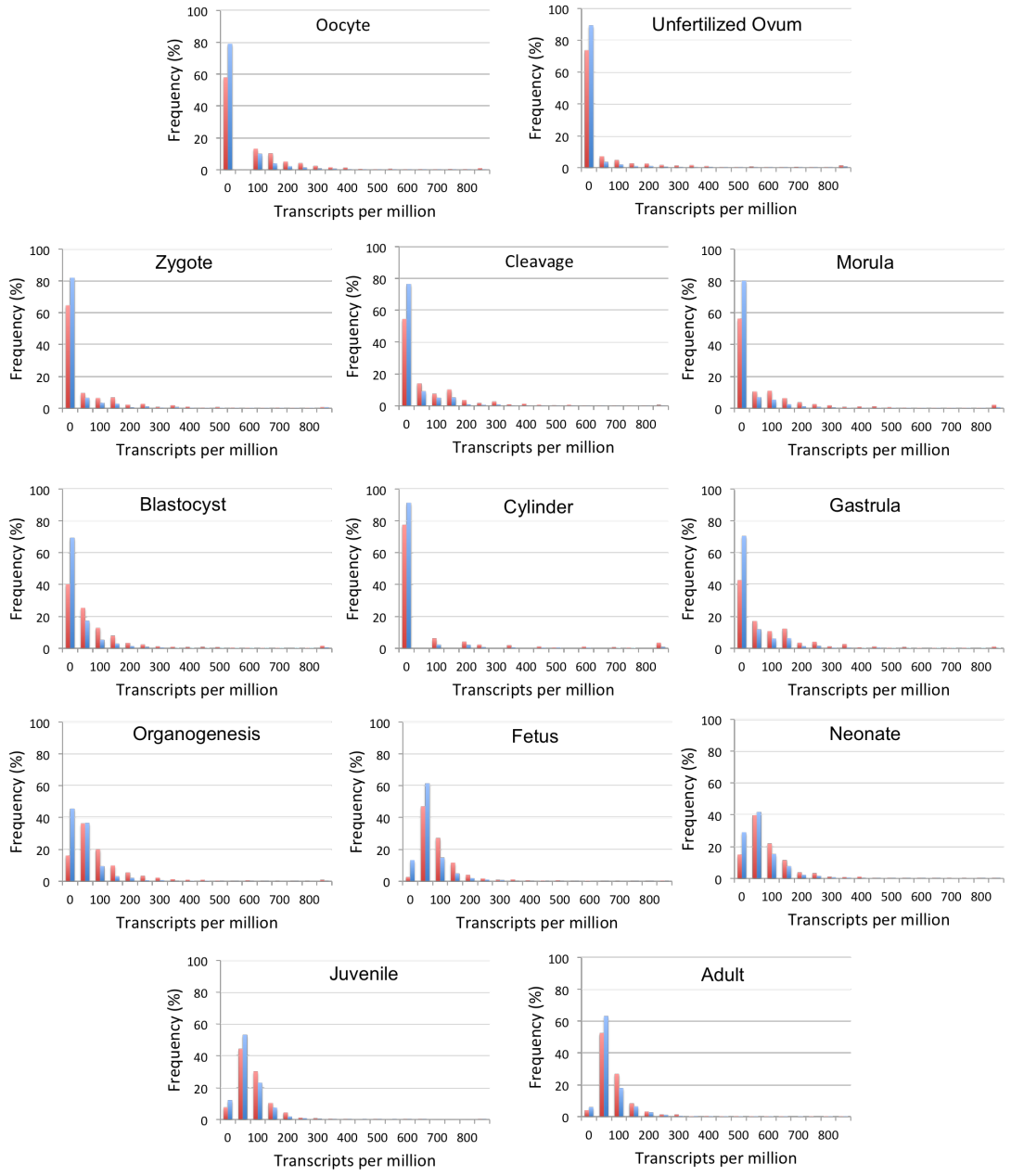


FIGURE 3.8: Gene expression distributions of lethal (red) and viable (blue) genes in the non–culled dataset across 13 stages of mouse development. Here, the bin size is 50.

TABLE 3.3: Differences in proportions of lethal versus viable mouse genes in the non–culled dataset that are expressed across 13 embryonic developmental stages. Highlighted cells in gray represent statistically insignificant results based on the Bonferroni corrected p–value of 0.00385.

Developmental stages	**Lethal(%)**	**Viable(%)**	**p–value**
Oocyte	42.01	20.86	3.59×10^{-36}
U_ovum	26.27	10.50	5.96×10^{-36}
Zygote	35.33	17.84	2.95×10^{-29}
Cleavage	45.47	23.23	3.00×10^{-36}
Morula	43.63	19.71	2.21×10^{-46}
Blastocyst	59.75	30.36	1.23×10^{-47}
Egg_Cylinder	22.35	8.30	6.01×10^{-35}
Gastrula	57.14	29.25	8.07×10^{-45}
Organogenesis	83.79	54.12	1.02×10^{-30}
Fetus	97.31	86.42	4.27×10^{-04}
Neonate	84.95	70.72	5.07×10^{-07}
Juvenile	92.40	87.80	0.135
Adult	95.93	94.05	0.562

3.3.5 Evolutionary Age

The evolutionary age of a gene represents the time that has passed since the gene evolved from its ancestor either by duplication or speciation. Studies in bacteria and yeast found essential genes to be evolutionarily more conserved than non-essential genes (Jordan et al., 2002; Giaever et al., 2002; Gustafson et al., 2006). We, therefore, expected that gene evolutionary age could be informative of mammalian gene essentiality.

The evolutionary age reported in millions of years ago (MYA) were estimated by analysing the Ensembl (release 75) gene trees (mentioned in section 2.2.1.3). The age of the duplicate (most distant) common ancestor (DCA) or the most recent duplication (MRD) event was assigned to each mouse duplicate gene. The age of the single common ancestor (SCA) was assigned to each mouse singleton gene. We observed 16 representative age groups for our mouse genes (Table 3.4A). We found

TABLE 3.4: (A) Mouse age groups in million years ago (MYA) obtained from the Ensembl (release 75) database (B) Statistical test results presenting lethal versus viable mouse genes frequencies for different MRD age groups. (C) Statistical test results presenting lethal versus viable mouse genes frequencies for different DCA age groups. These results are observed for the non–culled dataset. Highlighted cells in yellow represent statistically significant results based on Bonferroni corrected p–value of 0.003125.

(A)

Taxon	**Age**
Murinae	25
Rodentia	77
Sciurognathi	78
Glires	86
Euarchontoglires	92
Eutheria	104
Theria	162
Mammalia	167
Amniota	296
Tetrapoda	371
Euteleostomi	400
Sarcopterygii	414
Vertebrata	535
Chordata	722
Bilateria	937
Opisthokonta	1215

(B)

MRD Age	**Lethal (%)**	**Viable (%)**	**p–value**
25	0.63	2.17	3.7×10^{-4}
77	0.00	0.09	0.286
78	0.00	0.06	0.383
86	0.24	0.06	0.104
92	0.55	0.51	0.858
104	11.60	11.23	0.737
162	1.25	2.80	2.3×10^{-3}
167	3.45	5.03	0.025
296	5.41	4.79	0.403
371	1.41	2.56	0.020
400	31.58	41.69	8.3×10^{-7}
414	1.80	2.74	0.070
535	13.48	14.80	0.290
722	4.00	3.25	0.219
937	15.67	6.08	4.8×10^{-23}
1215	8.93	2.14	6.6×10^{-25}

(C)

Taxon	**DCA Age**	**Lethal (%)**	**Viable (%)**	**p–value**
Murinae	25	0.16	0.36	0.266
Rodentia	77	0.00	0.00	0
Sciurognathi	78	0.00	0.00	0
Glires	86	0.00	0.00	0
Euarchontoglires	92	0.00	0.00	0
Eutheria	104	0.39	1.36	4.5×10^{-3}
Theria	162	0.31	1.24	4.2×10^{-3}
Mammalia	167	0.39	1.13	0.019
Amniota	296	0.70	2.10	1.1×10^{-3}
Tetrapoda	371	0.16	1.13	1.4×10^{-3}
Euteleostomi	400	10.34	16.58	7.8×10^{-7}
Sarcopterygii	414	0.00	1.36	2.9×10^{-5}
Vertebrata	535	10.42	14.51	6.5×10^{-4}
Chordata	722	5.60	7.94	8.3×10^{-3}
Bilateria	937	52.72	41.01	7.8×10^{-8}
Opisthokonta	1215	18.82	11.28	3.1×10^{-10}

ages for 1,276 (98.1%) lethal and 3,358 (97.3%) viable genes from Ensembl. The age of the oldest genes is approximately 1215 MYA, whereas the youngest genes

belong to the class Murinae and are approximately 25 MYA old. We compared the enrichment of lethal and viable genes in different age groups. We found that lethal genes are older than viable genes for both non–culled and culled datasets (Figure 3.9). We observed that a significantly greater percentage of lethal genes have evolutionary age of 1215 and 937 MYA compared to viable genes in the non–culled dataset (Table 3.4B and Table 3.4C). The majority of the viable genes are 400 MYA old. Also, viable genes are found in great proportions to have the age of 25 and 162 MYA while considering MRD ages (Table 3.4B). We further observed a significantly greater percentage of viable genes that have DCA age of 296, 371, 414 and 535 MYA (Table 3.4C). We found similar trends for the culled datasets, which further confirms that genes essential for mouse development are more ancient.

3.4 Analysis of Protein Sequence Features

Proteins are the mediators of different gene functions and thus, it is likely that gene essentiality links to many other characteristics that could be gleaned from protein sequence data. Prior research established that different physical, functional and evolutionary properties of proteins can facilitate the prediction of gene essentiality (Gustafson et al., 2006; Seringhaus et al., 2006; Palaniappan and Mukherjee, 2011; Yuan et al., 2012). In this research, we explored a number of protein properties, which are easily obtainable from mouse protein sequence data. It is necessary to quantify how much information each of these properties carries with respect to essentiality. This section covers results of the analyses done with several protein

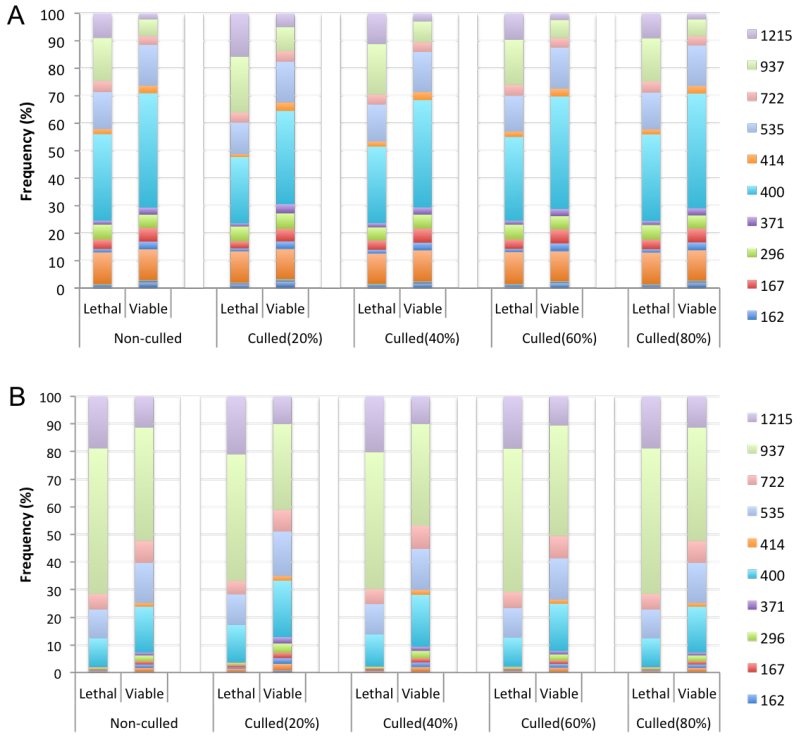


FIGURE 3.9: Proportions of lethal and viable genes for different age groups. Here, ages of mouse duplicates were calculated based on the Most Recent Duplication (MRD) event (A) or the Duplicate Common Ancestor (DCA) (B).

sequence features to test their efficacy of distinguishing lethal genes from viable genes in mouse.

3.4.1 Simple Sequence Features

When we investigated the lengths of lethal and viable protein sequences in the non–culled dataset, we found that lethal proteins tend to have significantly longer length than viable proteins (529*aa* versus 452*aa* (median length); p–value $= 1.0 \times 10^{-21}$). We observed this significant difference in protein lengths also for culled datasets. The estimated p–values for 20%, 40%, 60%, and 80% culled datasets were 6.6×10^{-10}, 3.8×10^{-21}, 7.3×10^{-22} and 3.7×10^{-21}, respectively. Figure

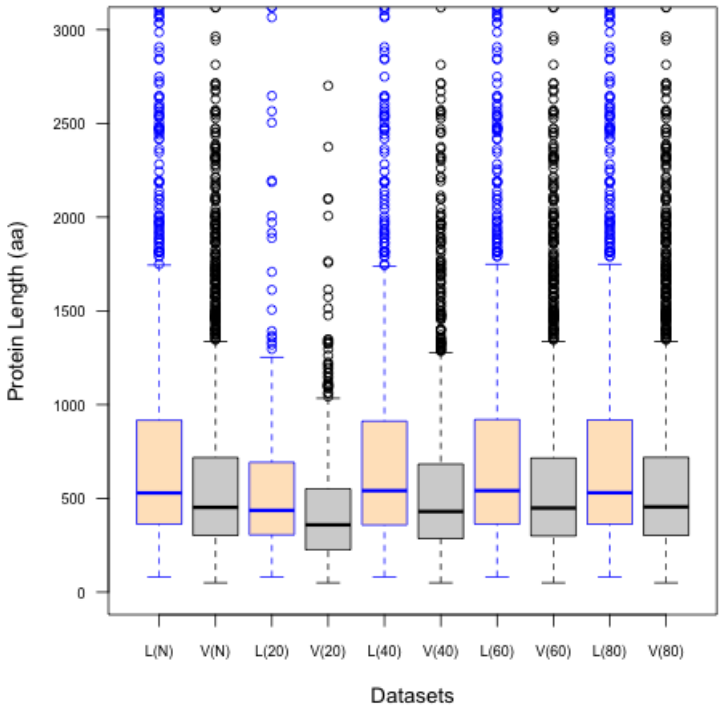


FIGURE 3.10: Distributions of the length of proteins encoded by lethal and viable genes. Here, L(N) refers to lethal and V(N) refers viable proteins in the non–culled dataset. L(xx) and V(xx) define lethal and viable proteins in the culled dataset, respectively, where all proteins share sequence similarity less than xx%. Top 2% longest proteins were excluded from the datasets to make plots more readable. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

3.10 shows the distributions of protein lengths between lethal and viable proteins within the non–culled and culled datasets.

We computed the amino acid frequencies from lethal and viable protein sequences. Table 3.5 shows differences in amino acid frequencies observed between lethal and viable proteins in the non–culled dataset. Our investigation found that proteins encoded by lethal genes in the non–culled dataset tend to have higher proportions of Alanine, Aspartic acid, Glutamic acid, Lysine, Glutamine and Serine residues. Distributions of Lysine residues demonstrated the same trend for

TABLE 3.5: Differences in 20 amino acid percentages observed between lethal and viable mouse proteins in the non–culled dataset. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.0025; blue cells refer to lethal/viable proteins where that particular amino acid is found in a higher proportion.

Amino acid	Lethal	Viable	p–value
A	6.87	6.74	1.05×10^{-03}
C	1.88	2.08	1.92×10^{-07}
D	4.91	4.73	4.14×10^{-07}
E	6.68	6.22	6.48×10^{-12}
F	3.39	3.80	1.77×10^{-17}
G	6.43	6.49	0.280
H	2.48	2.39	0.005
I	4.06	4.24	3.20×10^{-04}
K	5.67	5.15	3.64×10^{-14}
L	9.31	10.00	2.69×10^{-21}
M	2.19	2.21	0.841
N	3.63	3.50	0.002
P	5.86	5.72	0.026
Q	4.48	4.25	3.78×10^{-07}
R	5.41	5.38	0.326
S	8.01	7.78	0.001
T	5.14	5.24	0.008
V	5.89	6.25	2.73×10^{-12}
W	1.01	1.31	8.84×10^{-24}
Y	2.73	2.83	0.019

all culled datasets. Lethal proteins in the 40%, 60% and 80% culled dataset further had Aspartic acid, Glutamic acid and Glutamine residues in high proportions compared to viable proteins (Table 3.6). Differences in these amino acids were not statistically significant for the 20% culled dataset. Alanine and Serine residues also showed their enrichment in lethal proteins for the 80% culled dataset, whereas no statistically significant difference was found for Alanine in other culled datasets. On the other hand, viable proteins were found to have higher proportions of Leucine residues for non-culled and culled datasets (Table 3.5 and Table 3.6).

Viable genes were also found to be rich in Cysteine, Phenylalanine, Valine

TABLE 3.6: Differences in amino acid frequencies observed between lethal and viable mouse proteins in the culled datasets. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.0025; blue cells refer to lethal or viable proteins where the amino acid is found in higher proportion. L(xx) refers to lethal and V(xx) refers to viable proteins in the culled dataset, where all coded proteins share sequence similarity less than xx%.

Amino Acid	Lethal (20)	Viable (20)	p–value	Lethal (40)	Viable (40)	p–value
A	6.90	6.81	0.404	6.88	6.74	6.1×10^{-3}
C	1.95	2.07	0.028	1.85	2.09	1.7×10^{-7}
D	4.85	4.71	0.027	4.97	4.69	1.9×10^{-9}
E	6.64	6.30	4.0×10^{-3}	6.83	6.20	6.8×10^{-13}
F	3.79	3.94	0.132	3.47	3.79	5.6×10^{-8}
G	6.27	6.31	0.148	6.21	6.35	0.090
H	2.33	2.32	0.737	2.43	2.40	0.683
I	4.50	4.18	0.029	4.19	4.20	0.429
K	5.80	5.36	6.9×10^{-5}	5.81	5.14	7.9×10^{-15}
L	9.81	10.35	1.5×10^{-3}	9.52	10.18	3.2×10^{-14}
M	2.30	2.34	0.883	2.22	2.23	0.591
N	3.57	3.48	0.156	3.63	3.50	0.012
P	5.28	5.42	0.374	5.62	5.66	0.706
Q	4.28	4.25	0.353	4.50	4.34	2.3×10^{-3}
R	5.25	5.26	0.725	5.38	5.32	0.343
S	7.59	7.64	0.497	7.83	7.82	0.565
T	5.15	5.14	0.919	5.16	5.24	0.131
V	6.30	6.34	0.936	6.05	6.28	1.9×10^{-4}
W	1.17	1.36	0.017	1.03	1.32	5.9×10^{-14}
Y	2.99	2.84	0.346	2.72	2.80	0.078

Amino Acid	Lethal (60)	Viable (60)	p–value	Lethal (80)	Viable (80)	p–value
A	6.81	6.74	0.010	6.87	6.74	1.1×10^{-3}
C	1.86	2.10	2.2×10^{-8}	1.86	2.08	8.1×10^{-8}
D	4.93	4.68	1.4×10^{-9}	4.91	4.71	1.4×10^{-7}
E	6.71	6.19	1.7×10^{-13}	6.68	6.20	3.3×10^{-12}
F	3.41	3.77	2.3×10^{-14}	3.39	3.79	9.2×10^{-18}
G	6.36	6.46	0.134	6.42	6.49	0.217
H	2.47	2.40	0.036	2.48	2.39	7.5×10^{-3}
I	4.09	4.18	0.038	4.05	4.22	3.5×10^{-4}
K	5.69	5.08	9.9×10^{-16}	5.66	5.13	2.6×10^{-14}
L	9.33	10.07	5.5×10^{-22}	9.32	10.00	7.8×10^{-21}
M	2.17	2.19	0.624	2.18	2.20	0.998
N	3.64	3.50	1.6×10^{-3}	3.63	3.5	4.8×10^{-3}
P	5.82	5.74	0.186	5.88	5.73	0.020
Q	4.46	4.28	4.1×10^{-5}	4.47	4.24	3.3×10^{-7}
R	5.41	5.36	0.294	5.41	5.37	0.257
S	7.99	7.83	0.017	8.02	7.80	1.6×10^{-3}
T	5.15	5.23	0.061	5.13	5.24	0.012
V	5.94	6.25	8.3×10^{-9}	5.88	6.25	1.3×10^{-11}
W	1.00	1.32	1.3×10^{-23}	1.00	1.31	9.1×10^{-25}
Y	2.74	2.81	0.122	2.72	2.82	0.024

and Tryptophan residues, but this trend was not consistent with the 20% culled dataset. Viable proteins in the non–culled and 80% culled datasets further showed Isoleucine residues in high proportions. No such result was observed for other

TABLE 3.7: Median values of different protein features obtained from Pepstats and the p–values of their distribution calculated using Mann–Whitney U test. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.0038; blue cells refers to either lethal or viable proteins where it exhibits a higher value.

Datasets		Protein Sequence Features							
		Molecular weight(Da)	**Aliphatic** (%)	**Aromatic** (%)	**Non-polar** (%)	**Polar** (%)	**Charged** (%)	**Basic** (%)	**Acidic** (%)
Non–culled	Lethal	59146.21	27.00	10.05	52.12	47.88	25.79	13.97	11.78
	Viable	50446.09	27.81	10.75	53.73	46.27	24.53	13.26	11.03
	p–value	3.3×10^{-21}	3.5×10^{-13}	3.7×10^{-14}	4.4×10^{-27}	4.6×10^{-27}	2.2×10^{-18}	2.1×10^{-15}	1.7×10^{-13}
culled(20%)	Lethal	48925.94	28.21	10.51	53.03	46.97	25.80	13.99	11.69
	Viable	40326.80	28.36	10.82	54.04	45.96	24.65	13.37	11.02
	p–value	4.1×10^{-10}	0.41	0.19	1.0×10^{-04}	1.0×10^{-04}	7.6×10^{-05}	4.8×10^{-04}	6.3×10^{-04}
culled(40%)	Lethal	60211.77	27.40	10.08	52.10	47.90	26.02	13.96	11.97
	Viable	48362.45	27.96	10.67	53.68	46.32	24.47	13.26	10.98
	p–value	3.2×10^{-21}	6.0×10^{-06}	4.0×10^{-09}	1.5×10^{-20}	1.6×10^{-20}	3.7×10^{-19}	5.5×10^{-12}	3.4×10^{-16}
culled(60%)	Lethal	60237.43	27.08	10.08	52.04	47.96	25.86	13.96	11.83
	Viable	50057.85	27.81	10.69	53.76	46.24	24.39	13.23	10.96
	p–value	1.3×10^{-21}	3.3×10^{-11}	2.3×10^{-12}	4.8×10^{-28}	5.1×10^{-28}	1.9×10^{-21}	1.9×10^{-15}	4.6×10^{-17}
culled(80%)	Lethal	59284.56	27.00	10.05	52.10	47.90	25.81	13.97	11.79
	Viable	50479.42	27.78	10.73	53.74	46.26	24.50	13.24	11.01
	p–value	1.1×10^{-20}	1.1×10^{-12}	1.4×10^{-14}	1.0×10^{-27}	1.1×10^{-27}	3.8×10^{-19}	8.6×10^{-16}	3.3×10^{-14}

culled datasets. Differences between lethal and viable datasets with respect to proportions of other amino acids were not found to be statistically significant.

Protein average molecular weight, charge, isoelectric point and frequencies of different amino acid categories were computed using the tool Pepstats (Rice et al., 2000). Our analysis found that proteins encoded by lethal genes have a significantly higher average molecular weight (MW) compared to proteins encoded by viable genes (Table 3.7). Differences in charge, isoelectric point, tiny and small residues were not statistically significant. Lethal proteins were found to have greater proportions of polar (Figure 3.11), charged (Figure 3.12), basic (Figure 3.13) and acidic (Figure 3.14) amino acids. The results are meaningful because these four amino acid groups are interconnected. The Mann-Whitney U test further supported these observations for all datasets (Table 3.7).

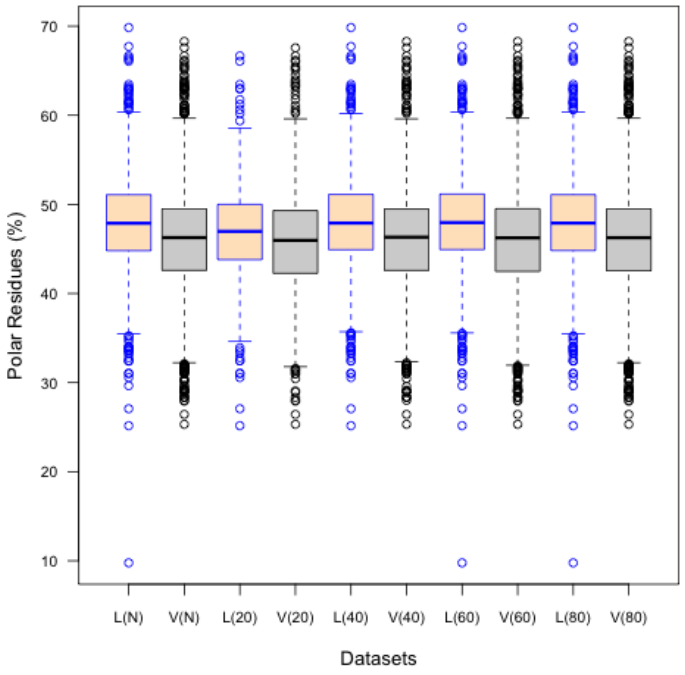


FIGURE 3.11: Distributions of the polar residues (%) between lethal and viable proteins.

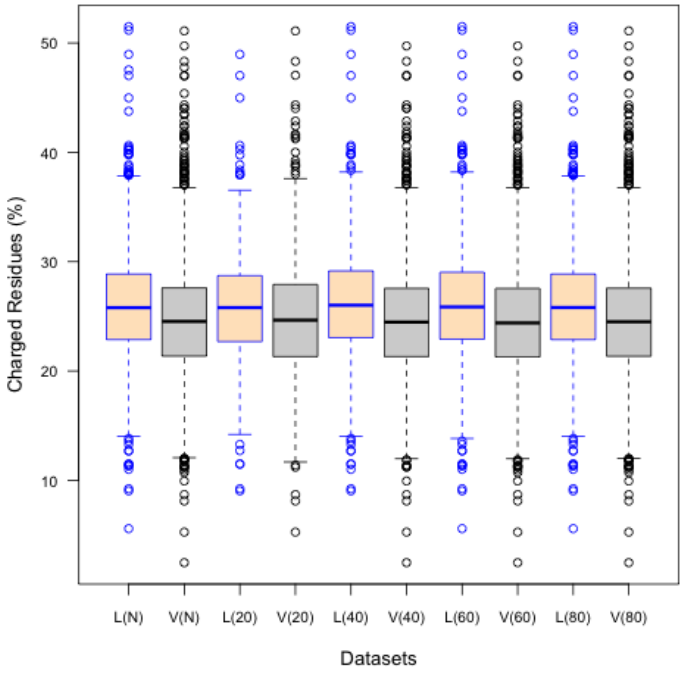


FIGURE 3.12: Distributions of the charged residues (%) between lethal and viable proteins.

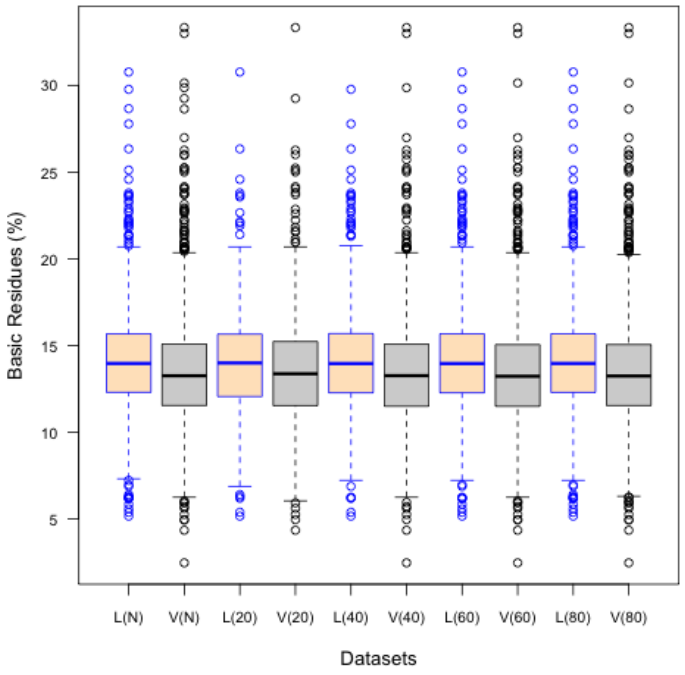


FIGURE 3.13: Distributions of the basic residues (%) between lethal and viable proteins.

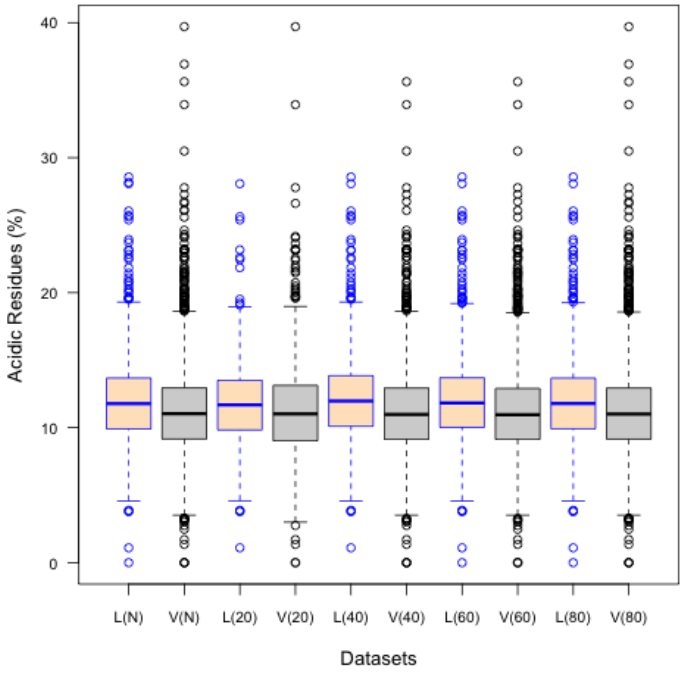


FIGURE 3.14: Distributions of the acidic residues (%) between lethal and viable proteins.

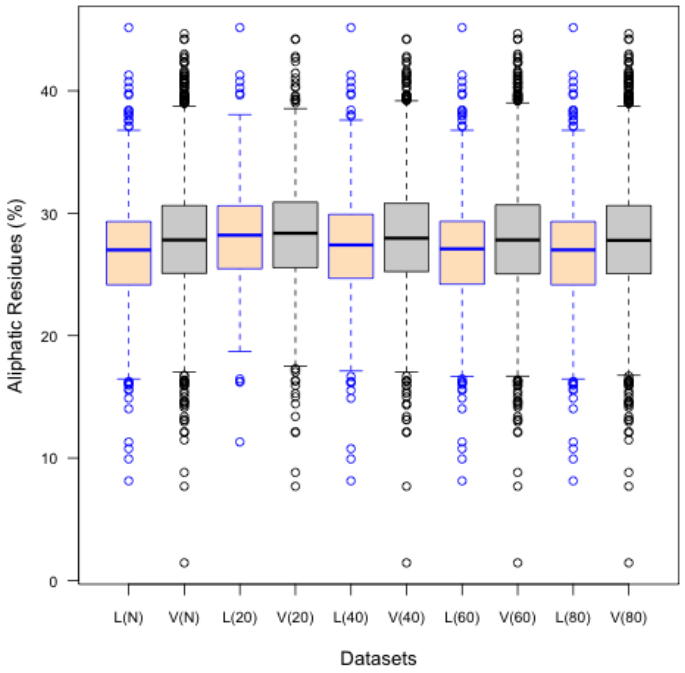


FIGURE 3.15: Distributions of the aliphatic residues (%) between lethal and viable proteins.

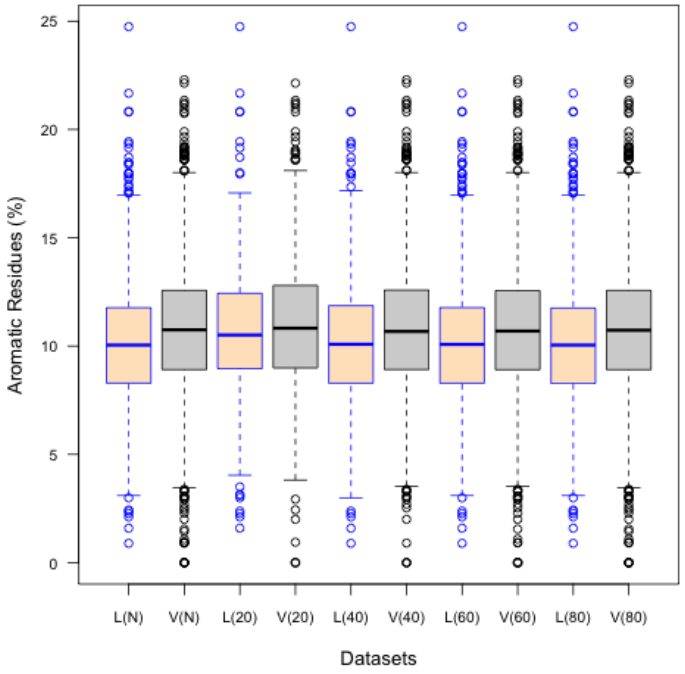


FIGURE 3.16: Distributions of the aromatic residues (%) between lethal and viable proteins.

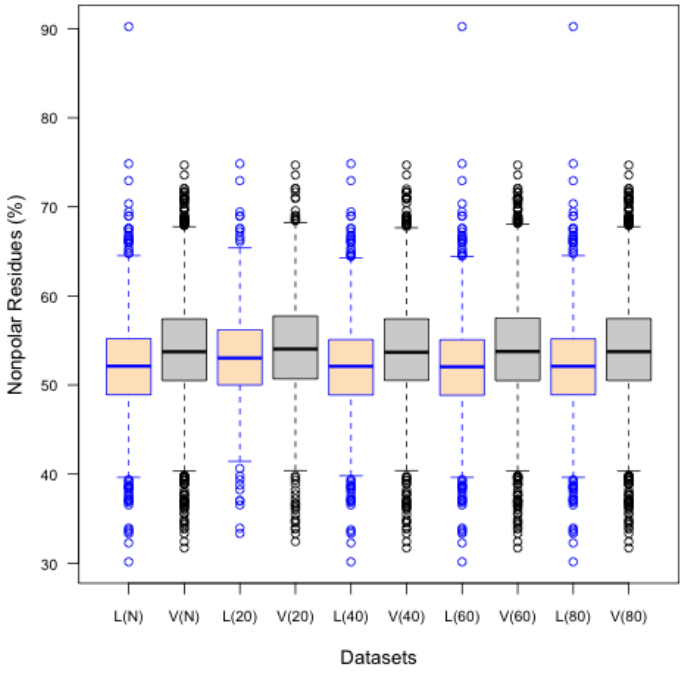


FIGURE 3.17: Distributions of the non-polar residues (%) between lethal and viable proteins.

We observed that proteins encoded by viable genes have significantly higher proportions of aliphatic (Figure 3.15), aromatic (Figure 3.16) and non-polar residues (Figure 3.17). However, the Mann–Whitney U test showed that differences of aliphatic (p–value = 0.419764) and aromatic (p–value = 0.191405) residues between lethal and viable proteins in the 20% culled datasets are not statistically significant. Table 3.7 summarises all these results. For Figures 3.11–3.17, L(N) refers to lethal and V(N) refers viable proteins in the non–culled dataset. L(xx) and V(xx) define lethal and viable proteins in the culled dataset, respectively, where all proteins share sequence similarity less than xx%. In this box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the median; and individual points denote the outliers.

3.4.2 Enzyme Class

Almost all cellular processes are dependent on the presence of enzymes at significant levels. Enzymatic function thereby could be another measure of gene essentiality. We extracted the annotations of six primary enzyme classes from the UniProt database (Apweiler et al., 2004) and counted the number of lethal and viable proteins belonging to each of these classes. In the non-culled datasets, 29.82% (388/1301) of the total number of lethal proteins exhibit enzymatic activity compared to 27.70% (956/3451) of viable proteins. The percentage of lethal genes that are an enzyme was found to be 36.33% (174/479), 32.57% (313/961), 30.62% (372/1215) and 29.82% (385/1291) for 20%, 40%, 60% and 80% culled datasets, respectively, whereas these numbers are 32.05% (326/1017), 28.37% (653/2302), 27.95% (868/3106) and 27.87% (945/3391) for viable proteins. Figure 3.18 shows the distribution of the six principal enzyme class numbers between lethal and viable datasets for the non–culled dataset. We observed that lethal genes are rich in Transferase (13.45% versus 10.08%, p–value = 1.81×10^{-3}) and Ligase (2.92% versus 1.22%, p–value = 5.43×10^{-5}).

On the other hand, Hydrolases were found to be strongly associated with viable proteins in the non–culled dataset (8.07% versus 10.92%, p–value = 5.9×10^{-3}). Table 3.8 shows that these observations were also consistent with the culled datasets though for the 20% culled dataset, these differences were not statistically significant. No statistically significant results were observed for Oxidoreductase, Lyase and Isomerase enzymes.

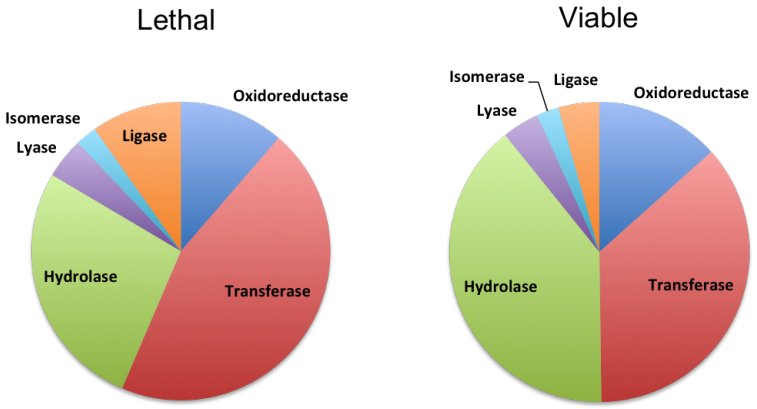


FIGURE 3.18: Percentages of lethal and viable proteins in the non–culled dataset for different enzyme classes

TABLE 3.8: Differences in the frequencies of different enzyme class observed between lethal and viable mouse proteins in the culled datasets. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.0083; blue cells refer to either lethal or viable proteins where it exhibits a higher value.

Datasets		Enzyme Classes					
		Oxidoreductase	Transferase	Hydrolase	Lyase	Isomerase	Ligase
Culled(20%)	Lethal	6.05	12.94	10.02	2.92	1.46	2.92
	Viable	5.99	9.63	12.58	1.17	1.37	1.27
	p–value	0.966	0.068	0.177	0.017	0.897	0.027
Culled(40%)	Lethal	3.95	13.42	9.26	1.76	0.93	3.22
	Viable	4.17	9.29	11.55	1.25	0.82	1.25
	p–value	0.781	9.18×10^{-04}	0.070	0.264	0.754	1.60×10^{-04}
Culled(60%)	Lethal	3.45	13.82	8.06	1.39	0.74	3.12
	Viable	3.76	9.52	11.59	1.15	0.70	1.19
	p–value	0.632	1.06×10^{-04}	1.38×10^{-03}	0.521	0.909	1.40×10^{-05}
Culled(80%)	Lethal	3.40	13.55	7.90	1.31	0.69	2.94
	Viable	3.77	10.05	11.02	1.12	0.67	1.20
	p–value	0.558	1.26×10^{-03}	2.69×10^{-03}	0.579	0.944	4.45×10^{-05}

3.4.3 Post–translational Modifications

We investigated four different keywords ('phosphoprotein', 'glycoprotein', 'acetylation' and 'transcription') that are present in UniProt protein annotations. In Uniprot, 'glycoprotein' and 'phosphoprotein' are the synonyms of glycosylation and phosphorylation processes. In this study, these keywords were selected, as

these are the most common PTMs and are important for controlling many fundamental cellular functions. Moreover, studies claimed that phosphorylation and glycosylation are crucial for predicting protein functions (Jensen et al., 2003).

Protein phosphorylation is a very common PTM that occurs when a phosphate group from ATP is added, normally to a hydroxyl group of the protein. Protein phosphorylation plays crucial roles in regulating various cellular and metabolic processes like cell differentiation, cell division, survival etc. Around 30% of all eukaryotic proteins are estimated to be phosphorylated (Mann et al., 2002). Our investigation showed that lethal proteins within the non–culled dataset are more likely to be phosphorylated than viable proteins. Statistical tests also showed the significance of this result (51.42% versus 35.50%, p–value = 8.9×10^{-15}). We observed the same trend for culled datasets (Table 3.9).

Glycosylation is another major form of PTM. More than 50% of all proteins are glycosylated (Apweiler et al., 1999). Glycoproteins are crucial for protein folding, solubility and localization (Weng et al., 2013). A large number of them are secreted extracellular proteins or are cell membrane proteins and they, therefore, have roles in transport and cell–cell interactions. UniProt only annotates glycoproteins with N–glycosylation sites. Table 3.9 shows the total percentage of N–linked glycoproteins that were found in all lethal and viable datasets. We observed that viable datasets contain significantly more N–linked glycoproteins than lethal datasets.

Acetylated proteins in eukaryotes are those proteins that are post–translationally

TABLE 3.9: Frequencies (%) of different keywords in lethal and viable mouse proteins and the corresponding p–values computed using the Chi–squared test. Highlighted cells in yellow represent statistically significant results based on Bonferroni corrected p–value of 0.0125; blue cells refer to either lethal or viable proteins where it exhibits a higher value.

Datasets		**Keywords**			
		Phosphoprotein	**Glycoprotein**	**Acetylation**	**Transcription**
Non–culled	Lethal	51.42	21.29	28.90	27.82
	Viable	35.50	38.19	12.75	11.45
	p–value	8.93×10^{-15}	3.05×10^{-19}	4.51×10^{-33}	1.77×10^{-36}
Culled(20%)	Lethal	40.50	20.04	30.69	15.87
	Viable	29.20	33.24	16.32	7.87
	p–value	3.73×10^{-04}	9.86×10^{-06}	1.45×10^{-08}	7.80×10^{-06}
Culled(40%)	Lethal	52.65	21.12	31.32	21.23
	Viable	32.71	38.10	13.64	9.56
	p–value	6.29×10^{-17}	1.57×10^{-14}	2.84×10^{-26}	3.45×10^{-17}
Culled(60%)	Lethal	52.26	21.40	29.71	26.09
	Viable	34.64	38.83	12.69	11.14
	p–value	1.28×10^{-16}	9.33×10^{-19}	2.25×10^{-33}	1.65×10^{-29}
Culled(80%)	Lethal	51.51	21.15	28.89	27.73
	Viable	35.39	38.40	12.59	11.47
	p–value	5.66×10^{-15}	9.54×10^{-20}	1.76×10^{-33}	1.45×10^{-35}

modified by the addition of an acetyl group, mostly at the N–terminus. The acetylation process is important for gene expression and metabolism as acetyl groups can cause genes and proteins to turn on and off. N–acetylated proteins also have vital roles in regulating of protein–protein interactions (Arnesen, 2011). Our analysis revealed that proteins encoded by lethal genes are likely to be more acetylated than proteins encoded by viable genes for all datasets (Table 3.9).

The keyword 'transcription' in UniProt annotates proteins that are involved in controlling the process of transcription (transcription factors). Our investigation found that lethal proteins in all datasets are more likely to be associated with regulating the transcription of genes. Table 3.9 shows the Chi–squared test results,

which further denote the statistical significance of this finding.

3.4.4 Signal Peptides

Signal peptides are short peptide sequences (usually 5–60 amino acids long) located at the N-terminus of a large number of newly synthesised proteins. These signal sequences control the targeting and translocation of the secreted or cell membrane proteins. Signal peptides have different structures, but all are described by three domains comprising a positively charged N–terminal region of 5–8 residues, followed by a region of hydrophobic residues (crucial for protein targeting) and a neutral polar C–region. Signal peptides mediate the protein translocation across the ER membrane, where the C–region sequence motif is recognised by a signal recognition particle (SRP) and cleaved off the protein by the signal peptidase enzyme; this cleaved sequence serves as the signal peptide. Signal peptides direct proteins to different cellular locations (*e.g.* nucleus, mitochondria, endoplasmic reticulum, endosome, Golgi apparatus) where proteins can carry out their functions.

Our analysis with the signal peptide (computed using UniProt annotation and SignalP servers (Petersen et al., 2011)) demonstrated that signal peptide motifs are more frequent in proteins encoded by viable genes compared to proteins encoded by lethal genes. The estimated p–value < 0.05 from the Chi–squared test, which are summarised in Table 3.10, further confirmed these differences are statistically significant.

TABLE 3.10: Signal peptide count in lethal and viable proteins and the corresponding p-values computed using the Chi-squared test. Highlighted cells in yellow represent statistically significant results; blue cells refer to viable proteins where the signal peptide is observed as more frequent.

Datasets	**Lethal**	**Viable**	**Lethal(%)**	**Viable(%)**	**p–value**
Non–culled	213	1004	16.37	29.09	1.23×10^{-19}
Culled(20%)	67	304	13.98	29.89	8.26×10^{-09}
Culled(40%)	151	698	15.71	30.32	8.84×10^{-14}
Culled(60%)	200	941	16.46	30.29	1.77×10^{-15}
Culled(20%)	210	993	16.27	29.28	4.08×10^{-15}

3.4.5 Transmembrane Domains

Integral membrane proteins are a form of membrane protein that are embedded in the cell membrane. Most of these integral membrane proteins are transmembrane proteins that extend through the lipid bilayer and span from the interior to the exterior of the cell. Transmembrane proteins usually adopt a α–helical structure while passing through the lipid bilayer one (single–pass proteins) or multiple times (multiple–pass proteins). These helical segments that cross the lipid bilayer are hydrophobic.

The hydrophilic regions of transmembrane proteins, located on either side of the membrane, are exposed to water. Due to this structure, transmembrane proteins can mediate cellular functions both inside and outside of the cell. Transmembrane proteins are important for cell–cell communication, maintenance of cell structure, signalling, and ion transport. Many receptor proteins have a number of α–helical transmembrane domains spanning the cell membrane. Thus, the presence of transmembrane domains in proteins encoded by lethal and viable genes could be informative for functional annotation.

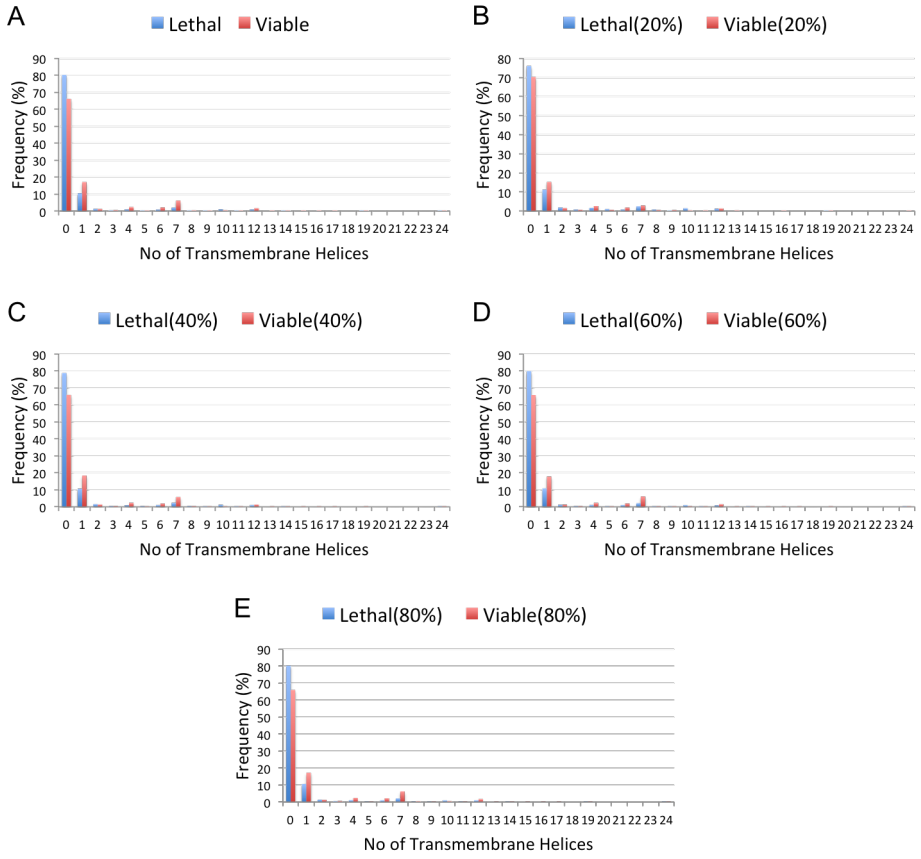


FIGURE 3.19: Distributions of the number of transmembrane helices between lethal and viable proteins in the non–culled (A) and culled datasets (B-E)

We found that our total viable dataset is significantly enriched in transmembrane proteins (p–value = 1.9×10^{-15}). Approximately 20% of total lethal proteins are found as transmembrane proteins, whereas the corresponding percentage is approximately 34% for viable proteins. In addition, distributions of the number of transmembrane helices between lethal and viable mouse proteins demonstrated that viable transmembrane proteins in the non–culled dataset tend to have more transmembrane helices than lethal proteins (p–value = 2.6×10^{-21}). The same trend was true for all culled datasets and the differences were statistically significant, with p–values (computed using the Mann–Whitney U test) of 0.029,

2.7×10^{-12}, 2.1×10^{-19} and 7.8×10^{-22} for 20%, 40%, 60% and 80% culled datasets, respectively. Figure 3.19 displays the enrichment of viable proteins (for non–culled and culled datasets) in a higher number of transmembrane domains.

3.5 Analysis of GO Terms

Gene Ontology (GO) (Ashburner et al., 2000) is the most widely used scheme for classifying gene functions. The GO consortium provides a set of controlled vocabularies (ontology) to annotate the functional properties of gene and gene products across all species. Gene functions are annotated by means of three aspects: (a) molecular function (b) cellular component and (c) biological process. Molecular functions refer to the biochemical activities of a gene, such as binding or signal transducer activity. Cellular components correspond to different parts within a cell where a gene product functions (*e.g.* nucleus, organelle, plasma membrane, extracellular region). Biological processes refer to a set of molecular events regulated by one or more genes (*e.g.* cell division, DNA replication). This section highlights results of the study done with different categories of GO annotations to test whether their distributions can vary between lethal genes and viable genes.

3.5.1 Cellular Components

Eukaryotic cells are subdivided into several membrane-bound compartments, which are functionally distinct. These include nucleus, cytoplasm, cytoskeleton, extracellular space, plasma membrane, endoplasmic reticulum (ER), mitochondria, Golgi

apparatus, lysosome, peroxisome, and vacuoles. Protein functions are closely related to the locations where they reside within a cell. Localisation information is already known to be crucial for predicting essential genes in prior studies (Gustafson et al., 2006; Seringhaus et al., 2006; Acencio and Lemke, 2009). As an example, eukaryotic proteins located in the nucleus are found to carry out essential functions including DNA replication, mRNA synthesis and recombination. Subcellular locations, therefore, should be useful in predicting mouse essential genes in the current study.

As mentioned in the method section (section 2.2.3), all GO terms were extracted from the DAVID functional annotation tool version 6.8 (Huang et al., 2007) by submitting Ensembl IDs of mouse lethal and viable genes. A total of 225 cellular component GO terms for lethal genes and 149 terms for viable genes were retrieved, of which 53 and 82 terms were found significant, utilising the Bonferroni corrected p–value ≤ 0.05 from the functional annotation output of DAVID. Table 3.11 and 3.12 summarise these cellular component GO terms favoured for lethal and viable genes, respectively. Results showed that majority of lethal genes are intracellular. Terms most frequently associated with lethal genes include: 'nucleus', 'transcription factor complex', 'nucleoplasm', 'nucleolus', and 'intracellular membrane–bounded organelle'. 57% of total lethal genes were found to be present in the nucleus.

However, viable genes were mainly found as membrane bound. Most of the viable genes were enriched for cellular component terms including 'membrane',

TABLE 3.11: Top 45 cellular component GO terms associated with lethal mouse genes. Information is generated using the functional annotation tool of David.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0005634	nucleus	747	57.77	9.7×10^{-95}
GO:0005667	transcription factor complex	104	8.04	1.4×10^{-50}
GO:0005654	nucleoplasm	309	23.89	2.7×10^{-50}
GO:0005737	cytoplasm	682	52.74	9.4×10^{-46}
GO:0043234	protein complex	126	9.74	1.4×10^{-27}
GO:0005829	cytosol	239	18.48	5.9×10^{-22}
GO:0000790	nuclear chromatin	64	4.94	1.4×10^{-20}
GO:0005925	focal adhesion	77	5.95	5.6×10^{-15}
GO:0048471	perinuclear region of cytoplasm	103	7.96	4.9×10^{-12}
GO:0009986	cell surface	97	7.50	5.4×10^{-12}
GO:0005911	cell-cell junction	45	3.48	6.6×10^{-10}
GO:0043025	neuronal cell body	80	6.18	4.9×10^{-09}
GO:0043005	neuron projection	68	5.25	7.0×10^{-09}
GO:0005730	nucleolus	107	8.27	8.3×10^{-08}
GO:0030424	axon	60	4.64	1.1×10^{-07}
GO:0000785	chromatin	31	2.39	5.0×10^{-07}
GO:0030054	cell junction	90	6.96	2.0×10^{-06}
GO:0005694	chromosome	54	4.17	2.8×10^{-06}
GO:0043231	intracellular membrane-bounded organelle	93	7.19	3.0×10^{-06}
GO:0017053	transcriptional repressor complex	20	1.54	3.4×10^{-06}
GO:0045121	membrane raft	45	3.48	7.8×10^{-06}
GO:0090575	RNA polymerase II transcription factor complex	15	1.16	8.0×10^{-06}
GO:0005794	Golgi apparatus	129	9.97	9.9×10^{-06}
GO:0030027	lamellipodium	33	2.55	1.7×10^{-05}
GO:0016323	basolateral plasma membrane	37	2.86	3.5×10^{-05}
GO:0005657	replication fork	11	0.85	1.1×10^{-04}
GO:0005856	cytoskeleton	118	9.12	1.2×10^{-04}
GO:0005938	cell cortex	29	2.24	1.2×10^{-04}
GO:0043235	receptor complex	28	2.16	1.3×10^{-04}
GO:0070062	extracellular exosome	241	18.63	1.6×10^{-04}
GO:0005913	cell-cell adherens junction	47	3.63	1.8×10^{-04}
GO:0016605	PML body	23	1.77	2.2×10^{-04}
GO:0005819	spindle	25	1.93	3.2×10^{-04}
GO:0016580	Sin3 complex	9	0.69	5.6×10^{-04}
GO:0014704	intercalated disc	16	1.23	5.7×10^{-04}
GO:0005912	adherens junction	16	1.23	1.3×10^{-03}
GO:0005813	centrosome	56	4.33	1.7×10^{-03}
GO:0032993	protein-DNA complex	12	0.92	1.8×10^{-03}
GO:0000781	chromosome, telomeric region	18	1.39	2.8×10^{-03}
GO:0030426	growth cone	28	2.16	3.6×10^{-03}
GO:1990909	Wnt signalosome	7	0.54	4.5×10^{-03}
GO:0014069	postsynaptic density	36	2.78	4.9×10^{-03}
GO:0016363	nuclear matrix	21	1.62	5.0×10^{-03}
GO:0005876	spindle microtubule	13	1.01	6.3×10^{-03}
GO:0005790	smooth endoplasmic reticulum	11	0.85	7.1×10^{-03}

TABLE 3.12: Top 45 cellular component GO terms associated with viable mouse genes. Information is generated using the functional annotation tool of David.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0016020	membrane	1794	52.30	1.7×10^{-111}
GO:0005886	plasma membrane	1298	37.84	2.1×10^{-79}
GO:0009986	cell surface	297	8.66	1.3×10^{-65}
GO:0005887	integral component of plasma membrane	426	12.42	8.7×10^{-62}
GO:0043025	neuronal cell body	249	7.26	7.3×10^{-54}
GO:0005615	extracellular space	503	14.66	4.2×10^{-52}
GO:0005576	extracellular region	549	16.01	5.4×10^{-49}
GO:0005829	cytosol	575	16.76	1.2×10^{-47}
GO:0009897	external side of plasma membrane	168	4.90	1.6×10^{-43}
GO:0045202	synapse	221	6.44	5.5×10^{-42}
GO:0030425	dendrite	206	6.01	1.2×10^{-35}
GO:0030424	axon	171	4.99	1.3×10^{-35}
GO:0045121	membrane raft	136	3.97	1.0×10^{-33}
GO:0043005	neuron projection	182	5.31	3.0×10^{-33}
GO:0005737	cytoplasm	1448	42.22	7.6×10^{-30}
GO:0070062	extracellular exosome	692	20.17	8.9×10^{-30}
GO:0016324	apical plasma membrane	145	4.23	1.3×10^{-26}
GO:0030054	cell junction	245	7.14	2.5×10^{-26}
GO:0048471	perinuclear region of cytoplasm	237	6.91	4.7×10^{-25}
GO:0045211	postsynaptic membrane	108	3.15	8.7×10^{-24}
GO:0014069	postsynaptic density	113	3.29	1.2×10^{-23}
GO:0016323	basolateral plasma membrane	102	2.97	1.6×10^{-23}
GO:0043197	dendritic spine	81	2.36	2.9×10^{-21}
GO:0043679	axon terminus	56	1.63	1.0×10^{-18}
GO:0043235	receptor complex	70	2.04	1.4×10^{-16}
GO:0043195	terminal bouton	61	1.78	2.8×10^{-16}
GO:0043234	protein complex	195	5.69	2.6×10^{-15}
GO:0043204	perikaryon	71	2.07	1.1×10^{-14}
GO:0042734	presynaptic membrane	47	1.37	1.2×10^{-14}
GO:0008021	synaptic vesicle	62	1.81	1.0×10^{-12}
GO:0005764	lysosome	115	3.35	7.0×10^{-12}
GO:0042383	sarcolemma	57	1.66	2.7×10^{-11}
GO:0005578	proteinaceous extracellular matrix	107	3.12	5.2×10^{-10}
GO:0043198	dendritic shaft	36	1.05	1.6×10^{-09}
GO:0005791	rough endoplasmic reticulum	37	1.08	3.3×10^{-08}
GO:0005901	caveola	42	1.22	6.2×10^{-08}
GO:0030141	secretory granule	49	1.43	1.7×10^{-07}
GO:0031225	anchored component of membrane	57	1.66	1.9×10^{-07}
GO:0005783	endoplasmic reticulum	317	9.24	2.2×10^{-07}
GO:0005768	endosome	153	4.46	2.2×10^{-07}
GO:0031012	extracellular matrix	95	2.77	2.4×10^{-07}
GO:0001750	photoreceptor outer segment	35	1.02	3.1×10^{-07}
GO:0031234	extrinsic component of cytoplasmic side of plasma membrane	35	1.02	5.2×10^{-07}
GO:0042995	cell projection	184	5.36	1.9×10^{-06}
GO:0001917	photoreceptor inner segment	28	0.82	1.9×10^{-06}

'plasma membrane', 'cell surface', 'extracellular region', 'extracellular space', and 'lysosome'. A high percentage of lethal (52.7%) and viable (42.2%) genes were also found being localised in the cytoplasm.

Subcellular locations were also analysed using the UniProt annotation and

TABLE 3.13: Subcellular locations of all lethal and viable mouse proteins as annotated in the UniProt database. p-values were computed using the Chi–squared test. Here, the Bonferroni corrected p–value = 0.0041.

Cellular Components	**Lethal**	**Viable**	**Lethal(%)**	**Viable(%)**	**p–value**
Nucleus	627	815	48.19	23.62	8.37×10^{-43}
Cytoplasm	433	1014	33.28	29.38	0.029
Plasma membrane	170	805	13.07	23.33	3.35×10^{-12}
Membrane (exculding plasma)	117	545	8.99	15.79	2.15×10^{-8}
Extracellular	95	504	7.30	14.60	2.58×10^{-10}
Mitochondrion	67	145	5.15	4.20	0.167
Endoplasmic Reticulum (ER)	70	192	5.38	5.56	0.810
Golgi	62	150	4.77	4.35	0.542
Lysosome	10	80	0.77	2.32	5.38×10^{-4}
Peroxisome	5	22	0.38	0.64	0.301
Cell Junction	78	199	6.00	5.77	0.770
Cell Projection	47	130	3.61	3.77	0.805

the WoLF PSORT tool (Horton et al., 2007) (see section 2.2.2.5). Table 3.13 summarises the results of the UniProt analysis. We found that a significantly higher proportion of viable proteins are likely to be localised in the plasma membrane (23%), membrane (15%) and extracellular region (14%) compared to lethal proteins. However, there is a notably high percentage of lethal proteins found within the nucleus (48%) compared to 23% of viable proteins. All these results are statistically significant (p–value ≤ 0.05) and we observed the same trend for culled datasets.

Subcellular location prediction results from WoLF PSORT are summarised in Table 3.14. In this case, the most significant enrichment for lethal proteins was again nucleus. We observed that 70% of total lethal proteins are located in the nucleus compared to 49% of viable proteins. A high percentage of lethal proteins were also found in the cytoplasm, but this result was not statistically significant for the analysis done with the UniProt annotation. The analysis of

TABLE 3.14: Subcellular locations of all lethal and viable mouse proteins, which were predicted by WoLF PSORT. p-values were computed using the Chi-square test. Here, the Bonferroni corrected p–value = 0.0056.

Cellular Components	Lethal	Viable	Lethal(%)	Viable(%)	p–value
Nucleus	921	1712	70.79	49.61	2.18×10^{-18}
Cytoplasm	700	1556	53.80	45.09	1.01×10^{-4}
Plasma membrane	307	1261	23.60	36.54	4.33×10^{-12}
Extracellular	353	1377	27.13	39.90	7.78×10^{-11}
Mitochondrion	321	890	24.67	25.79	0.496
Endoplasmic Reticulum (ER)	183	621	14.07	17.99	3.32×10^{-3}
Golgi	45	156	3.46	4.52	0.112
Lysosome	86	398	6.61	11.53	2.12×10^{-6}
Peroxisome	204	623	15.68	18.05	0.080

WoLF PSORT prediction results further confirmed the preferences for viable genes to be membrane bound (36%) and extracellular (39%). Viable proteins were also enriched for localisation to the endoplasmic reticulum (18%) and lysosome (11%) as compared to lethal proteins.

All these analyses with cellular localisations indicate that a major compartment for localisation of lethal proteins is the nucleus, whereas viable proteins are more likely to be extracellular or membrane bound. Viable proteins are also more likely to be located in the lysosome.

3.5.2 Biological Processes

A number of GO terms related to biological processes were examined in this study analysing the functional annotation output of DAVID. A total of 1,575 biological process terms were retrieved for lethal genes, with 1,777 terms for viable genes, of which 323 terms for lethal and 315 terms for viable datasets were significant meeting the Bonferroni corrected p–value ≤ 0.05. Table 3.15 summarises the top

TABLE 3.15: Top 45 preferred GO terms for lethal mouse genes that are related to biological processes. Information is retrieved using the functional annotation tool of DAVID.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0045944	positive regulation of transcription from RNA polymerase II promoter	294	22.74	3.3×10^{-105}
GO:0001701	in utero embryonic development	156	12.06	7.3×10^{-95}
GO:0045893	positive regulation of transcription, DNA-templated	202	15.62	7.0×10^{-84}
GO:0006351	transcription, DNA-templated	365	28.23	3.5×10^{-75}
GO:0000122	negative regulation of transcription from RNA polymerase II promoter	217	16.78	5.9×10^{-75}
GO:0007507	heart development	130	10.05	5.4×10^{-74}
GO:0007275	multicellular organism development	255	19.72	1.5×10^{-71}
GO:0006355	regulation of transcription, DNA-templated	398	30.78	1.4×10^{-69}
GO:0010628	positive regulation of gene expression	112	8.66	1.0×10^{-33}
GO:0008284	positive regulation of cell proliferation	132	10.21	1.9×10^{-33}
GO:0043066	negative regulation of apoptotic process	135	10.44	4.4×10^{-33}
GO:0009887	organ morphogenesis	55	4.25	1.7×10^{-29}
GO:0045892	negative regulation of transcription, DNA-templated	131	10.13	3.6×10^{-29}
GO:0006357	regulation of transcription from RNA polymerase II promoter	105	8.12	7.9×10^{-29}
GO:0009952	anterior/posterior pattern specification	55	4.25	1.0×10^{-28}
GO:0001525	angiogenesis	78	6.03	4.6×10^{-27}
GO:0008285	negative regulation of cell proliferation	100	7.73	1.4×10^{-26}
GO:0003007	heart morphogenesis	41	3.17	1.6×10^{-26}
GO:0001568	blood vessel development	42	3.25	2.7×10^{-25}
GO:0001570	vasculogenesis	41	3.17	4.7×10^{-25}
GO:0001947	heart looping	39	3.02	1.3×10^{-24}
GO:0060021	palate development	45	3.48	1.3×10^{-24}
GO:0009790	embryo development	42	3.25	8.1×10^{-23}
GO:0030154	cell differentiation	143	11.06	1.0×10^{-22}
GO:0030324	lung development	51	3.94	2.3×10^{-21}
GO:0008283	cell proliferation	67	5.18	1.8×10^{-20}
GO:0060070	canonical Wnt signaling pathway	42	3.25	3.1×10^{-20}
GO:0045165	cell fate commitment	38	2.94	3.7×10^{-20}
GO:0001822	kidney development	51	3.94	6.0×10^{-20}
GO:0003151	outflow tract morphogenesis	33	2.55	2.4×10^{-19}
GO:0042475	odontogenesis of dentin-containing tooth	35	2.71	3.1×10^{-19}
GO:0030182	neuron differentiation	48	3.71	6.3×10^{-18}
GO:0030326	embryonic limb morphogenesis	34	2.63	9.0×10^{-18}
GO:0001889	liver development	44	3.40	1.6×10^{-17}
GO:0007389	pattern specification process	32	2.47	3.7×10^{-17}
GO:0007399	nervous system development	83	6.42	7.8×10^{-17}
GO:0001843	neural tube closure	40	3.09	1.6×10^{-16}
GO:0001666	response to hypoxia	58	4.49	1.9×10^{-16}
GO:0001707	mesoderm formation	25	1.93	2.4×10^{-16}
GO:0010629	negative regulation of gene expression	67	5.18	3.8×10^{-16}
GO:0010468	regulation of gene expression	73	5.65	4.4×10^{-16}
GO:0035115	embryonic forelimb morphogenesis	25	1.93	6.6×10^{-16}
GO:0001569	patterning of blood vessels	26	2.01	1.2×10^{-15}
GO:0001658	branching involved in ureteric bud morphogenesis	28	2.17	2.1×10^{-15}
GO:0030900	forebrain development	37	2.86	2.3×10^{-15}

45 biological process terms significantly favoured for lethal genes. Our analysis showed that lethal genes are often involved in different developmental processes, as expected, including 'in utero embryonic development', 'embryonic development', 'heart development', 'blood vessel development','nervous system development', 'brain development' and 'lung development'. Significant enrichment of the lethal

TABLE 3.16: Top 45 preferred GO terms for viable mouse genes that are related to biological processes. Information is retrieved using the functional annotation tool of DAVID.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0006954	inflammatory response	208	6.06	1.3×10^{-61}
GO:0002376	immune system process	219	6.38	3.8×10^{-61}
GO:0007165	signal transduction	427	12.45	2.3×10^{-38}
GO:0042493	response to drug	182	5.31	1.9×10^{-37}
GO:0032496	response to lipopolysaccharide	124	3.62	3.5×10^{-35}
GO:0043065	positive regulation of apoptotic process	158	4.61	8.9×10^{-29}
GO:0045087	innate immune response	170	4.96	1.2×10^{-24}
GO:0007204	positive regulation of cytosolic calcium ion concentration	90	2.62	1.4×10^{-24}
GO:0045944	positive regulation of transcription from RNA polymerase II promoter	327	9.53	1.8×10^{-24}
GO:0042981	regulation of apoptotic process	105	3.06	1.9×10^{-23}
GO:0007568	aging	99	2.89	1.4×10^{-22}
GO:0006955	immune response	141	4.11	5.5×10^{-22}
GO:0006468	protein phosphorylation	212	6.18	8.2×10^{-22}
GO:0006915	apoptotic process	207	6.03	2.8×10^{-20}
GO:0019233	sensory perception of pain	56	1.63	3.5×10^{-20}
GO:0007155	cell adhesion	183	5.34	8.6×10^{-20}
GO:0006811	ion transport	207	6.03	2.6×10^{-19}
GO:0045471	response to ethanol	73	2.13	1.7×10^{-17}
GO:0006816	calcium ion transport	76	2.22	1.2×10^{-16}
GO:0002250	adaptive immune response	74	2.16	3.3×10^{-16}
GO:0007268	chemical synaptic transmission	86	2.51	4.5×10^{-16}
GO:0035556	intracellular signal transduction	151	4.40	3.9×10^{-15}
GO:0016310	phosphorylation	203	5.92	6.3×10^{-15}
GO:0042127	regulation of cell proliferation	100	2.92	1.0×10^{-14}
GO:0070374	positive regulation of ERK1 and ERK2 cascade	89	2.59	1.8×10^{-14}
GO:0010628	positive regulation of gene expression	148	4.31	1.9×10^{-14}
GO:0007613	memory	53	1.55	2.4×10^{-14}
GO:0001666	response to hypoxia	91	2.65	7.0×10^{-14}
GO:0071456	cellular response to hypoxia	60	1.75	7.8×10^{-14}
GO:0007200	phospholipase C-activating G-protein coupled receptor signaling pathway	42	1.22	2.6×10^{-13}
GO:0007166	cell surface receptor signaling pathway	95	2.77	2.6×10^{-13}
GO:0042632	cholesterol homeostasis	43	1.25	8.6×10^{-13}
GO:0050731	positive regulation of peptidyl-tyrosine phosphorylation	56	1.63	8.6×10^{-13}
GO:0032729	positive regulation of interferon-gamma production	38	1.11	8.6×10^{-13}
GO:0032355	response to estradiol	65	1.90	5.1×10^{-12}
GO:0008284	positive regulation of cell proliferation	178	5.19	9.4×10^{-12}
GO:0050728	negative regulation of inflammatory response	51	1.49	9.4×10^{-12}
GO:0008285	negative regulation of cell proliferation	138	4.02	1.5×10^{-11}
GO:0051384	response to glucocorticoid	52	1.52	1.9×10^{-11}
GO:0043066	negative regulation of apoptotic process	183	5.34	2.7×10^{-11}
GO:0007169	transmembrane receptor protein tyrosine kinase signaling pathway	54	1.57	6.7×10^{-11}
GO:0009612	response to mechanical stimulus	42	1.22	9.8×10^{-11}
GO:0051092	positive regulation of NF-kappaB transcription factor activity	58	1.69	1.2×10^{-10}
GO:0007193	adenylate cyclase-inhibiting G-protein coupled receptor signaling pathway	33	0.96	1.5×10^{-10}
GO:0050729	positive regulation of inflammatory response	41	1.20	1.6×10^{-10}

genes is also observed for processes related to 'transcription', 'cell proliferation', 'cell differentiation', 'organ morphogenesis', 'cell division', 'DNA repair', and 'DNA replication'. On the other hand, biological process terms favoured for viable genes include 'inflammatory response', 'signal transduction', 'ion transport', 'immune response', 'response to drug', 'response to stimulus', 'behaviour', 'transmembrane

transport', 'aging' and 'regulation of apoptotic process' (Table 3.16).

3.5.3 Molecular Functions

Molecular functions refer to the biochemical activities of a gene. A number of different GO terms related to molecular functions were analysed. Analysing the annotation output generated by DAVID, a total of 265 terms for lethal genes and 105 terms for viable genes were retrieved. Based on the Bonferroni corrected p–value ≤ 0.05, we considered 75 and 81 significant molecular function terms for lethal (Table 3.17) and viable (Table 3.18) datasets, respectively. We found that lethal genes are involved in 'DNA binding', 'transcription factor activity', 'transcription factor binding', and 'transferase activity'. Viable genes are more likely to have the annotations of 'signal transducer activity', 'ion channel activity', 'hydrolase activity', 'transporter activity', 'calcium ion binding', 'receptor binding', 'SH3 domain binding', and 'lipid binding'. A higher percentage of lethal and viable genes were also found to be annotated as being involved in 'protein binding', 'ATP binding', 'protein kinase binding', and 'protein kinase activity'.

3.6 Analysis of Protein Domains

Protein domains are spatially distinct structural and/or functional units of a protein. They carry out particular functions or interactions, thereby contributing towards the overall functionality of a protein. Proteins can have single or multiple domains. We obtained domain data for lethal and viable mouse protein analysing

TABLE 3.17: Preferred GO terms for lethal mouse genes that are related to molecular function. Information is retrieved using the functional annotation tool of DAVID.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0005515	protein binding	669	51.74	1.7×10^{-121}
GO:0003677	DNA binding	356	27.53	3.3×10^{-71}
GO:0043565	sequence-specific DNA binding	186	14.39	1.1×10^{-62}
GO:0003700	transcription factor activity, sequence-specific DNA binding	206	15.93	4.5×10^{-51}
GO:0003682	chromatin binding	131	10.13	8.1×10^{-40}
GO:0008134	transcription factor binding	108	8.35	5.4×10^{-37}
GO:0001077	transcriptional activator activity, RNA polymerase II core promoter proximal region	94	7.27	3.2×10^{-36}
GO:0000978	RNA polymerase II core promoter proximal region sequence-specific DNA binding	108	8.35	2.3×10^{-35}
GO:0044212	transcription regulatory region DNA binding	86	6.65	6.6×10^{-35}
GO:0046982	protein heterodimerization activity	109	8.43	3.3×10^{-21}
GO:0001228	transcriptional activator activity, RNA polymerase II transcription regulatory region	43	3.33	2.0×10^{-18}
GO:0001085	RNA polymerase II transcription factor binding	31	2.40	5.4×10^{-17}
GO:0019901	protein kinase binding	90	6.96	2.6×10^{-16}
GO:0032403	protein complex binding	79	6.11	7.5×10^{-16}
GO:0003705	transcription factor activity, RNA polymerase II distal enhancer	32	2.47	1.9×10^{-15}
GO:0019899	enzyme binding	80	6.19	3.2×10^{-14}
GO:0000979	RNA polymerase II core promoter sequence-specific DNA binding	31	2.40	3.6×10^{-14}
GO:0042826	histone deacetylase binding	38	2.94	1.2×10^{-12}
GO:0000977	RNA polymerase II regulatory region sequence-specific DNA binding	54	4.18	4.7×10^{-12}
GO:0042803	protein homodimerization activity	121	9.36	9.8×10^{-12}
GO:0008013	beta-catenin binding	32	2.47	1.2×10^{-11}
GO:0000981	RNA polymerase II transcription factor activity, sequence-specific DNA binding	46	3.56	3.1×10^{-11}
GO:0003713	transcription coactivator activity	47	3.63	4.1×10^{-11}
GO:0042802	identical protein binding	105	8.12	4.4×10^{-11}
GO:0000980	RNA polymerase II distal enhancer sequence-specific DNA binding	29	2.24	8.4×10^{-11}
GO:0046983	protein dimerization activity	47	3.63	2.2×10^{-10}
GO:0003690	double-stranded DNA binding	39	3.02	5.0×10^{-10}
GO:0001078	transcriptional repressor activity, RNA polymerase II core promoter proximal region	37	2.86	2.3×10^{-09}
GO:0001158	enhancer sequence-specific DNA binding	17	1.31	4.1×10^{-09}
GO:0001102	RNA polymerase II activating transcription factor binding	20	1.55	4.2×10^{-09}
GO:0046332	SMAD binding	23	1.78	4.6×10^{-09}
GO:0019904	protein domain specific binding	58	4.49	4.9×10^{-09}
GO:0031490	chromatin DNA binding	25	1.93	9.8×10^{-09}
GO:0004672	protein kinase activity	84	6.50	1.5×10^{-08}
GO:0016301	kinase activity	99	7.66	1.6×10^{-08}
GO:0005524	ATP binding	179	13.84	1.8×10^{-08}
GO:0016740	transferase activity	173	13.38	2.1×10^{-08}
GO:0002039	p53 binding	25	1.93	8.6×10^{-08}
GO:0001105	RNA polymerase II transcription coactivator activity	19	1.47	1.4×10^{-07}
GO:0047485	protein N-terminus binding	31	2.40	1.3×10^{-06}
GO:0003714	transcription corepressor activity	36	2.78	2.0×10^{-06}
GO:0019903	protein phosphatase binding	26	2.01	2.2×10^{-06}
GO:0070888	E-box binding	16	1.24	4.8×10^{-06}
GO:0046872	metal ion binding	323	24.98	6.2×10^{-06}
GO:0031625	ubiquitin protein ligase binding	51	3.94	7.1×10^{-06}

the functional annotation output of DAVID (mentioned in section 2.2.3). We observed a total of 11 and 30 domains from the Pfam protein domain database (Bateman et al., 2004) that are significantly enriched in lethal and viable proteins, respectively. Domains such as homeobox, T–box, helix–loop–helix DND–binding

TABLE 3.18: Preferred GO terms for viable mouse genes that are related to molecular function. Information is retrieved using the functional annotation tool of DAVID.

GO Term ID	GO Term Annotation	Count	%	Bonferroni
GO:0005515	protein binding	1242	36.21	3.3×10^{-93}
GO:0004871	signal transducer activity	259	7.55	1.4×10^{-33}
GO:0042803	protein homodimerization activity	293	8.54	1.8×10^{-30}
GO:0005216	ion channel activity	97	2.83	1.2×10^{-25}
GO:0005102	receptor binding	173	5.04	2.6×10^{-24}
GO:0019901	protein kinase binding	170	4.96	2.7×10^{-19}
GO:0004672	protein kinase activity	192	5.60	1.4×10^{-18}
GO:0046982	protein heterodimerization activity	188	5.48	8.1×10^{-18}
GO:0016301	kinase activity	221	6.44	9.6×10^{-16}
GO:0005125	cytokine activity	95	2.77	2.8×10^{-14}
GO:0042802	identical protein binding	215	6.27	3.1×10^{-14}
GO:0043565	sequence-specific DNA binding	198	5.77	4.1×10^{-11}
GO:0008083	growth factor activity	67	1.95	3.2×10^{-10}
GO:0004872	receptor activity	75	2.19	5.5×10^{-10}
GO:0002020	protease binding	58	1.69	6.5×10^{-10}
GO:0019899	enzyme binding	134	3.91	9.2×10^{-10}
GO:0008201	heparin binding	67	1.95	6.7×10^{-09}
GO:0008144	drug binding	54	1.57	4.4×10^{-08}
GO:0004896	cytokine receptor activity	30	0.87	4.8×10^{-08}
GO:0097110	scaffold protein binding	33	0.96	5.6×10^{-08}
GO:0005244	voltage-gated ion channel activity	59	1.72	8.1×10^{-08}
GO:0005516	calmodulin binding	72	2.10	2.9×10^{-07}
GO:0016787	hydrolase activity	384	11.20	3.5×10^{-07}
GO:0004713	protein tyrosine kinase activity	54	1.57	4.6×10^{-07}
GO:0017046	peptide hormone binding	25	0.73	9.3×10^{-07}
GO:0032403	protein complex binding	117	3.41	1.3×10^{-06}
GO:0044325	ion channel binding	53	1.55	1.6×10^{-06}
GO:0017124	SH3 domain binding	52	1.52	1.8×10^{-06}
GO:0005262	calcium channel activity	42	1.22	2.3×10^{-06}
GO:0005518	collagen binding	33	0.96	2.6×10^{-06}
GO:0005179	hormone activity	50	1.46	3.4×10^{-06}
GO:0004674	protein serine/threonine kinase activity	132	3.85	5.0×10^{-06}
GO:0016849	phosphorus-oxygen lyase activity	17	0.50	1.2×10^{-05}
GO:0004715	non-membrane spanning protein tyrosine kinase activity	27	0.79	1.6×10^{-05}
GO:0005178	integrin binding	45	1.31	1.9×10^{-05}
GO:0005524	ATP binding	367	10.70	4.0×10^{-05}
GO:0030165	PDZ domain binding	49	1.43	4.3×10^{-05}
GO:0005184	neuropeptide hormone activity	20	0.58	9.9×10^{-05}
GO:0016740	transferase activity	351	10.23	1.0×10^{-04}
GO:0046983	protein dimerization activity	68	1.98	1.1×10^{-04}
GO:0019900	kinase binding	41	1.20	1.4×10^{-04}
GO:0005164	tumor necrosis factor receptor binding	21	0.61	1.4×10^{-04}
GO:0004950	chemokine receptor activity	16	0.47	1.7×10^{-04}
GO:0004970	ionotropic glutamate receptor activity	15	0.44	2.2×10^{-04}
GO:0005234	extracellular-glutamate-gated ion channel activity	15	0.44	2.2×10^{-04}

TABLE 3.19: Key domains from the Pfam database that are enriched for proteins encoded by lethal mouse genes. Information is retrieved using the functional annotation tool of DAVID.

Term ID	Term Annotation	Count	%	Bonferroni
PF00046	Homeobox domain	63	4.87	1.9×10^{-17}
PF00010	Helix-loop-helix DNA-binding domain	28	2.17	4.8×10^{-07}
PF07714	Protein tyrosine kinase	29	2.24	5.0×10^{-05}
PF00110	wnt family	11	0.85	8.7×10^{-05}
PF00907	T-box	10	0.77	3.8×10^{-04}
PF00008	EGF-like domain	19	1.47	7.1×10^{-04}
PF00105	Zinc finger, C4 type (two domains)	14	1.08	7.7×10^{-03}
PF00688	TGF-beta propeptide	10	0.77	8.8×10^{-03}
PF00104	Ligand-binding domain of nuclear hormone receptor	14	1.08	1.3×10^{-02}
PF00019	Transforming growth factor beta like domain	12	0.93	1.8×10^{-02}
PF00069	Protein kinase domain	49	3.79	1.9×10^{-02}

TABLE 3.20: Key domains from the Pfam database that are enriched for proteins encoded by viable mouse genes. Information is retrieved using the functional annotation tool of DAVID.

Term ID	Term Annotation	Count	%	Bonferroni
PF00001	7 transmembrane receptor (rhodopsin family)	162	4.72	4.3×10^{-41}
PF00017	SH2 domain	56	1.63	2.2×10^{-14}
PF07714	Protein tyrosine kinase	65	1.90	1.1×10^{-12}
PF00520	Ion transport protein	53	1.55	2.7×10^{-10}
PF00018	SH3 domain	49	1.43	2.4×10^{-08}
PF00069	Protein kinase domain	121	3.53	4.8×10^{-08}
PF00104	Ligand-binding domain of nuclear hormone receptor	28	0.82	4.1×10^{-06}
PF13895	Immunoglobulin domain	32	0.93	6.6×10^{-06}
PF00105	Zinc finger, C4 type (two domains)	26	0.76	4.2×10^{-05}
PF10613	Ligated ion channel L-glutamate- and glycine-binding site	15	0.44	5.6×10^{-05}
PF00060	Ligand-gated ion channel	15	0.44	5.6×10^{-05}
PF01582	TIR domain	16	0.47	1.2×10^{-04}
PF00211	Adenylate and Guanylate cyclase catalytic domain	16	0.47	1.2×10^{-04}
PF00619	Caspase recruitment domain	17	0.50	2.0×10^{-04}
PF02931	Neurotransmitter-gated ion-channel ligand binding domain	23	0.67	4.2×10^{-04}
PF02932	Neurotransmitter-gated ion-channel transmembrane region	23	0.67	4.2×10^{-04}
PF00229	TNF(Tumour Necrosis Factor) family	14	0.41	7.6×10^{-04}
PF00020	TNFR/NGFR cysteine-rich region	15	0.44	1.3×10^{-03}
PF00041	Fibronectin type III domain	49	1.43	1.3×10^{-03}
PF00045	Hemopexin	16	0.47	1.7×10^{-03}
PF00102	Protein-tyrosine phosphatase	21	0.61	2.3×10^{-03}
PF00433	Protein kinase C terminal domain	18	0.52	2.3×10^{-03}
PF00595	PDZ domain (Also known as DHR or GLGF)	45	1.31	5.3×10^{-03}
PF00413	Matrixin	15	0.44	6.2×10^{-03}
PF01471	Putative peptidoglycan binding domain	14	0.41	1.1×10^{-02}
PF00005	ABC transporter	24	0.70	1.4×10^{-02}
PF00019	Transforming growth factor beta like domain	19	0.55	2.0×10^{-02}
PF00130	Phorbol esters/diacylglycerol binding domain (C1 domain)	24	0.70	2.1×10^{-02}
PF00230	Major intrinsic protein	10	0.29	3.1×10^{-02}
PF00664	ABC transporter transmembrane region	14	0.41	4.2×10^{-02}

domain, protein kinase domain and Zinc finger C4 type (zf–c4) domain (many of which are found in transcription factors) showed a higher preference for lethal proteins (Table 3.19). Domains including 7–transmembrane receptor, SH2, ion transport, Fibronectin type III domain (fn3), and SH3 (many of which are found in membrane proteins) were favoured for viable proteins (Table 3.20). Although viable proteins were annotated with having protein kinase and zf–c4 domains, these domains were more frequently found within lethal proteins.

3.7 Analysis of Protein–Protein Interactions

Protein–protein interactions (PPI) are intrinsic to almost all biological processes. Since the majority of proteins interact with each other to expedite accurate functionality, knowledge about their interactions is crucial to understand the molecular mechanisms of cellular processes. A prior study found significant differences in PPI network properties between the essential and non–essential genes of *S. cerevisiae* and *E. coli* (Hwang et al., 2009). Network-based attributes were also found to be fundamental to elucidate proteins activities within the cell in other studies (Coulomb et al., 2005; Yang et al., 2014). We, therefore, expected that the study of PPI networks would provide new insights into the essentiality of mouse proteins.

Mouse protein–protein interaction data was obtained from the I2D database (Brown and Jurisica, 2007), which is a database of known and predicted protein interactions for human, mouse, rat, fly, yeast and worm genomes. The PPI data

was examined with the intention of learning whether the lethal PPI networks differ in their network properties from their viable counterparts. We analysed both known and predicted mouse PPIs to assure high quality PPIs. As mentioned in section 2.2.4, two PPI networks namely *Known* (**K**) and *Known–Predicted* (**KP**) were constructed from all mouse PPIs. After removing self and duplicate interactions, the network lethal–**K** contained 3,988 protein nodes and 8,074 interactions; the network viable–**K** included 4,879 protein nodes and 9,624 interactions. The network lethal–**KP** consisted of 12,001 nodes and 73,426 interactions, whereas the corresponding numbers are 11,686 and 75,040 for the viable–**KP** network. We computed 9 network properties for each lethal and viable protein to recognise their importance in each of the PPI networks. Lethal and viable proteins were found to have differences in these network properties from our analyses. We could not estimate network properties for 403 (30%) lethal and 1,622 (47%) viable proteins in the PPI network **K** because no known interaction was found for them. For **KP** network, these numbers were 61 (4.69%) and 371 (10.75%), respectively. Hence, lethal proteins are more likely to participate in PPIs than viable proteins.

Our results demonstrated that lethal proteins have more interactions (higher degrees) than viable proteins in both **K** (p–value $= 8.2 \times 10^{-16}$) and **KP** (p–value $= 4.1 \times 10^{-63}$) interaction networks (Figure 3.20). The mean degree of lethal proteins was higher than viable proteins for **K** (10.47 versus 6.41) and **KP** (57.68 versus 27.97). The average shortest path (ASP) length is an indicator of a protein node's efficiency in transporting information on a PPI network (mentioned in section

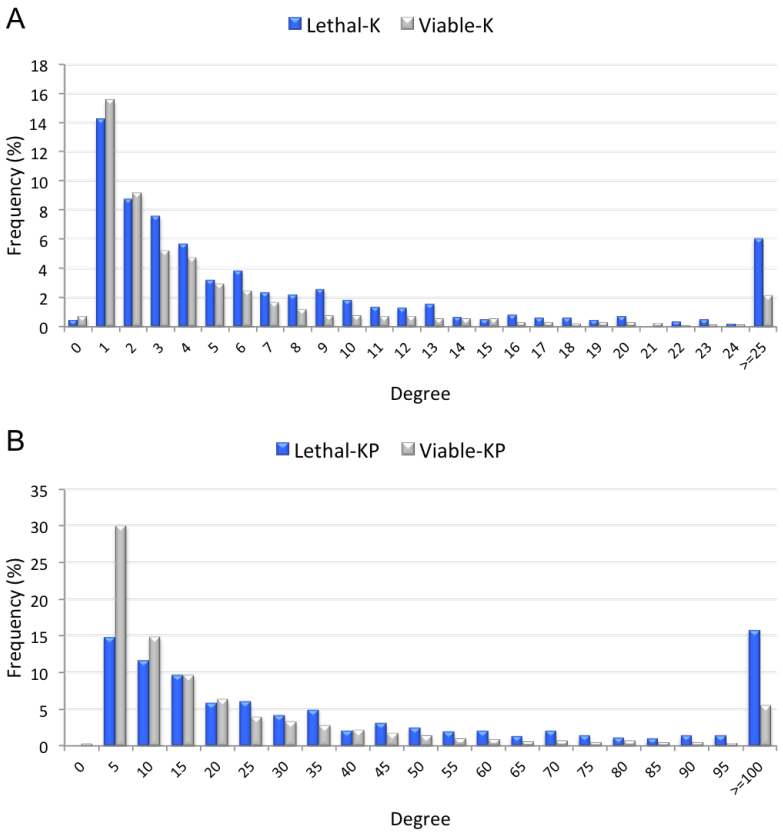


FIGURE 3.20: Degree distributions of lethal and viable proteins involved in the *Known* (A) and *Known–Predicted* (B) PPI networks. The bin size is 5 in (B)

2.2.4). We observed that the ASP length of lethal proteins tend to be shorter than the ASP length of viable proteins (Figure 3.21). The Mann–Whitney U test further confirmed that this difference is statistically significant for both **K** (p–value $= 8.6 \times 10^{-26}$) and **KP** (p–value $= 1.2 \times 10^{-260}$) networks. The betweenness centrality is an indicator of the centrality of a protein node in the PPI network. Our analysis demonstrated that the betweenness centrality of lethal proteins in each of the interaction network is significantly higher than that of viable proteins with p–values of 1.9×10^{-15} and 3.2×10^{-12}, respectively (Figure 3.22).

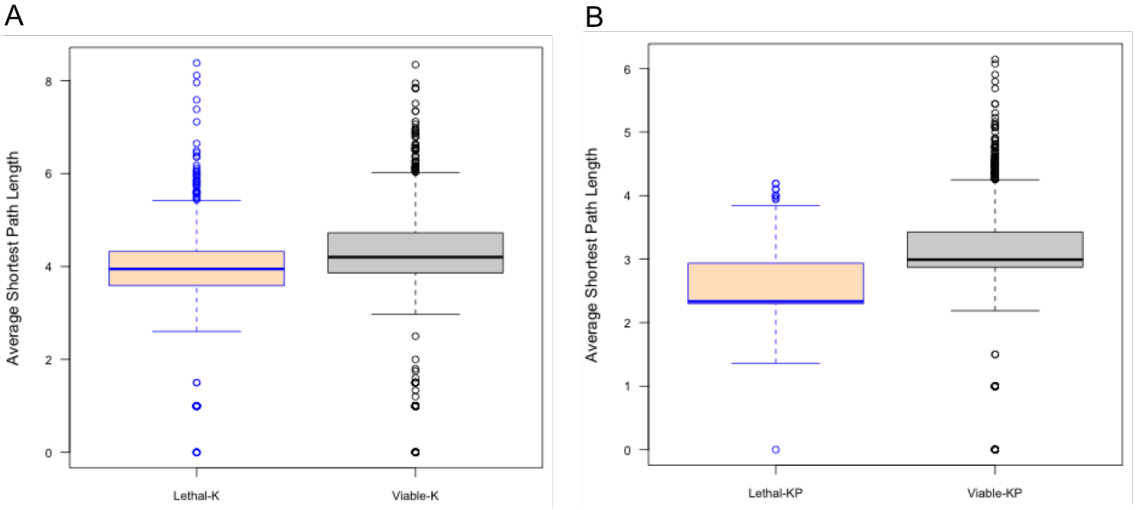


FIGURE 3.21: Length of average shortest path (ASP) of lethal and viable proteins in the *Known* (A) and *Known–Predicted* (B) protein–protein interaction (PPI) networks.

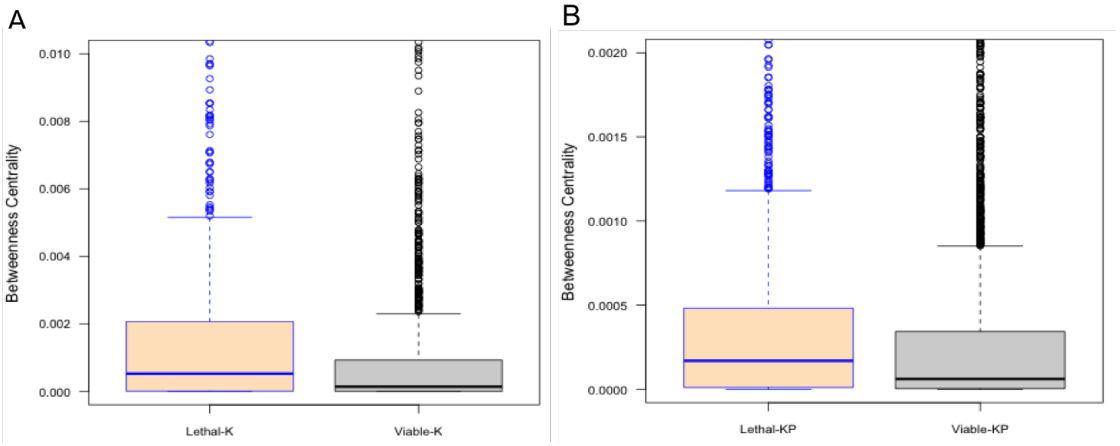


FIGURE 3.22: Betweenness centrality of lethal and viable proteins in the *Known* (A) and *Known–Predicted* (B) protein–protein interaction (PPI) networks. In each box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

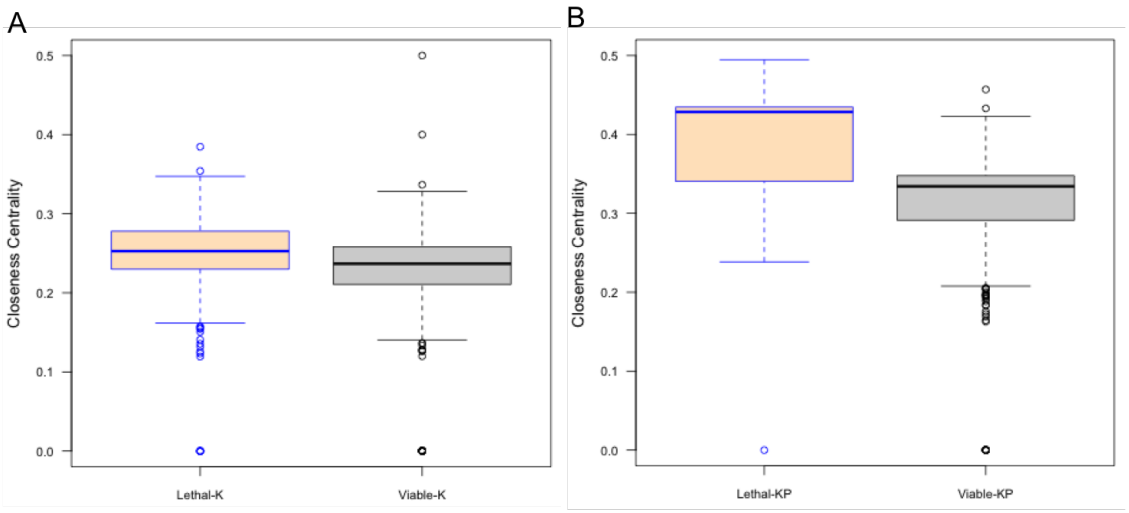


FIGURE 3.23: Closeness centrality of lethal and viable proteins in the *Known* (A) and *Known–Predicted* (B) protein–protein interaction (PPI) networks. In each box plot, the top and bottom of the box denote the upper and lower quartiles; the line inside the box denotes the middle quartile or the median; and individual points denote the outliers.

We also found significantly higher clustering coefficient values for lethal proteins in **K** (p–value $= 9.7 \times 10^{-4}$) and **KP** (p–value $= 1.2 \times 10^{-39}$) networks compared to viable proteins. Furthermore, subsequent analyses demonstrated that lethal proteins tend to have significantly high closeness centrality than viable proteins (Figure 3.23). This difference was statistically significant for both networks with p–values of 1.3×10^{-28} and 5.8×10^{-266}.

We also wanted to identify protein nodes with a large number of interactions (hubs) in the PPI network. Cytoscape does not provide this functionality, so we used the Hub object Analyser (Hubba) (Lin et al., 2008) to explore four additional network properties including BottleNeck (BN), Edge Percolation Component (EPC), Maximum Neighbourhood Component (MNC) and Density of Maximum Neighbourhood Component (DMNC). These properties define probable

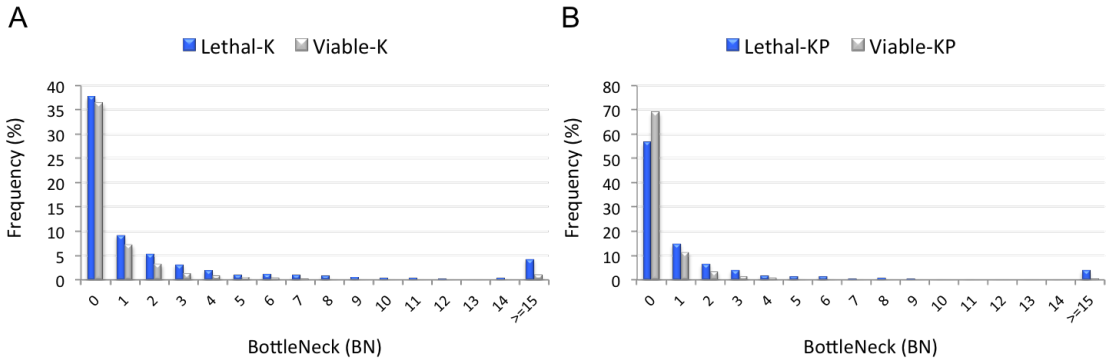


FIGURE 3.24: BottleNeck (BN) of lethal and viable proteins in the *Known* (A) and *Known–Predicted* (B) protein–protein interaction (PPI) networks.

hubs in the PPI network. Our investigation demonstrated that lethal proteins tend to have high BN in both **K** and **KP** networks (Figure 3.24). We further found that EPC and MNC of lethal proteins are significantly higher than that of viable genes. Table 3.21 shows the statistical significance of these differences. Although lethal proteins exhibited high DMNC in the **K** network, the same trend was not observed for the **KP** network.

TABLE 3.21: Distributions of four network properties including BottleNeck (BN), Edge Percolation Component (EPC), Maximum Neighbourhood Component (MNC) and Density of Maximum Neighbourhood Component (DMNC) between lethal and viable proteins. The Bonferroni corrected p–value in the Mann–Whitney U test is 0.0125. Here, mean rank, a parameter of the Mann–Whitney U test, indicates which protein group holds higher values for a network property.

Network		Network Properties			
		BN	EPC	MNC	DMNC
Known(K)	Lethal (Mean Rank)	1521.85	1519.73	1485.79	1441.51
	Viable (Mean Rank)	1286.50	1287.54	1304.21	1325.95
	p–value	1.02×10^{-17}	4.58×10^{-13}	5.57×10^{-10}	7.20×10^{-05}
Known–Predicted(KP)	Lethal (Mean Rank)	2467.60	2682.56	2685.01	2146.61
	Viable (Mean Rank)	2036.86	1950.32	1949.33	2166.09
	p–value	9.05×10^{-39}	3.07×10^{-68}	2.42×10^{-69}	0.641

3.8 Discussion

Essential genes are crucial for organism's survivable and development. The ability to determine essential genes in mammals is one of the fundamental aspects of development biology as it facilitates understanding of cellular, developmental and vital tissue-specific processes and functions. Existing experimental methods (Giaever et al., 2002; Cullen and Arndt, 2005; Roemer et al., 2003; Gallagher et al., 2007) are accurate for identifying mammalian essential genes, but most of them are not always feasible to execute due to massive investment of resources and time. Computational approaches, which exploit gene properties to characterise essentiality, are a fast and low cost alternative to these conventional experimental techniques.

Our research is based on the hypothesis that mammalian essential (lethal) and non-essential (viable) genes are distinguishable by their properties. If all properties are similar between lethal and viable datasets, then these cannot be of value in determining gene essentiality. Our dataset (non–culled) contained a total of 1,301 lethal and 3,451 viable mouse genes, which were obtained from the Mouse Genome Informatics (MGI) database considering the knockout mouse phenotypes. The presence of multiple copies of similar proteins could bias our overall analysis; we thereby remove redundant proteins from our dataset and generated non–redundant or culled datasets. Our aim was to assure that the key features differentiating lethal and viable genes are not over-representative though it remains likely that these hallmark features follow similar trends for both non–culled and culled datasets.

We studied a wide range of gene and protein properties of *Mus musculus* genes that are representative of different aspects of mouse biology so that we could quantify their abilities to differentiate lethal genes from viable genes. Some of these features have been found to be correlated with essentiality in yeast or bacteria in prior studies but have not been studied in mammals. Our investigation was focused on those features that are easily attainable from existing databases and web-based tools. These properties fall into three categories: (1) genomic properties, which are based on gene sequence data. This group also include features like evolutionary age and gene expression; (2) protein sequence properties, which can be estimated from protein sequence data. This covers features like amino acid composition, enzyme class, post-translational modifications, signal peptide and transmembrane domain; (3) functional properties, which facilitate biological interpretations of gene functionality. These include gene ontology (GO) annotations and protein–protein interactions (PPIs). Our investigation confirmed that many of these properties are strongly associated with essentiality. These features were also found to be interrelated, which implies that they are not independent of each other. We identified a total of 75 features that offer significant differences between lethal and viable genes (based on the p–value of statistical tests). These significant features are summarised as follows:

- **Genomic features:**
  - Feature based on gene sequence gene length, the number of transcripts, the number of exons, the length of exons and length of introns.

  - Evolutionary age.
  - Features based on gene expression expressions at 11 developmental stages including oocyte, unfertilized ovum, zygote, cleavage, morula, blastocyst, egg cylinder, gastrula, organogenesis, fetus and neonate.

- **Protein sequence features:** protein length, frequencies of residues (Alanine, Aspartic acid, Cysteine, Phenylalanine, Isoleucine, Glutamic acid, Lysine, Glutamine, Serine, Leucine, Valine, Tryptophan, polar, charged, basic, acidic, aliphatic, aromatic and non-polar), molecular weight, enzyme classes (Transferase, Ligase and Hydrolases), post-translational modifications (phosphoprotein, glycoprotein, acetylation and transcription), signal peptide and transmembrane helices.

- **Features based on Gene Ontology:** localisation at the nucleus, plasma membrane, extracellular region and lysosome, numerous terms linked to embryonic development and fundamental cellular processes including terms linked to embryonic development, transcription, cell morphogenesis, cell differentiation, immune response, cell communication, transporter activity and signal transducer activity.

- **Features based on protein–protein interaction (PPI) network topology:** degree, the average shortest path length, closeness centrality, betweenness centrality, clustering coefficient, BottleNeck and Maximum Neighbourhood Component in PPI networks.

3.8.1 Genomic Features and Gene Essentiality

We observed that mouse lethal genes are more likely to be longer in length, and have more transcripts than viable genes. Lethal genes also tend to exhibit more exons and have a longer exon length. These results are in agreement with a prior study (Budagyan and Loraine, 2004) which showed that longer genes with a large number of exons tend to exhibit a higher degree of alternative transcripts compared to smaller genes with fewer exons. These properties of lethal genes show that the functions they perform involve complex proteins having multiple domains and diverse cellular or tissue specialisations. Lethal genes also had a significantly longer length of introns and a low percentage of GC content compared to viable genes. A low GC content in lethal genes is consistent with a prior study (Gazave et al., 2007) which showed that intron length varies inversely with GC content. A recent study also claimed a strong negative correlation of GC content with exon and intron lengths in six mammalian genomes including human, chimpanzee, cow, dog, mouse and rat (Zhu et al., 2009). GC content is also correlated with gene length (Duret et al., 1995) and recombination (Montoya-Burgos et al., 2003) in mammalian genomes.

Gene expression data over 13 stages of mouse development showed that lethal genes are expressed in greater proportions at all early developmental stages compared to viable genes. This result makes sense because mouse genes which are expressed at early stages of development are more likely to be lethal as their disruption could affect all downstream events, thereby resulting in more severe

phenotypes. This result also agrees with prior studies which showed that housekeeping genes are likely to be highly expressed (Vinogradov, 2004; Liao and Zhang, 2006).

We also observed that lethal genes are older than viable genes. A significantly higher percentage of lethal genes have the evolutionary age of 1215 and 937 MYA. However, most of the viable genes are 400 MYA old. This result supports the notion that essential genes are evolutionarily more conserved than non–essential genes (Giaever et al., 2002; Jordan et al., 2002). This result is further explained by a previously reported observation that essential and highly expressed genes evolve slowly than non–essential genes (Drummond et al., 2005). Overall, this analysis indicates that older mouse genes are more likely to be indispensable for fundamental cellular processes. Lethal genes might be undergoing positive selection to retain their functionality, giving a lower mutation rate.

3.8.2 Protein Features and Gene Essentiality

We further checked whether gene essentiality might be correlated with different features derived from protein sequences. We found that proteins encoded by lethal genes tend to be longer in length and have greater molecular weight than proteins encoded by viable genes. This result is in agreement with a prior study which stated that functionally essential proteins are more evolutionarily conserved and conserved proteins are, in general, longer in length (Lipman et al., 2002). Longer

proteins may possibly have multiple domains to contribute diverse cellular functionalities (Brocchieri and Karlin, 2005) and our analysis with GO terms and protein domains further support this fact (see section 3.8.4).

Lethal proteins were found to have A, D, E, K, Q and S residues in great proportions. These residues are polar, which suggests that lethal proteins are unlikely to be located in membranes. In contrast, viable proteins have higher proportions of C, F, I, L, V and W residues. These residues are hydrophobic which indicates the tendency of viable proteins of being membrane proteins. In addition, C is often found in the SS bonds in extracellular proteins. The result of L residue follows a previous study which suggests that Leucine correlates negatively with the likelihood of being lethal (Yuan et al., 2012). The enrichment of K residues in lethal proteins agrees with our findings that they are likely to be more acetylated, as proteins are acetylated on lysine residues (Henriksen et al., 2012). The enrichment of acetylated proteins in lethal datasets implies that they are imperative for regulating protein–protein interactions, gene expression and metabolic processes.

Lethal proteins are likely to have more polar, charged, basic and acidic amino acids, whereas viable proteins have more aliphatic, aromatic and non-polar residues. A possible reason behind the enrichment of polar, charged and basic residues in lethal proteins is that they are less likely to be membrane proteins. Similarly, the presence of non-polar residues in high proportions for viable proteins is linked to their propensity to be located in membranes. Signal peptide motifs are found to be more frequent in viable proteins agreeing with our result

that viable proteins are more likely to be secreted (see section 3.8.3).

Lethal datasets are found being enriched in Ligase and Transferase among all classes of enzyme. This result is consistent with biological process terms analysis, as it reveals that lethal genes are greatly involved in regulating DNA replication, DNA repair and transferase activity. In contrast, the enrichment of Hydrolases in viable datasets makes sense, as Hydrolases are functionally less critical. Ligases also perform more complex chemistry than Hydrolases.

We demonstrated that lethal proteins are likely to be more phosphorylated as expected, as phosphoproteins are crucial for almost all cellular processes including cell differentiation, gene transcription and cell division (Puente et al., 2006). Our analysis with the GO terms of biological processes further confirmed the connection of lethal genes in controlling these fundamental processes. In addition, a greater number of N–glycosylated proteins in the viable datasets suggests their propensity to mediate cell–signalling following prior studies (Yan et al., 2002; Weng et al., 2013). The subcellular localisation analysis also confirms this result. Furthermore, the significant enrichment of the keyword 'transcription' in lethal datasets implies that a great proportion of lethal proteins function as transcription factors. Analysis of the GO–based annotations also established this result showing that lethal proteins are greatly engaged in gene transcription.

Our results further revealed the abundance of transmembrane proteins in the viable datasets. In addition, viable protein showed to have a greater number of transmembrane helices compared to lethal proteins. This result makes sense

because viable proteins tend to have a significantly high percentage of non–polar residues and a low percentage of polar residues, thereby are more likely to be hydrophobic. This result is further explained by their roles in cell communication, transport and signal transduction.

3.8.3 Differences in Subcellular Locations

Gene products are confined to different subcellular compartments to carry out their specific functions. Subcellular localisation can, therefore, provide useful information for gene essentiality. Localisation information has already recognised as a significant attribute for essential genes in bacteria and yeast (Gustafson et al., 2006; Seringhaus et al., 2006; Acencio and Lemke, 2009; Deng et al., 2010). Accordingly, our analysis demonstrated that proteins encoded by lethal genes are more likely to be intracellular. The majority of these proteins are located in the nucleus. This result conforms to the fact that almost one–third of the eukaryotic nuclear proteins are encoded by essential genes and are responsible for carrying out vital cellular processes like DNA replication, DNA repair and transcription (Kumar et al., 2002; Zhang and Zhang, 2008). Our analysis of biological processes further confirms this result. In addition, we found that lethal proteins are enriched for DND–binding domain, protein kinase domain, helix–loop–helix (HLH), homeobox, and Zinc finger, C4 type (zf–c4) protein domains. Many of these domains are found in transcription factors, which agrees with our finding that lethal genes

are nuclear localised. Proteins encoded by viable genes tend to be secreted (extracellular). The enrichment of signal peptide cleavage sites, fibronectin type III (fn3) domain and signal transducer activity conforms to this result. In addition, we found that a greater proportion of viable proteins are frequently located in membranes. Their enrichment in non-polar residues and contribution to transport activity further justifies this result. In addition, viable proteins are found being enriched for Src Homology 2 (SH2), Src Homology 2 (SH3), and ion transport domains. This further agrees with our finding that viable genes are membrane bound as these domains are mainly found in membrane proteins.

3.8.4 Differences in Biological Processes and Molecular Functions

We expected that the potential enrichment of a number of Gene Ontology (GO) (Ashburner et al., 2000) terms would be different between lethal and viable datasets. Our analysis showed that lethal genes are more likely to be involved in embryonic development, heart development, nervous system development, blood vessel development, brain development' and lung development. These cellular processes are indispensable for the progressive development of an embryo or fetus. This result is expected because lethal proteins are enriched in T–box domains, which are vital for heart development. In addition, we observed a significant enrichment of lethal genes in cell morphogenesis, cell division, cell proliferation, DNA replication, cell differentiation, DNA repair and transcription, which are crucial for

life. The presence of homeobox domain further confirms their vital role in morphogenesis. Moreover, the enrichment of protein kinase domain agrees with their involvement in transcription, cell differentiation and embryonic development. In contrast, viable genes were associated with cellular processes like ion transport, signal transduction, apoptosis, behaviour, and immune response.

Unlike viable genes, lethal genes showed DNA binding activity, transcription factor activity, transferase activity, transcription factor binding and ATP binding activity. The presence of HLH protein domains indicates their involvement in DNA binding activity. However, viable genes were found to be significantly linked to transporter activity, hydrolase activity, transmembrane transporter activity, ion channel activity, signal transducer activity, and receptor binding. This result makes sense because viable proteins are enriched in SH2 protein domains, which aids in signal transduction activity. Also, viable genes were greatly found for lipid binding activity. This is consistent with viable gene products tending to be transmembrane proteins.

3.8.5 PPI Networks and Essentiality

The correlation between PPI networks and gene essentiality has already been established in bacteria (Gustafson et al., 2006; Hwang et al., 2009; Deng et al., 2010), yeast (Chen and Xu, 2005; Saha and Heber, 2006; Gustafson et al., 2006; Hwang et al., 2009; Acencio and Lemke, 2009; Zhong et al., 2013), fly (Hahn and Kern, 2005), and human (Goh et al., 2007; Yang et al., 2014), but not in mouse.

The most significant properties of PPI networks that we found to be linked to mouse essential genes are: degree, average shortest path (ASP) length, betweenness centrality, closeness centrality and clustering coefficient. The enrichment of high degree proteins in the lethal dataset indicates that proteins encoded by mouse lethal genes are more likely to be hubs in the PPI network (He and Zhang, 2006). This result conforms to the fact that highly connected proteins or hubs tend to be essential and evolve slowly (Yu et al., 2004; Kim et al., 2006), and their absence disrupts cell viability (Jeong et al., 2001).

Lethal proteins having a significantly shorter length of ASP implies that they communicate with each other quickly in the PPI network. Lethal proteins with high betweenness centrality characterise their propensity to be bottlenecks in interaction networks. The link between lethal proteins and larger closeness centrality suggests that lethal genes can quickly transfer information in the PPI network. Furthermore, high values of clustering coefficient indicate that many of the interacting partners of lethal proteins are also interact with each other. In addition, we found significant enrichment of BN, EPC and MNC for lethal proteins. These results further justify that lethal proteins function as hubs in interacting networks. We conclude that essential genes in mouse play crucial roles in PPI networks.

3.9 Summary

We found a large number of features that show significant differences between lethal and viable mouse genes. To our best knowledge, this is the first study

where topological, biological and sequence-based gene properties have been systematically investigated in the mouse genome. These features are interrelated and they represent different aspects of mouse gene essentiality. Many of these features are found to be associated with essentiality in previous studies. In addition to that, we identified a number of novel features that also showed a strong connection with essentiality. These features include the number of transcripts, number of exons, exon length, intron length, post-translational modifications like phosphorylation, N–glycosylation, acetylation, transferase and ligase enzymatic function, developmental gene expression, and enrichment of numerous key cellular processes. These results validate our research hypothesis by showing that mouse essential and non-essential genes are distinguishable by various sequence and functional properties. We, therefore, suggest that these features could offer valuable insight into the mammalian gene essentiality. These features can further be used in developing a machine learning model, which may potentially enable us to predict mammalian essential genes with high precision.

Chapter 4

Mammalian Essential Gene Prediction

4.1 Introduction

Essential gene identification has already been achieved for various organisms through different experimental techniques namely single gene knockouts (Crawley, 1999; Giaever et al., 2002; Kobayashi et al., 2003), conditional knockouts (Liu et al., 2000; Roemer et al., 2003), RNA interference (Cullen and Arndt, 2005; Kamath et al., 2003), and transposon mutagenesis (Gallagher et al., 2007). Each of these experimental methodologies assaying gene essentiality is time–consuming and resource intensive, and also limited to few species. A complement to these existing experimental techniques is computational approaches, which have already shown their ability to predict essential genes accurately at a reduced effort and cost (Zhang et al., 2016).

Due to the availability of genome sequences and functional genomics data, previous studies deciphered the associations of different gene characteristics with the experimentally determined essential genes. These gene features were subsequently used in developing computational models to make reliable predictions of essential genes in worm, bacteria and yeast (Gustafson et al., 2006; Seringhaus et al., 2006; Hwang et al., 2009; Deng et al., 2010; Yang et al., 2014). In addition, a prior study showed the feasibility of predicting essential genes in mouse using their features (Yuan et al., 2012). However, this study had limited success and it failed to justify why these features are imperative for essentially. Accordingly, we assembled a wide range of sequence and functional properties of mouse genes from diverse data sources to further characterise lethal (essential) and viable (non–essential) genes in mouse, which should ultimately lead us to infer gene essentiality in human since mouse and human demonstrate high level of similarity in their genomes. We anticipated that various features would differentiate lethal and viable genes.

In Chapter 3, it was established that a large number of features including genomic features, protein sequence features, GO terms and protein–protein interaction (PPI) network features, differ significantly between lethal and viable genes in mouse. These features are interrelated and signify different aspects of mouse gene essentiality. If all features were identical within the lethal and viable datasets, then they could not be used to predict gene essentiality. Integrating these statistically significant features in developing a computational model should in turn

enable us to predict mammalian essential genes with high precision. We, therefore, aim to develop a machine learning classifier using these hallmark features to address to what extent mouse lethal and viable genes can be predicted from their sequence and functional attributes.

This chapter presents a number of Random Forest classifiers, which were developed and trained to predict whether a mouse gene better fits the profile of a lethal gene or viable gene on the basis of sequence and functional features. The selection of training and test datasets is discussed here. We evaluated the predictive power of our classifier based on cross–validation and observed high prediction accuracy. Further validation of our model performance was achieved by predicting lethal and viable genes in separate test datasets, confirming its ability to learn traits associated with gene essentiality. Moreover, the most relevant features were selected from the pool of all features by applying a feature selection method and the Random Forest classifier was further trained with these selected features. Feature selection also confirmed our classifier's capability in predicting gene essentially without compromising the high prediction accuracy.

4.2 Results

4.2.1 Datasets Generated

4.2.1.1 Features describing mouse genes in the datasets

In Chapter 3, we reported a large number of gene and protein features that exhibit significant differences between lethal and viable mouse genes. These gene properties were used to generate training and test datasets of our mammalian essential gene prediction classifier. We used 102 features with the corresponding class label to describe each mouse gene in the training and test datasets. These features are as follows:

- **Genomic features:**
  - Gene length; percentage of GC content; the number of transcripts; the number of exons; the length of exons and length of introns (section 2.2.1.1)
  - The oocyte, unfertilized ovum, zygote, cleavage, morula, blastocyst, egg cylinder, gastrula, organogenesis, fetus, neonate, juvenile and adult developmental stages expression level (section 2.2.1.2)
  - Evolutionary age based on most recent duplication (section 2.2.1.3)
- **Protein sequence features:**
  - Protein sequence length; the proportions of 20 amino acid residues; the proportions of polar, charged, basic, acidic, aliphatic, aromatic and non–polar residues; molecular weight (section 2.2.2.1)
  - Presence of six enzyme classes (section 2.2.2.2)

  - Presence of post–translational modifications (phosphoprotein, glycoprotein, acetylation) and transcription (section 2.2.2.2)
  - Presence of a signal peptide (section 2.2.2.3)
  - The number of transmembrane helices (section 2.2.2.4)
  - Localisation at the nucleus, cytoplasm, plasma membrane, extracellular region, Golgi apparatus, endoplasmic reticulum, membrane (excluding plasma), mitochondrion, peroxisome, lysosome, cell junction and cell projection; localisation score of nucleus, cytoplasm, plasma membrane, extracellular region, Golgi apparatus, mitochondrion, endoplasmic reticulum, mitochondria, peroxisome, and lysosome predicted by WoLF PSORT (section 2.2.2.5)

- **Protein–protein interaction (PPI) network based features:**
  - The number of interactions (degree), the length of average shortest path, betweenness centrality, closeness centrality, clustering coefficient, topological coefficient, BottleNeck, Edge Percolation Component (EPC), Maximum Neighbourhood Component (MNC) and Density of Maximum Neighbourhood Component (DMNC) for both *Known* (**K**) and *Known–Predicted* (**KP**) (section 2.2.4)

4.2.1.2 Training and testing datasets

Our original dataset containing 1,301 lethal and 3,451 viable mouse genes is an imbalanced dataset as the number of viable genes is much bigger than the number of lethal genes. Studies showed that imbalanced datasets degrade the classification performance of machine learning classifiers due to their bias towards classifying instances belonging to the majority class (Visa and Ralescu, 2005) . We, therefore, constructed three balanced training datasets containing equal number of lethal and

viable mouse genes, each consisting of 102 gene features. We could not find some features for a number of genes. Those missing entries were assigned the numeric value of -1. Each training and test dataset covers different subsets of lethal and viable genes as a result of random selection. Genes in test datasets were not included in training our classifier. The following strategies were used to generate balanced datasets.

- **Balanced training dataset (train–01) and balanced test dataset (test–b):** These training and test datasets consist of equal number of lethal and viable mouse genes. We created these datasets by including all the 1,301 lethal genes along with a randomly selected 1,301 viable genes. The training dataset is then created by randomly selecting 1,040 (80%) genes from each class. The remaining 261 (20%) genes from each class were used to create the test dataset. Figure 4.1 shows the overall workflow.

- **Balanced training dataset (train–02) and unbalanced test dataset (test–u01):** This training dataset contains equal number of lethal and viable mouse genes. However, the test dataset contains more viable genes compared to the number of lethal genes. We randomly selected 1,040 (80%) lethal genes from 1,301 lethal genes. All of these 1,040 lethal genes were included in the training dataset. The remaining 261 (20%) lethal genes were used to create the test dataset. We further randomly selected 2,760 (80%) genes from 3,451 viable genes. The remaining 691 (20%) viable genes were included in the test dataset. Moreover, for viable genes we selected 1,040

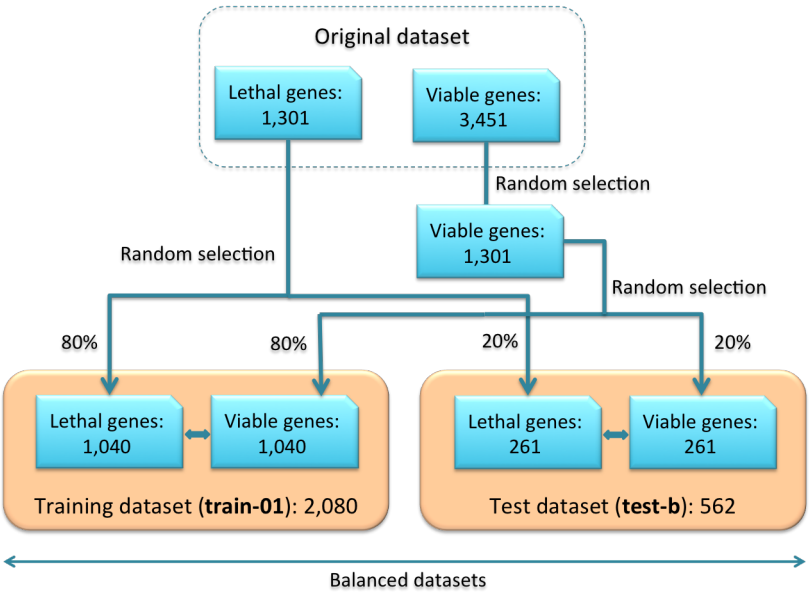


FIGURE 4.1: The workflow of generating the balanced **train–01** and **test–b** datasets.

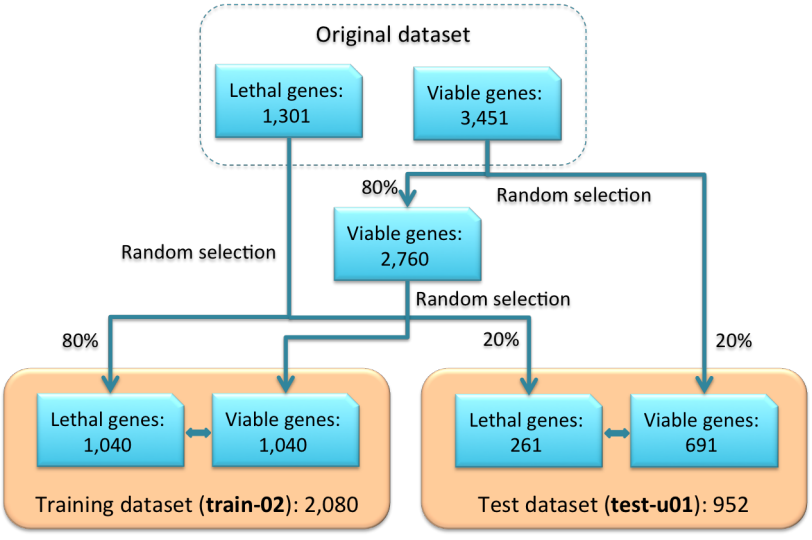


FIGURE 4.2: The workflow of generating the balanced **train–02** dataset and unbalanced **test–u01** dataset.

genes from the pool of 2,760 viable genes. These genes were also included in the training dataset. Figure 4.2 shows the overall workflow.

- **Balanced training dataset (train–03) and unbalanced test dataset (test–u02):** This training dataset also consists of equal number of lethal

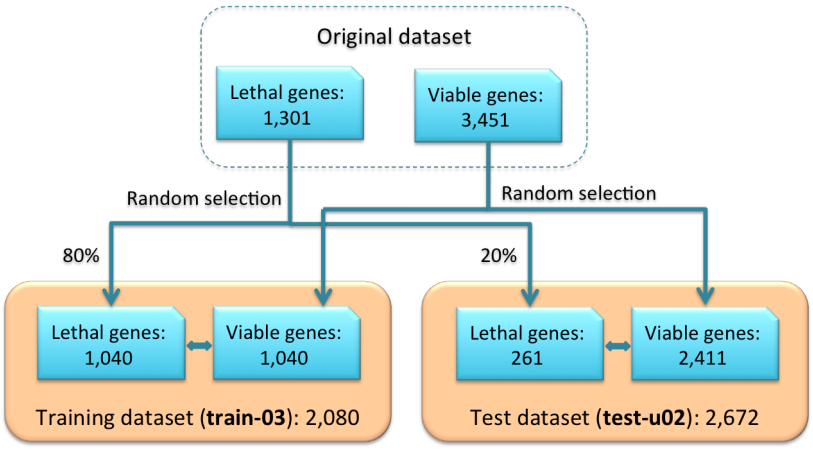


FIGURE 4.3: The workflow of generating the balanced **train–03** dataset and unbalanced **test–u02** dataset.

and viable mouse genes, whereas these numbers are unequal for the test dataset. In this case, 80% (1,040) lethal genes were randomly selected from the pool of 1,301 lethal genes. The remaining 20% lethal genes were included in the test dataset. From 3,451 viable genes, we randomly selected a total of 1,040 genes (equal to the size of lethal genes in the training dataset). The remaining 2,411 viable genes were included in the test dataset. Figure 4.3 shows the workflow of generating these datasets.

We further retrieved a total of 229 and 803 newly annotated lethal and viable mouse genes from the International Mouse Phenotyping Consortium (IMPC) (http://www.mousephenotype.org), who are generating and characterising new mouse knockouts on a large–scale. After we retrieved our mouse gene dataset from the MGI database, these targeted mouse genes were then published. In this newly annotated gene list, a lethal gene is defined as a gene knockout causing lethality before the weaning stage, whereas we defined lethal genes as those that produce lethality prior to postnatal day 3 in single gene knockout experiments.

Even though the classification of lethality in this newly annotated lethal list is slightly different from our definition, we sought to use this list along with the new viable list as a blind test dataset (**test–new**) to evaluate the performance of our classifier.

4.2.2 Selection of the Mammalian Gene Prediction Model

Our aim is to construct a machine learning classifier that could accurately classify mouse lethal (essential) and viable (non–essential) genes using their sequence and functional properties. The Naive Bayes, decision tree, Support Vector Machine (SVM) and Random Forest have been widely used in sequence-based prediction of essential genes for various organisms (Acencio and Lemke, 2009; Hwang et al., 2009; Yuan et al., 2012) (section 1.4). Among all of these machine–learning methods, Random Forest has already been found to outperform due to the following reasons:

- High level of prediction accuracy
- Robust. Therefore, it has the ability to accurately make predictions even the data has missing values.
- Can accommodate very large datasets and runs faster
- Faster in training
- Resistant to overfitting

We further sought to validate its superiority in predicting mouse genes in our datasets. Hence, we used Weka (Hall et al., 2009) to develop these four classifiers

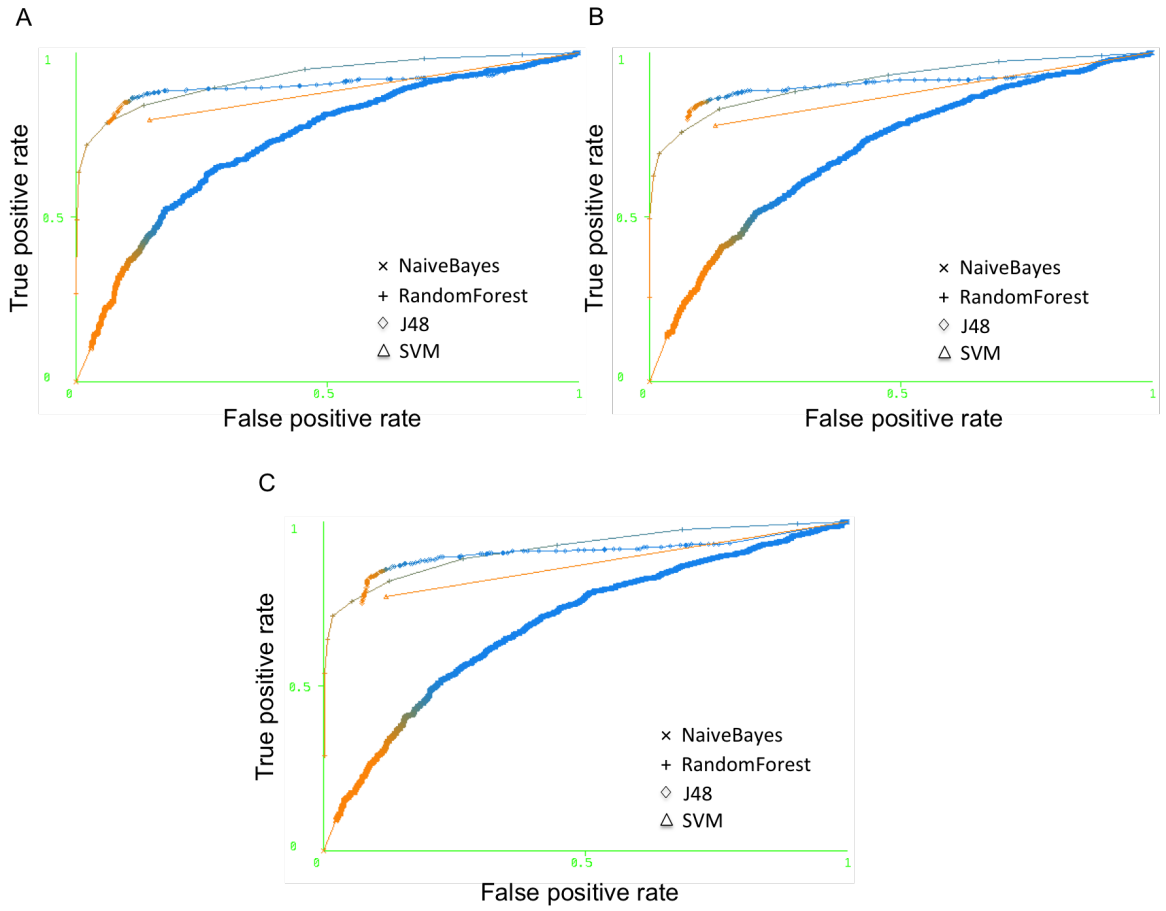


FIGURE 4.4: Lethal gene prediction performance of the Naive Bayes, J48 decision tree, SVM and Random Forest classifiers on balanced training datasets **train–01** (A), **train–02** (B) and **train–03** (C).

to predict gene essentiality in mammals (section 2.5.1). The Naive Bayes, J48 decision tree, SVM and Random Forest methods in Weka were developed with default parameters using three balanced training datasets and their prediction performance were evaluated based on a 10–fold cross validation method (section 2.5.2). Comparing the AUC values of the ROC curves (Huang and Ling, 2005), we again confirmed that the Random Forest classifier demonstrates the best performance (Figure 4.4 and 4.5; Table 4.1). Based on this observation, we selected Random Forest as our mammalian gene prediction classifier.

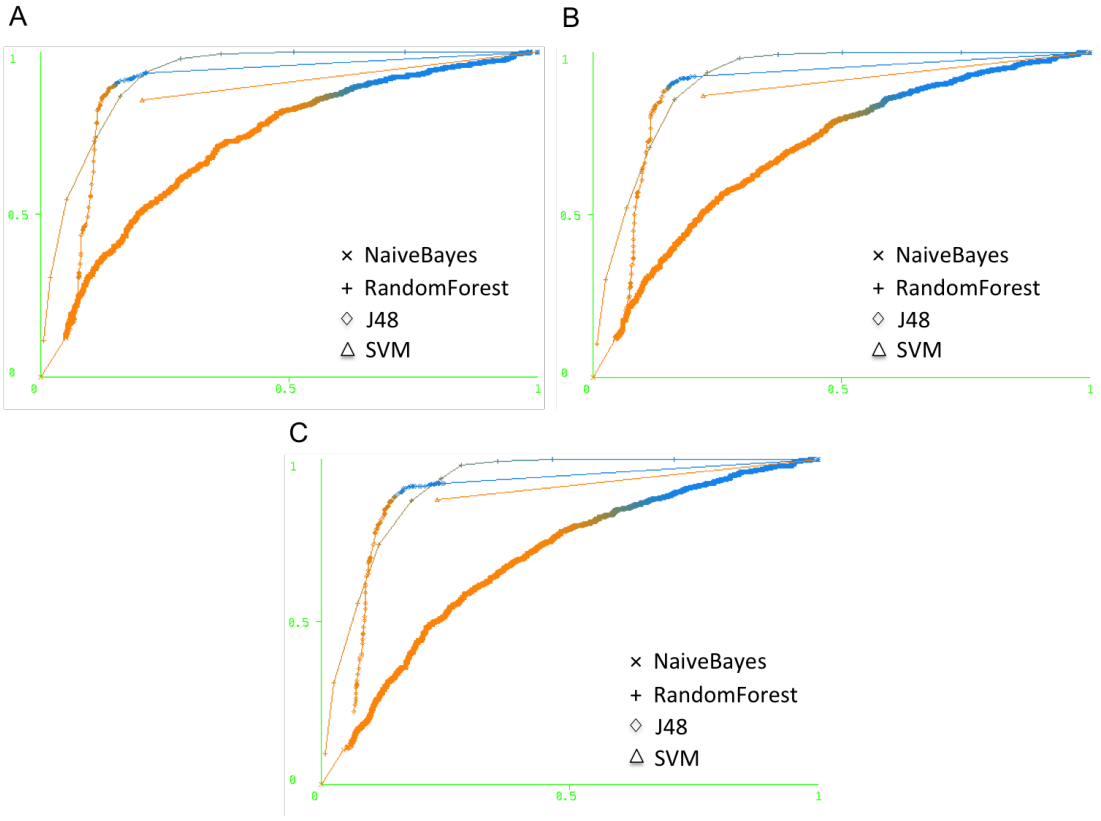


FIGURE 4.5: Viable gene prediction performance of the Nave Bayes, J48 decision tree, SVM and Random Forest classifiers on balanced training datasets **train–01** (A), **train–02** (B) and **train–03** (C)

TABLE 4.1: AUC values obtained from the 10–fold cross validation of the Naive Bayes, J48 decision tree, SVM and Random Forest classifiers trained and evaluated on **train–01**, **train–02** and **train–03** datasets.

	Training Datasets		
Classifiers	**train–01**	**train–02**	**train–03**
Naive Bayes	0.722	0.701	0.687
Random Forest	0.922	0.909	0.913
J48	0.877	0.875	0.877
SVM	0.824	0.822	0.825

4.2.3 Prediction of Mammalian Essential Genes using the Random Forest Classifier

The Random Forest is an ensemble classifier, which was first implemented in 2001 (Breiman, 2001) and since then it has been found as a highly accurate classification model in a number of studies (Bureau et al., 2005; Acencio and Lemke, 2009; Yuan et al., 2012). It operates by building multiple decision trees during training time. Each of these decision trees denotes prediction of a class. Two parameters regulate the growth of Random Forests: $numTrees$, the number of decision trees to be considered to grow the forest and $numFeatures$, the number of randomly selected features to evaluate at each tree node (section 2.5.1). The default value of $numTrees$ is 10 in Weka. However, the default value of $numFeatures$ is $log_2(numberoffeatures) + 1$, which is $log_2(102) + 1 = 7$ in this study. We trained the Random Forest classifier on the training datasets with different $numTrees$ (50, 100, 150, 200, 250, 300, 350, 400, 450, 500) and $numFeatures$ (7, 10, 15, 20, 25, 30, 35, 40, 45, 50) values and choose the optimal values of these parameters.

4.2.4 Classifier trained on train–01 dataset and evaluated on test–b test dataset

In this case, at first, we developed a Random Forest classifier (**RF–1**) by 10–fold cross validation on the balanced **train–01** dataset setting the missing attribute values to 1. Here, the best combination of parameter values giving the highest cross-validation accuracy was $numTree = 200$ and $numFeatures = 20$ (Table

TABLE 4.2: Accuracy of Random Forest classifiers trained on the **train–01** dataset with different combination of *numTrees* and *numFeatures* values.

	numTrees			
numFeatures	**50**	**100**	**150**	**200**
5	86.53	87.54	87.93	87.69
7	87.93	87.69	87.78	87.83
10	88.50	88.70	88.60	88.79
15	89.13	89.51	89.61	89.18
20	89.95	89.75	89.95	90.10

TABLE 4.3: 10-fold cross validation performance of the Random Forest classifier (**RF–1**) trained and evaluated on the **train–01** dataset setting missing attribute entries to -1.

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.838	0.963
FP Rate	0.037	0.162
Precision	0.958	0.856
F–Measure	0.894	0.907
AUC	0.961	0.961

4.2). The cross–validation accuracy of this classifier was 90.10% (1874/2080) with 872 true–positives (TPs), 168 false–negatives (FNs), 1002 true–negatives (TNs) and 38 false–positives (FPs) predictions. We also evaluated the predictive power of this classifier by different performance measures. Table 4.3 lists their values in detail by class. In addition, ROC curves were generated to confirm the classifier performance (Figure 4.6).

Predicting mouse genes in the balanced **test–b** dataset with a high accuracy of 90.99% (475/522) further validates the performance of the **RF–1** classifier. Table 4.4 displays the performance evaluation in detail. ROC curves of predicting lethal and viable mouse genes in the **test–b** dataset are shown in Figure 4.7.

Weka uses '?' symbol to represent the missing attribute values. Thus, we

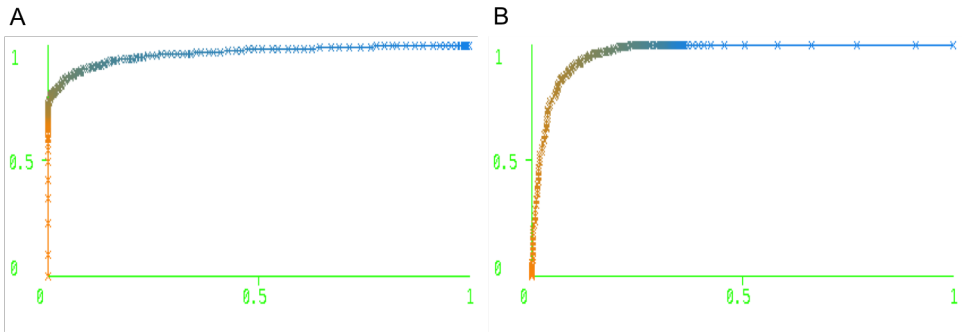


FIGURE 4.6: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–01** dataset (cross–validation) by the **RF–1** classifier

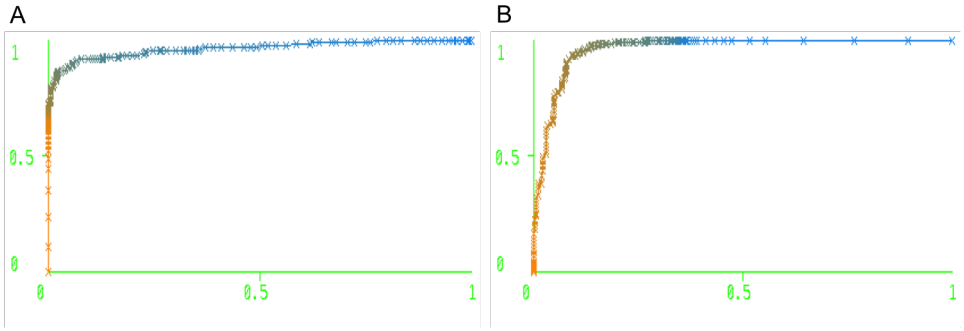


FIGURE 4.7: ROC curves for predicting lethal (A) and viable (B) genes in the **test–b** dataset by the **RF–1** classifier.

TABLE 4.4: Prediction of lethal and viable mouse genes in the **test–b** dataset using the **RF–1** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	219	42
	Viable	5	256

(B)

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.839	0.981
FP Rate	0.019	0.161
Precision	0.978	0.859
F–Measure	0.903	0.916
AUC	0.962	0.962

further developed a Random Forest classifier (**RF–1′**) on the balanced **train–01** dataset setting the missing attribute values to '?'. The cross–validation accuracy of this classifier was 87.74% (1825/2080) with 829 true–positives (TPs), 211

TABLE 4.5: 10-fold cross validation performance of the Random Forest classifier (**RF–1′**) trained and evaluated on the **train–01** dataset setting missing attribute entries to '?'.

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.797	0.958
FP Rate	0.042	0.203
Precision	0.950	0.825
F–Measure	0.867	0.887
AUC	0.951	0.951

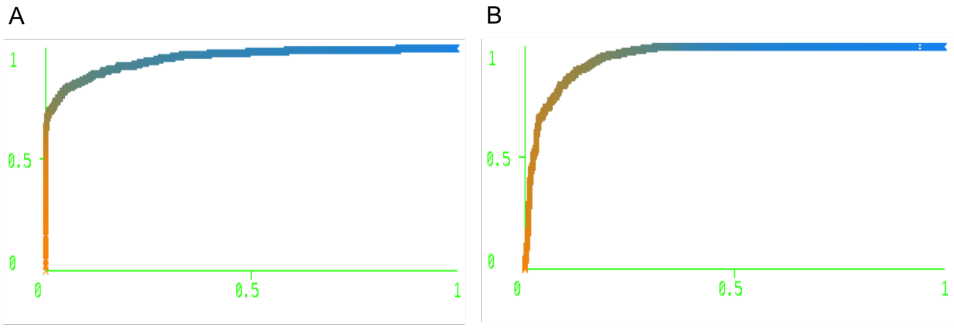


FIGURE 4.8: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–01** dataset (cross–validation) by the **RF–1′** classifier setting missing attribute entries to '?'

false–negatives (FNs), 996 true–negatives (TNs) and 44 false–positives (FPs) predictions. Table 4.5 displays the performance evaluation in detail. ROC curves of these cross-validation analysis are shown in Figure 4.8.

Predicting mouse genes in the balanced **test–b** dataset with a high accuracy of 86.59% (452/522) further validates the performance of the **RF–1′** classifier. Table 4.6 displays the performance evaluation in detail. ROC curves of predicting lethal and viable mouse genes in the **test–b** dataset are shown in Figure 4.9.

TABLE 4.6: Prediction of lethal and viable mouse genes in the **test–b** dataset using the **RF–1′** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	201	60
	Viable	10	251

(B)

Performance Measures	Class	
	Lethal	**Viable**
TP Rate (Recall)	0.770	0.962
FP Rate	0.038	0.230
Precision	0.953	0.807
F–Measure	0.852	0.878
AUC	0.942	0.942

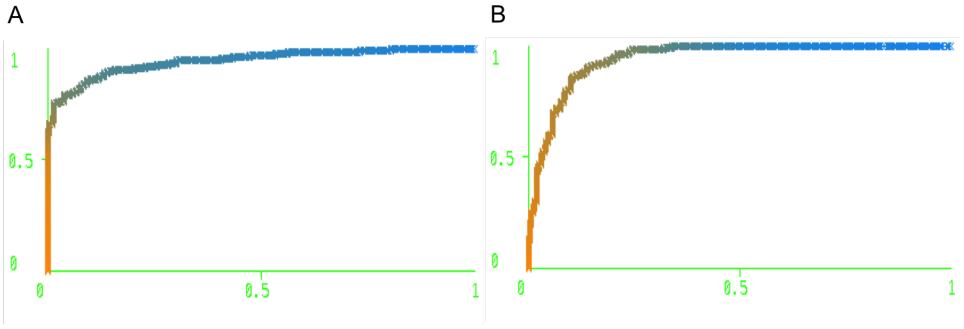


FIGURE 4.9: ROC curves for predicting lethal (A) and viable (B) genes in the **test–b** dataset by the **RF–1′** classifier.

4.2.5 Classifier trained on train–02 dataset and evaluated on test–u01 test dataset

We further developed a Random Forest classifier (**RF–2**) on the balanced training dataset **train–02** using 10–fold cross validation. In this case, the best combination of parameters values giving the highest cross–validation accuracy was again $numTree = 200$ and $numFeatures = 20$ (Table 4.7). The overall cross–validation accuracy of this classifier was 90.05% (1873/2080) with 852 TPs, 182 FNs, 1015 TNs and 25 FPs predictions. Table 4.8 demonstrates the high classification performance of **RF–2** classifier by means of different performance metrics. Moreover, ROC curves were generated to confirm its high prediction capability (Figure 4.10).

TABLE 4.7: Accuracy of Random Forest classifiers trained on the **train–02** dataset with different combination of *numTrees* and *numFeatures* values.

	numTrees			
numFeatures	**50**	**100**	**150**	**200**
5	86.77	86.92	86.92	87.01
7	87.40	87.78	87.88	87.45
10	89.18	88.36	88.41	88.65
15	89.23	89.32	89.27	89.56
20	89.71	89.9	90.04	90.05

TABLE 4.8: 10–fold cross validation performance of the Random Forest classifier (**RF–2**) trained and evaluated on the **train–02** dataset.

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.825	0.976
FP Rate	0.024	0.175
Precision	0.972	0.848
F–Measure	0.892	0.907
AUC	0.961	0.961

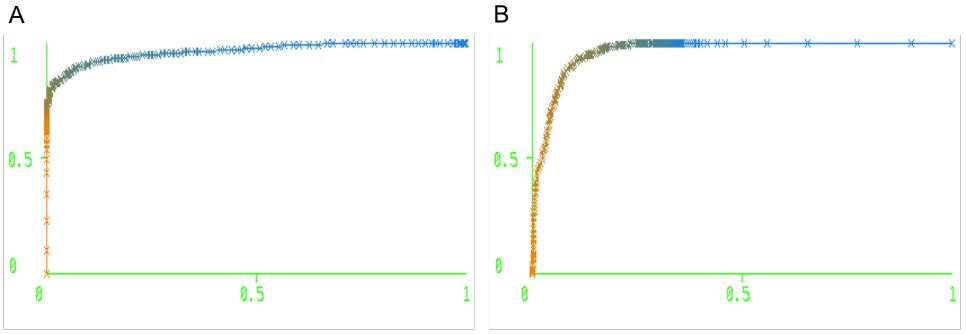


FIGURE 4.10: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–02** dataset (cross–validation) by the **RF–2** classifier.

Furthermore, we validated superiority of the **RF–2** classifier by classifying lethal and viable mouse genes in the **test–u01** dataset with an accuracy of 94.96% (904/952). Table 4.9 shows the confusion matrix along with performance metrics. ROC curves of predicting lethal and viable mouse genes in the **test–u01** dataset are shown in Figure 4.11.

Furthermore, we developed another Random Forest classifier (**RF–2′**) on

TABLE 4.9: Prediction of lethal and viable mouse genes in the **test–u01** dataset using the **RF–2** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	233	28
	Viable	20	671

(B)

Performance Measures	Class	
	Lethal	**Viable**
TP Rate (Recall)	0.893	0.971
FP Rate	0.029	0.107
Precision	0.921	0.960
F–Measure	0.907	0.965
AUC	0.968	0.968

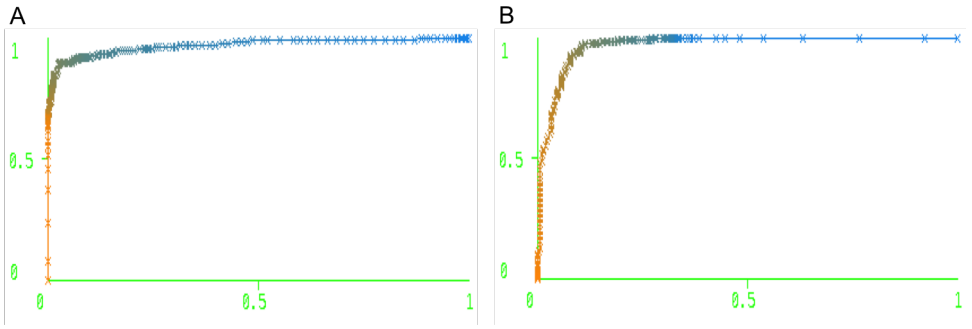


FIGURE 4.11: ROC curves for the lethal (A) and viable (B) genes prediction on the **test–u01** dataset (cross–validation) by the **RF–2** classifier.

TABLE 4.10: 10–fold cross validation performance of the Random Forest classifier (**RF–2′**) trained and evaluated on the **train–02** dataset.

Performance Measures	Class	
	Lethal	**Viable**
TP Rate (Recall)	0.780	0.958
FP Rate	0.042	0.220
Precision	0.949	0.813
F–Measure	0.856	0.879
AUC	0.945	0.945

the balanced **train–02** dataset setting the missing attribute values to '?'. The cross–validation accuracy of this classifier was 86.87% (1807/2080) with 811 true–positives (TPs), 229 false–negatives (FNs), 996 true–negatives (TNs) and 44 false–positives (FPs) predictions. Table 4.10 displays the performance evaluation in detail. ROC curves of these cross-validation analysis are shown in Figure 4.12.

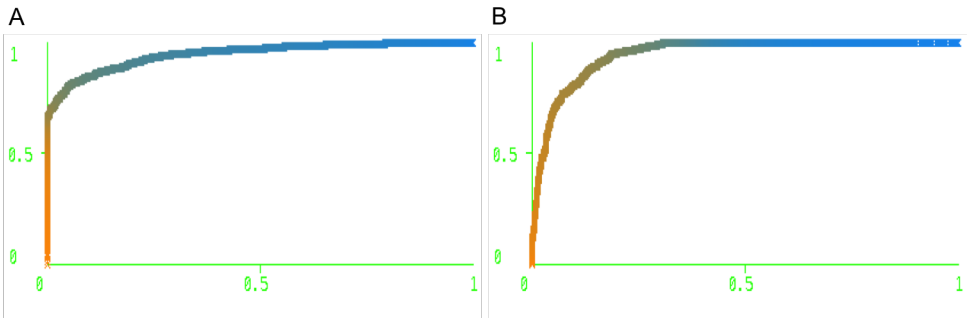


FIGURE 4.12: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–02** dataset (cross–validation) by the **RF–2′** classifier.

TABLE 4.11: Prediction of lethal and viable mouse genes in the **test–u01** dataset using the **RF–2′** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	216	45
	Viable	29	662

(B)

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.828	0.958
FP Rate	0.042	0.172
Precision	0.882	0.936
F–Measure	0.854	0.947
AUC	0.958	0.958

The superiority of the **RF–2′** classifier was further validated by classifying lethal and viable mouse genes in the **test–u01** dataset with an accuracy of 92.23% (878/952). Table 4.11 shows the confusion matrix along with performance metrics. ROC curves of predicting lethal and viable mouse genes in the **test–u01** dataset are shown in Figure 4.13.

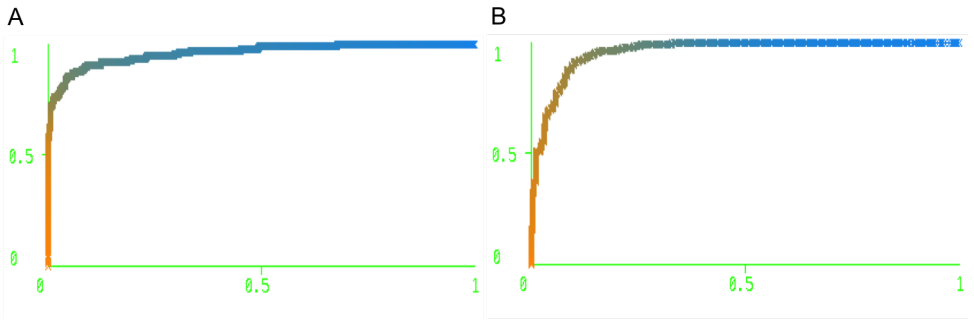


FIGURE 4.13: ROC curves for the lethal (A) and viable (B) genes prediction on the **test–u01** dataset (cross–validation) by the **RF–2′** classifier.

TABLE 4.12: 10–fold cross validation performance of the Random Forest classifier (**RF–3**) trained and evaluated on the **train–03** dataset.

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.877	0.974
FP Rate	0.026	0.123
Precision	0.971	0.888
F–Measure	0.922	0.929
AUC	0.967	0.967

4.2.6 Classifier trained on train–03 dataset and evaluated on test–u02 test dataset

In this case, 10–fold cross validation was used to develop a Random Forest classifier (**RF–3**) on the **train–03** dataset. Here, the best combination of parameters values giving the highest cross-validation accuracy was $numTree = 200$ and $numFeatures = 50$. This classifier gave the overall cross–validation accuracy of 92.56% (1925/2080) with 812 TPs, 128 FNs, 1013 TNs and 27 FPs predictions. Table 4.12 determines the high classification capability of **RF–2** classifier by means of different performance metrics. ROC curves further confirmed this result (Figure 4.14).

Additionally, predicting lethal and viable mouse genes in the **test–u02** dataset

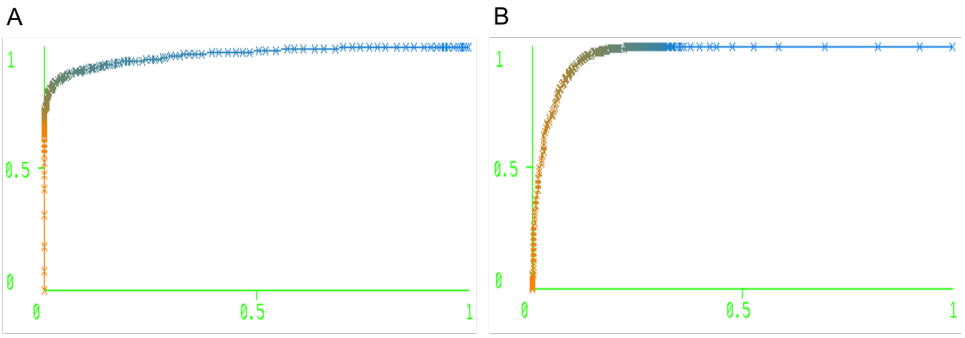


FIGURE 4.14: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–03** dataset (cross–validation) by the **RF–3** classifier.

TABLE 4.13: Prediction of lethal and viable mouse genes in the **test–u02** dataset using the **RF–3** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	216	45
	Viable	69	2342

(B)

Performance Measures	Class	
	Lethal	**Viable**
TP Rate (Recall)	0.828	0.971
FP Rate	0.029	0.172
Precision	0.758	0.981
F–Measure	0.791	0.976
AUC	0.962	0.962

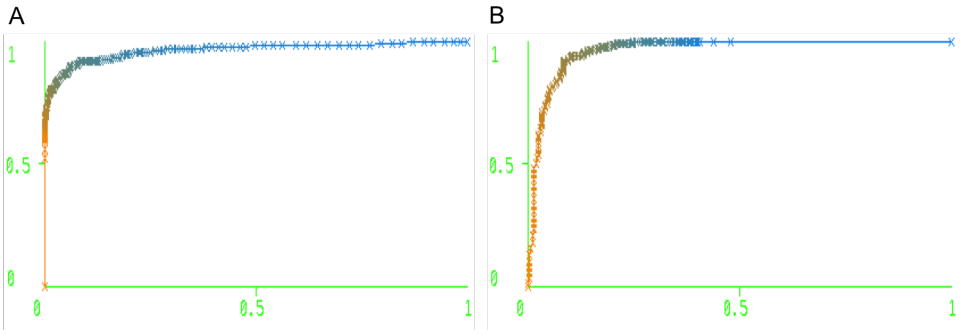


FIGURE 4.15: ROC curves for the lethal (A) and viable (B) genes prediction on the **test–u02** dataset (cross–validation) by the **RF–3** classifier.

with an accuracy of 95.73% (2558/2672) validated the superiority of the **RF–3** classifier. Table 4.13 shows the confusion matrix along with performance metrics. ROC curves of predicting lethal and viable mouse genes in the **test–u02** dataset are shown in Figure 4.15.

TABLE 4.14: 10–fold cross validation performance of the Random Forest classifier (**RF–3′**) trained and evaluated on the **train–03** dataset.

	Class	
Performance Measures	**Lethal**	**Viable**
TP Rate (Recall)	0.778	0.969
FP Rate	0.031	0.222
Precision	0.962	0.814
F–Measure	0.860	0.885
AUC	0.952	0.952

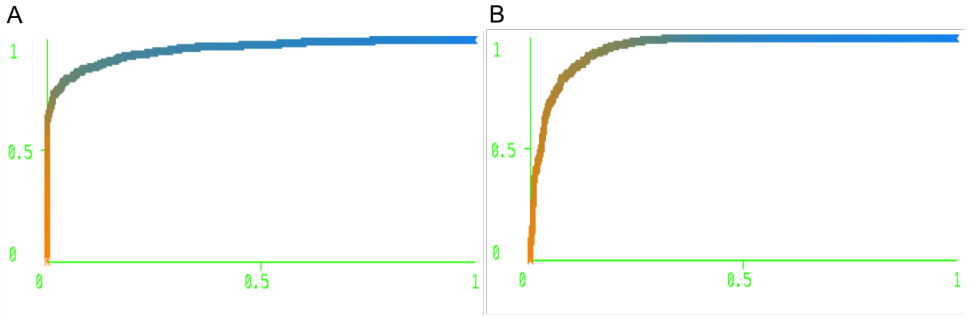


FIGURE 4.16: ROC curves for the lethal (A) and viable (B) genes prediction on the **train–03** dataset (cross–validation) by the **RF–3′** classifier.

We further developed a new Random Forest classifier (**RF–3′**) on the balanced **train–03** dataset setting the missing attribute values to '?'. The cross–validation accuracy of this classifier was 87.26% (1817/2080) with 809 true–positives (TPs), 231 false–negatives (FNs), 1008 true–negatives (TNs) and 32 false–positives (FPs) predictions. Table 4.14 displays the performance evaluation in detail. ROC curves of these cross-validation analysis are shown in Figure 4.16.

Predicting lethal and viable mouse genes in the **test–u02** dataset with an accuracy of 94.61% (2528/2672) also validated the superiority of the **RF–3′** classifier. Table 4.15 shows the confusion matrix along with performance metrics. ROC curves of predicting lethal and viable mouse genes in the **test–u02** dataset are shown in Figure 4.17.

TABLE 4.15: Prediction of lethal and viable mouse genes in the **test–u02** dataset using the **RF–3′** classifier. (A) The confusion matrix highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

Genes		Predicted	
		Lethal	Viable
Actual	Lethal	189	72
	Viable	72	2339

(B)

Performance Measures	Class	
	Lethal	**Viable**
TP Rate (Recall)	0.724	0.970
FP Rate	0.030	0.276
Precision	0.724	0.970
F–Measure	0.724	0.970
AUC	0.940	0.940

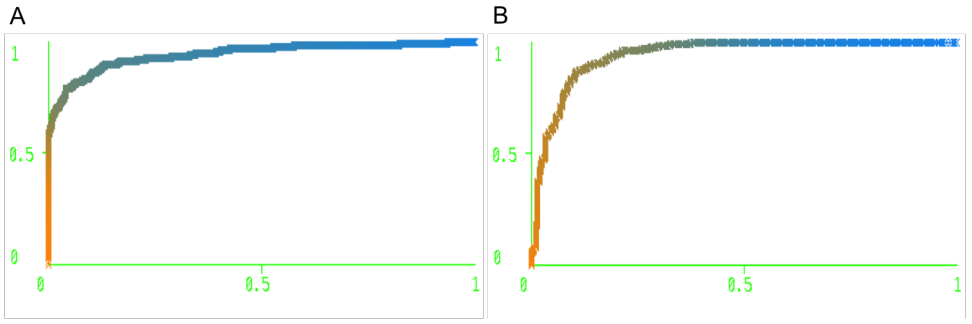


FIGURE 4.17: ROC curves for the lethal (A) and viable (B) genes prediction on the **test–u02** dataset (cross–validation) by the **RF–3′** classifier.

4.2.7 Predicting essentially of genes in the test–new dataset

The **test–new** test dataset consists of a number of targeted mouse genes that are recently annotated by the International Mouse Phenotyping Consortium (IMPC), in which 229 genes are lethal (targeted genes die before the weaning stage) and 803 genes are viable. We further evaluated the performance of **RF–1**, **RF–2**, **RF–3**, **RF–1′**, **RF–2′** and **RF–3′** classifiers with this test dataset. The **RF–1** classifier gave the overall accuracy of 65.21% (673/1032) with 32 TPs, 197 FNs, 641 TNs and 162 FPs. The **RF–2** classifier gave the accuracy of 61.43% (673/1032) with 22 TP, 207 FNs, 612 TNs and 191 FPs. The prediction accuracy of the **RF–3** classifier was 56.49% with 34 TPs, 195 FNs, 549 TNs and 254 FPs. The **RF–1′**

TABLE 4.16: Prediction of lethal and viable mouse genes in the **test–new** dataset using **RF–1**, **RF–2**, **RF–3**, **RF–1′**, **RF–2′** and **RF–3′** classifiers.

Classifiers	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
RF–1	Lethal	0.14	0.20	0.16	0.15	0.46
	Viable	0.79	0.86	0.76	0.78	0.46
RF–2	Lethal	0.09	0.23	0.10	0.10	0.44
	Viable	0.76	0.90	0.74	0.75	0.44
RF–3	Lethal	0.14	0.31	0.11	0.13	0.35
	Viable	0.68	0.85	0.60	0.58	0.35
RF–1′	Lethal	0.17	0.09	0.35	0.23	0.58
	Viable	0.91	0.82	0.79	0.85	0.58
RF–2′	Lethal	0.18	0.08	0.37	0.25	0.59
	Viable	0.91	0.81	0.79	0.85	0.59
RF–3′	Lethal	0.16	0.07	0.40	0.23	0.59
	Viable	0.93	0.83	0.79	0.85	0.59

classifier gave the overall accuracy of 74.51% (769/1032) with 41 TPs, 188 FNs, 728 TNs and 75 FPs. The **RF–2′** classifier gave the accuracy of 75.09% (673/1032) with 43 TP, 186 FNs, 732 TNs and 71 FPs. However, the prediction accuracy of the **RF–3′** classifier was 75.96% with 38 TPs, 191 FNs, 746 TNs and 57 FPs. Table 4.16 shows the values of different performance metrics of these analyses.

All six classifiers showed poor prediction performance on the **test–new** dataset, especially predicting lethal mouse genes. This is likely to be partly due to the definition of lethality for the newly annotated lethal genes in the **test–new** dataset differing from our definition of lethal genes. Also, we did not find information about many gene features of these genes from the existing data sources. Correcting these missing features might improve the overall prediction performance.

4.3 Correcting Features having Missing Values

While analysing gene features, we found that there was no information about the *Known* (**K**) protein–protein interaction (PPI) network (section 2.2.4) based features for 40% mouse genes. In addition, 8% mouse genes did not have information for 24 features including evolutionary age, the *Known–Predicted* (**KP**) PPI network based topological features (10 features) and developmental gene expression (13 features). These missing feature entries were set to -1 and '?' while constructing the training and test datasets. Though our Random Forest classifiers (**RF–1**, **RF–2**, **RF–3**, **RF–1′**, **RF–2′** and **RF–3′**) showed great accuracy for predicting mouse genes even with these missing values, we expected that correcting them could further improve the performance of our classifiers. Therefore, we replaced the missing values of the above-mentioned features in the **train–01**, **train–02** and **train–03** datasets with their mean value using the $ReplaceMissingValues$ filter in Weka.

Hence, 10–fold cross validation was used to develop three new Random Forest classifiers **RF–4**, **RF–5** and **RF–6** on the corrected **train–01**, **train–02** and **train–03** datasets, respectively. The parameter settings remained the same of **RF–1**, **RF–2** and **RF–3** classifiers.

The cross-validation accuracy of **RF–4** classifier was 89.81% (1868/2080) with 862 TPs, 178 FNs, 1006 TNs and 34 FPs. The cross–validation accuracy of **RF–5** classifier was 90.01% (1874/2080) with 857 TPs, 183 FNs, 1017 TNs and 23 FPs. However, **RF–6** classifier gave an overall cross–validation accuracy

TABLE 4.17: 10–fold cross validation performance of Random Forest classifiers **RF–4**, **RF–5** and **RF–6** trained and evaluated on the corrected **train–01**, **train–02** and **train–03** datasets, respectively.

Classifiers	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
RF–4	Lethal	0.82	0.03	0.96	0.89	0.96
	Viable	0.96	0.17	0.85	0.90	0.96
RF–5	Lethal	0.82	0.22	0.97	0.89	0.95
	Viable	0.97	0.17	0.84	0.90	0.95
RF–3	Lethal	0.85	0.01	0.98	0.91	0.96
	Viable	0.98	0.14	0.92	0.92	0.96

of 92.16% (1917/2080) with 893 TPs, 147 FNs, 1024 TNs and 16 FPs. Table 4.17 demonstrated the cross–validation performance of **RF–4**, **RF–5** and **RF–6** classifiers by means of different metrics.

Furthermore, we validated superiority of these classifiers by classifying lethal and viable mouse genes in the corrected **test–b**, **test–u01** and **test–u02** datasets with an accuracy of 90.80% (474/522), 93.80% (893/952) and 94.61% (2528/2672) for **RF–4**, **RF–5** and **RF–6** classifiers, respectively. Table 4.18 shows the high classification capability of these classifiers by means of confusion matrix and different performance metrics. ROC curves are shown in Figure 4.18. The adjustments of missing attributes, in general, improved the performance of our classifiers.

TABLE 4.18: Prediction of lethal and viable mouse genes in different test datasets using **RF–4**, **RF–5** and **RF–6** classifiers. (A) Confusion matrices highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

		Predicted					
		RF–4, test–b		**RF–5, test–u01**		**RF–6, test–u02**	
Genes		**Lethal**	**Viable**	**Lethal**	**Viable**	**Lethal**	**Viable**
Actual	Lethal	220	41	230	31	219	42
	Viable	7	254	28	663	102	2309

(B)

Classifiers	Test Datasets	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
RF–4	**test–b**	Lethal	0.843	0.027	0.969	0.902	0.966
		Viable	0.973	0.157	0.861	0.914	0.966
RF–5	**test–u01**	Lethal	0.881	0.041	0.891	0.886	0.971
		Viable	0.959	0.119	0.955	0.957	0.971
RF–6	**test–u02**	Lethal	0.839	0.042	0.682	0.753	0.959
		Viable	0.958	0.161	0.982	0.970	0.959

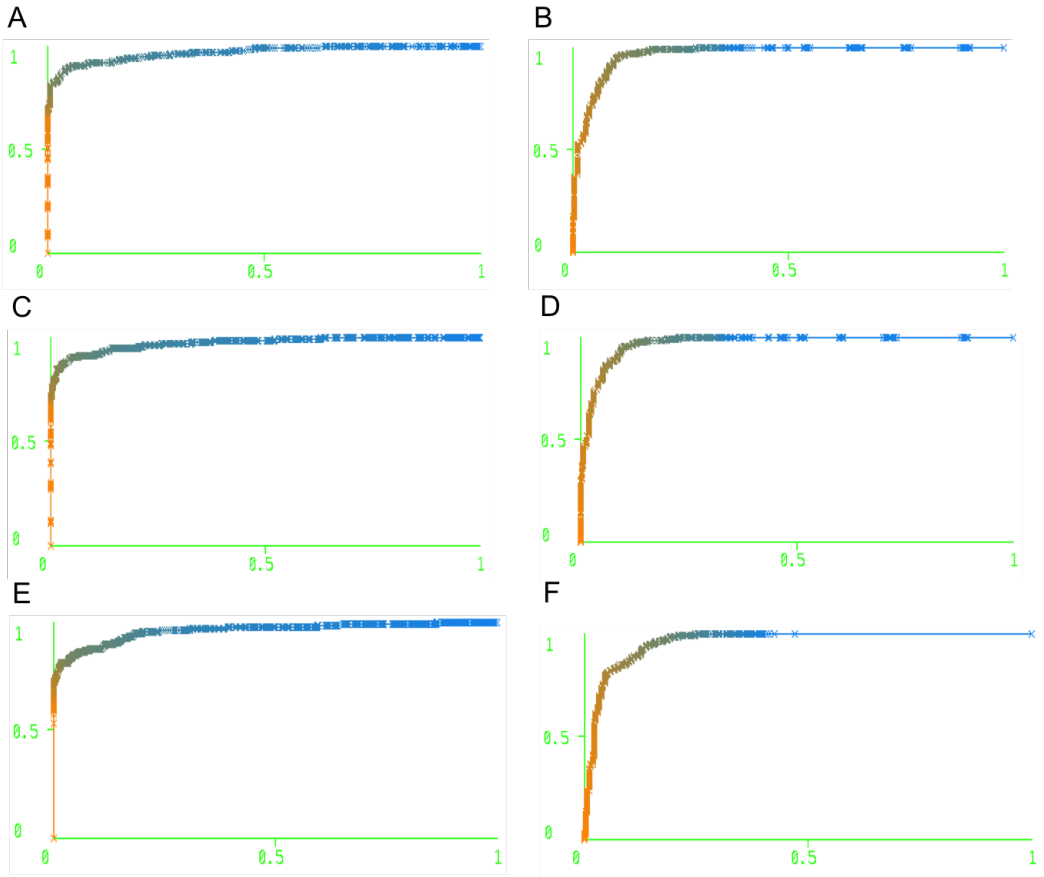


FIGURE 4.18: ROC curves for predicting lethal (A, C, E) and viable (B, D, F) genes in **test–b** (A, B), **test–u01** (C, D) and **test–u02** (E, F) datasets by the **RF–4**, **RF–5** and **RF–6** classifiers.

We further assessed the performance of **RF–4**, **RF–5** and **RF–6** classifiers in predicting mouse genes of the **test–new** dataset. The prediction accuracies were 73.35% (757/1032; TP = 44, FN = 185, TN = 713, FP = 90), 73.06% (754/1032; TP = 47, FN = 182, TN = 707, FP = 96), 72.86% (752/1032; TP = 51, FN = 178, TN = 701, FP = 102) for **RF–4**, **RF–5** and **RF–6** classifiers, respectively. These

new classifiers also showed poor performance on predicting lethal genes in the **test–new** dataset, however, the performance has improved compared to previous **test–new** analysis (section 4.3.4).

4.4 Integration of Discretisation

Discretisation maps a continuous feature to nominal values or intervals. Usage of nominal features has been shown to improve the performance of some classification methods (Liu et al., 2002). Studies have also demonstrated that discretisation facilitates faster learning and leads more accurate prediction (Dougherty et al., 1995). Following the correction of missing values, we, therefore, applied the Filtered classifier method in Weka to build Random Forest classifier with training and test datasets being passed through a supervised discretise attribute filter to discretise mouse gene features. We developed three new Random Forest classifier **RF–7**, **RF–8** and **RF–9** on the corrected and discretised **train–01**, **train–02** and **train–03** datasets, respectively. The previous parameter settings demonstrated the best performance.

The cross–validation accuracy of **RF–7** classifier was 88.61% (1843/2080) with 868 TPs, 172 FNs, 975 TNs and 65 FPs. The **RF–8** classifier showed the cross–validation accuracy of 89.57% (1863/2080) with 869 TPs, 171 FNs, 994 TNs and 46 FPs. However, **RF–9** classifier gave an overall cross–validation accuracy of 89.13% (1854/2080) with 866 TPs, 174 FNs, 988 TNs and 52 FPs. Table

TABLE 4.19: 10–fold cross validation performance of Random Forest classifiers **RF–7**, **RF–8** and **RF–9** trained and evaluated on the corrected and discretised **train–01**, **train–02** and **train–03** datasets, respectively.

Classifiers	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
RF–7	Lethal	0.835	0.063	0.930	0.880	0.949
	Viable	0.938	0.165	0.850	0.892	0.949
RF–8	Lethal	0.836	0.044	0.950	0.889	0.950
	Viable	0.956	0.164	0.853	0.902	0.950
RF–9	Lethal	0.833	0.050	0.850	0.885	0.947
	Viable	0.950	0.167	0.850	0.897	0.947

4.19 demonstrated the cross-validation performance of **RF–7**, **RF–8** and **RF–9** classifiers by means of different metrics.

RF–7, **RF–8** and **RF–9** classifiers further exhibited great performance by classifying lethal and viable mouse genes in the corrected and discretised **test–b**, **test–u01** and **test–u02** datasets with an accuracy of 89.46% (467/522, 91.07% (867/952) and 93.11% (2488/2672), respectively. Table 4.20 shows classification performance of these classifiers by means of different metrics. ROC curves are shown in Figure 4.19.

Furthermore, we evaluated the performance of **RF–7**, **RF–8** and **RF–9** classifiers in predicting mouse genes of the **test–new** dataset. The prediction accuracies were 66.96% (691/1032; TP = 72, FN = 157, TN = 619, FP = 184), 69.77% (720/1032; TP = 61, FN = 168, TN = 659, FP = 144) and 66.96% (691/1032; TP = 64, FN = 165, TN = 627, FP = 176) for **RF–7**, **RF–8** and **RF–9** classifiers, respectively. These results indicate that our Random Forest classifiers had difficulty in classifying mouse genes in the **test–new** dataset. The performance of our classifiers on the other datasets indicate that the poor prediction accuracy

on the **test–new** dataset is more likely due to the way lethal genes defined in the dataset.

Overall, these three classifiers built on discretised data exhibited lower prediction accuracy among all other Random Forest classifiers trained in our study.

4.5 Feature Selection using Information Gain

Accurate and reliable classification mainly relies upon the predictive strength of the quantifiable features used to train the classifier. Features may provide little or no information at all, or may be correlated to others, or they may be useful when integrated with other features. It is not always best to use all features to train a classifier. Usage of a subset of features can lower overfitting of the classifier, can improve classification accuracy and can speed up the overall training

TABLE 4.20: Prediction of lethal and viable mouse genes in different test datasets using **RF–7**, **RF–8** and **RF–9** classifiers. (A) Confusion matrices highlighting the number of TPs, FNs, TNs and FPs. (B) Different performance measures.

(A)

		Predicted					
		RF–7, test–b		**RF–8, test–u01**		**RF–9, test–u02**	
Genes		**Lethal**	**Viable**	**Lethal**	**Viable**	**Lethal**	**Viable**
Actual	Lethal	223	38	223	38	207	54
	Viable	17	244	47	644	130	2281

(B)

Classifiers	**Test Datasets**	**Gene Class**	**TP Rate (Recall)**	**FP Rate**	**Precision**	**F–Measure**	**AUC**
RF–7	**test–b**	Lethal	0.854	0.065	0.929	0.890	0.954
		Viable	0.935	0.146	0.865	0.899	0.954
RF–8	**test–u01**	Lethal	0.854	0.068	0.826	0.840	0.956
		Viable	0.932	0.146	0.944	0.938	0.956
RF–9	**test–u02**	Lethal	0.793	0.054	0.614	0.692	0.937
		Viable	0.946	0.207	0.977	0.961	0.937

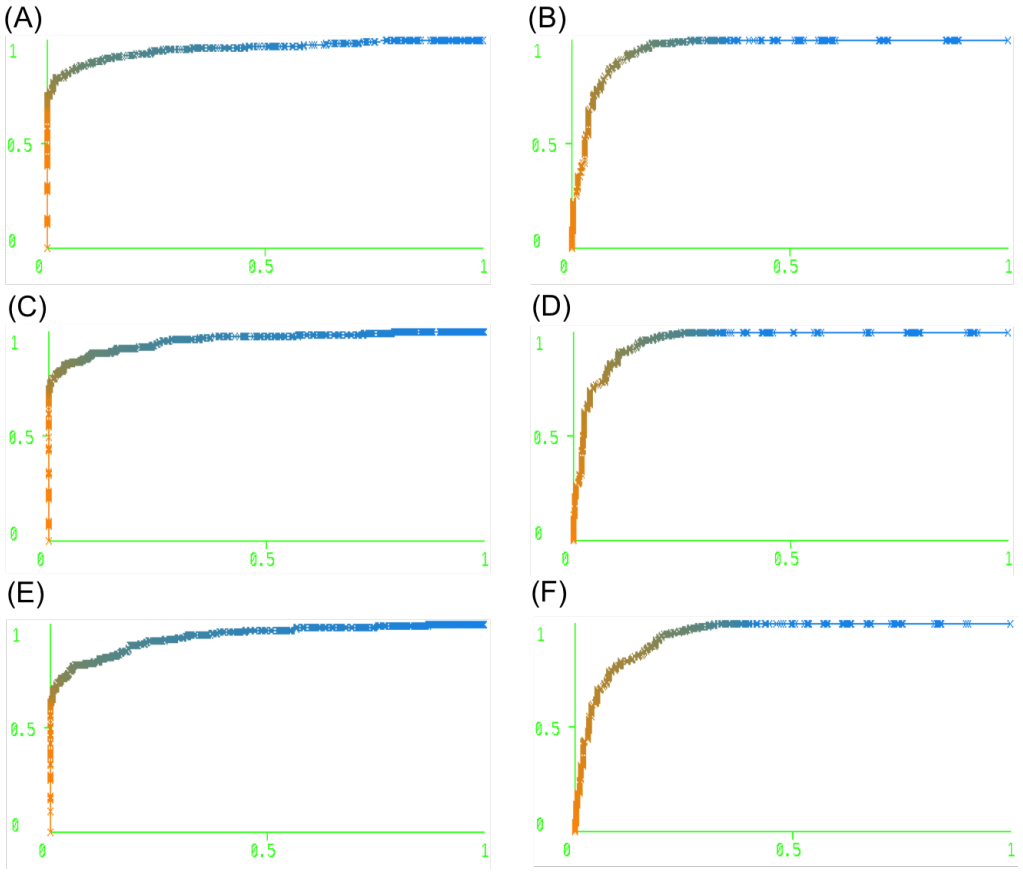


FIGURE 4.19: ROC curves for predicting lethal (A, C, E) and viable (B, D, F) genes in **test–b** (A, B), **test–u01** (C, D) and **test–u02** (E, F) datasets by the **RF–7**, **RF–8** and **RF–9** classifiers.

process. Hence, feature selection is imperative for the classification problem to handle datasets with a large number of features and to select informative features among all features. We, therefore, sought to use feature selection methods to improve classification accuracy by significantly reducing the number of features needed to accurately identify essential genes.

A number of feature selection methods are presently available to choose a smaller set of features from larger datasets. A feature could be strongly relevant, weakly relevant or irrelevant. Feature selection methods return a set of all relevant features that gives the best predictive accuracy. A set comprising of all relevant

features is more informative on the importance of features for classification. The Information Gain feature selection algorithm (Kira and Rendell, 1992) was found to be efficient for handling large datasets and selecting all relevant informative features (Lee and Lee, 2006). This filter method scores the features of training dataset using information gain and selects only top scored features satisfying a threshold. The information gain of a feature measures how important the feature is with respect to a classification target. A widely used measure of information gain is Shannon entropy (Shannon, 2001). Wrapper methods (Kohavi and John, 1997) are another form of feature selection algorithms for selecting all relevant features. They choose useful features in relation to the chosen classifier. They use the classifier to score feature subsets according to their predictive strength. The classifier is trained using each new subset and further tested on a hold–out set. Counting the number of false classifications made on the hold–out set (the error rate) provides the score for the respective feature subset. Genetic algorithms are one of the common wrappers which search for a best subset of features giving the highest classification accuracy. Though these methods are effective for selecting all informative features (Maldonado and Weber, 2009; Kursa and Rudnicki, 2011), they require greater computational efforts and time. Thus, we used the Information Gain feature selection method to select all relevant features for classifying mouse genes in our training datasets.

We applied the *InfoGainAttributeEval* filter in Weka as an Information Gain feature selection method to select a subset of most informative mouse gene features

among all features in the training datasets. The threshold value was set as 0.01, *i.e.*, only those features were selected whose information gain was greater than 0.01. The default threshold was −1.79 in Weka. We then developed Random Forest classifiers based on this set of selected features.

The Information Gain method selected 65 significant features from the pool of all features present in the corrected **train–01** dataset. Table 4.21 shows top 30 of these features, which are sorted in descending order with respect to the corresponding information gain value. The average shortest path (ASP) length within the *Known–Predicted* (**KP**) PPI network came on top of the list, making it the most important feature for the classifier. Developmental expression, localisations at nucleus, other PPI network features, gene age, exon length and transcript count were also present in the list of the selected features.

We trained a Random Forest classifier (**RF–10**) on this reduced **train–01** dataset using 10 fold cross–validation. The cross–validation accuracy was 90.72% (1887/2080) with 889 TPs, 151 FNs, 998 TNs and 42 FPs. Predicting mouse genes in the **test–b** dataset further confirmed the high prediction accuracy (92.53%, 483/522) of this classifier with 230 TPs, 31 FNs, 253 TNs and 8 FPs. Table 4.22 demonstrates performance of this classifier by means of different metrics. Our cross validation analyses showed that the classifier **RF–10** provided more accurate classification compared to **RF–1**, **RF–4** and **RF–7** classifiers (trained on all features).

From the corrected **train–02** dataset, the Information Gain method output

TABLE 4.21: Top 30 features selected from the **train–01** dataset using the information gain feature selection method

Information Gain	Gene Features
0.5637	ASP (KP PPI Network)
0.5562	Closeness Centrality (KP PPI Network)
0.1345	Organogenesis (Transcript/Million)
0.1046	Fetus (Transcript/Million)
0.0851	TC (KP PPI Network)
0.0834	EPC (KP PPI Network)
0.0811	Blastocyst (Transcript/Million)
0.0773	MNC (KP PPI Network)
0.0697	Degree (KP PPI Network)
0.0694	DMNC (KP PPI Network)
0.0678	Gastrula (Transcript/Million)
0.0601	Nucleus (UniProt)
0.0591	ASP (Known PPI Network)
0.0586	Closeness Centrality (Known PPI Network)
0.0547	Oocyte (Transcript/Million)
0.0542	Morula (Transcript/Million)
0.0519	Nuclues (WoLF_PSORT Score)
0.0511	Cleavage (Transcript/Million)
0.0504	BN (KP PPI Network)
0.0425	BC (Known PPI Network)
0.0398	Neonate (Transcript/Million)
0.0395	Clustering Coefficient (KP PPI Network)
0.0385	Acetylation
0.0378	Transcription
0.0378	TC (Known PPI Network)
0.0364	Age
0.0339	Glycoprotein
0.0330	BN (Known PPI Network)
0.0330	Zygote (Transcript/Million)
0.0326	Degree (Known PPI Network)

TABLE 4.22: Performance metrics of the **RF–10** classifier trained on selected features of the **train–01** dataset and evaluated on the **test–b** dataset.

Datasets	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
train–01	Lethal	0.855	0.040	0.955	0.902	0.964
	Viable	0.960	0.145	0.869	0.912	0.964
test–b	Lethal	0.881	0.031	0.966	0.922	0.971
	Viable	0.969	0.119	0.891	0.928	0.971

61 features those found most relevant among all. Table 4.23 lists top 30 of these features sorted in descending order with respect to information gain. We further developed a Random Forest classifier (**RF–11**) on this reduced dataset. This

TABLE 4.23: Top 30 features selected from the **train–u01** dataset using the information gain feature selection method

Information Gain	Gene Features
0.5479	ASP (KP PPI Network)
0.5478	Closeness Centrality (KP PPI Network)
0.1148	Organogenesis (Transcript/Million)
0.0939	Fetus (Transcript/Million)
0.0832	TC (KP PPI Network)
0.0813	EPC (KP PPI Network)
0.0779	Blastocyst (Transcript/Million)
0.0755	Degree (KP PPI Network)
0.0741	MNC (KP PPI Network)
0.0632	ASP (Known PPI Network)
0.0612	Gastrula (Transcript/Million)
0.0612	Closeness Centrality (Known PPI Network)
0.0536	Morula (Transcript/Million)
0.0503	BC (Known PPI Network)
0.0498	BN (KP PPI Network)
0.0474	Oocyte (Transcript/Million)
0.0461	Nuclues (WoLF_PSORT Score)
0.0447	Nucleus (UniProt)
0.0424	Age
0.0410	Cleavage (Transcript/Million)
0.0384	DMNC (KP PPI Network)
0.0380	Egg_Cylinder (Transcript/Million)
0.0372	Zygote (Transcript/Million)
0.0359	Clustering Coefficient (KP PPI Network)
0.0352	Acetylation
0.0341	Unfertilized_Ovum (Transcript/Million)
0.0334	Neonate (Transcript/Million)
0.0317	Transcription
0.0314	BN (Known PPI Network)
0.0299	TC (Known PPI Network)

TABLE 4.24: Performance metrics of the **RF–11** classifier trained on selected features of the **train–02** dataset and evaluated on the **test–u01** dataset.

Datasets	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
train–02	Lethal	0.856	0.029	0.967	0.908	0.963
	Viable	0.971	0.144	0.871	0.918	0.963
test–u01	Lethal	0.912	0.054	0.865	0.888	0.970
	Viable	0.946	0.088	0.966	0.956	0.970

classifier built on selected features had the cross–validation accuracy of 91.35%

TABLE 4.25: Top 30 features selected from the **train–u02** dataset using the information gain feature selection method

Information Gain	Gene Features
0.5653	Closeness Centrality (KP PPI Network)
0.5603	ASP (KP PPI Network)
0.1140	Organogenesis (Transcript/Million)
0.0970	Fetus (Transcript/Million)
0.0861	TC (KP PPI Network)
0.0756	MNC (KP PPI Network)
0.0746	Degree (KP PPI Network)
0.0731	Blastocyst (Transcript/Million)
0.0708	EPC (KP PPI Network)
0.0593	Gastrula (Transcript/Million)
0.0549	Morula (Transcript/Million)
0.0546	Closeness Centrality (Known PPI Network)
0.0489	Nucleus (UniProt)
0.0472	Oocyte (Transcript/Million)
0.0455	BN (KP PPI Network)
0.0444	Age
0.0433	Acetylation
0.0413	Cleavage (Transcript/Million)
0.0398	ASP (Known PPI Network)
0.0396	DMNC (KP PPI Network)
0.0380	BC (Known PPI Network)
0.0359	Neonate (Transcript/Million)
0.0347	BN (Known PPI Network)
0.0340	Egg Cylinder (Transcript/Million)
0.0305	Transcription
0.0295	Clustering Coefficient (Known PPI Network)
0.0293	Degree (Known PPI Network)
0.0292	MNC (Known PPI Network)
0.0283	Clustering Coefficient (KP PPI Network)
0.0279	Nuclues (WoLF PSORT Score)

(1900/2080) with 890 TPs, 150 FNs, 1010 TNs and 30 FPs. We further validated the performance of this classifier by predicting mouse genes in the **test–u01** dataset with 93.70% (892/952), where TPs, FNs, TNs and FPs were 238, 23, 654 and 37, respectively. Table 4.24 shows different performance measures of this analysis. Our cross validation analyses demonstrated that the classifier **RF–11** provided more accurate classification compared to **RF–2**, **RF–5** and **RF–8** classifiers.

TABLE 4.26: Performance metrics of the **RF–12** classifier trained on selected features of the **train–03** dataset and evaluated on the **test–u02** dataset.

Datasets	Gene Class	TP Rate (Recall)	FP Rate	Precision	F–Measure	AUC
train–03	Lethal	0.851	0.032	0.964	0.904	0.964
	Viable	0.968	0.149	0.867	0.915	0.964
test–u02	Lethal	0.828	0.039	0.699	0.758	0.958
	Viable	0.961	0.172	0.981	0.971	0.958

Furthermore, the Information Gain method selected 55 most informative features from the corrected **train–03** dataset out of 102 features (top 30 features in Table 4.25). The closeness centrality feature of the *Known–Predicted* (**KP**) PPI network was present on top of the list, making it the most significant feature for the classifier. A Random Forest classifier (**RF–12**) was then developed on this reduced dataset. The cross–validation accuracy of this classifier was 90.96% (1892/2080) with 885 TPs, 155 FNs, 1007 TNs and 33 FPs. Predicting mouse genes in the **test–u02** dataset further validated the great classification capability of this classifier. The overall accuracy on the **test–u02** dataset was 94.84% (2534/2672) in which TP = 216, FN = 45, TN = 2318 and FP = 93. Table 4.26 illustrates the classification efficiency of the **RF–12** classifier by means of different performance measures.

4.6 Discussion

Determining essential genes in mammals is a key concern of development biology as it facilitates understanding of cellular, developmental and vital tissue-specific processes and functions. Experimental methods for essential gene identification are

accurate, but usually require immense investment of time and resources. Computational methods circumvent these experimental constraints and offer accurate prediction of essential genes from their characteristics at a much reduced time and cost.

In Chapter 3, we found a number of gene features that significantly vary between lethal (essential) and viable (non–essential) genes in mouse. We, therefore, proposed a number of Random Forest classifiers to predict gene essentiality from these hallmark features. We sought to address to what extent these sequence and functional attributes can predict mouse lethal and viable genes. We constructed three training datasets to train Random Forest classifiers, which contained equal number of mouse lethal and viable genes, each comprised of 102 features. Balanced training datasets were used, as imbalanced datasets could lower the classification performance by making more false positive predictions. In addition, we made three separate test datasets to validate the performance of our classifiers. Mouse genes in the test datasets were not included in classifier training. Our Random Forest classifiers, which were built on all 102 gene features demonstrated high accuracy of ∼91% and AUC of ∼0.963, suggesting that lethal and viable genes are highly predictable from their characteristics. The prediction of mouse genes in test datasets with accuracy of ∼93% and AUC of ∼0.964 further confirmed the superiority of our classifiers.

Our classifiers showed poor performance on predicting lethal genes in the **test–new** dataset containing mouse genes annotated recently by the IMPC. The

poor performance on this test dataset is attributed to differences in the class labelling since IMPC defines the lethal gene as a gene knockout causing lethality before the weaning stage. In contrast, we defined lethal genes as those that produce lethality prior to postnatal day 3 in single gene knockout experiments. We further checked whether the feature distributions in the **test–new** dataset (for each class) differ from the training data. The difference of different features between lethal and viable genes in this test dataset are listed in **AppendexA**. We have found many of the important features including gene length, exon length, intron length, proportions of aliphatic, polar and non–polar residues, molecular weight, protein length, MNC, proportion of Transferases and Hydrolases, and plasma membrane score, which offered statistically significant differences for genes in the training dataset, were not anymore statistically different in the **test–new** dataset. This is expected as many of the lethal genes present in this test dataset would have been included in the viable dataset if they were labelled based on the criteria that we used for defining lethal genes. In addition, for gene length, intron length, and proportions of serine and alipahtic residues, we observed opposite trend, whereas for all other features, the trends were consistent with that of the training dataset. We believe this is due to the fact that membership of certain genes in the lethal dataset may have influenced the overall pattern that were observed. Additional information of the time when the lethality occurs for these mouse genes may help us to better understand the source of the discrepancy that we have observed from the **test–new** dataset. We also believe that this result does not necessarily

undermine the efficacy of our model rather it highlights the problem of addressing the issue of gene essentiality without a global standard.

Our proposed classifiers showed the highest classification performance compared to the AUC values of 0.803 (Yang et al., 2014), 0.782 (Yuan et al., 2012), 0.9 (Deng et al., 2010) and 0.773 (Acencio and Lemke, 2009) in previous studies. The other performance measures including recall, precision and F–measure also proved the high performance of our classifiers. Correcting missing values of gene features in the datasets further improved the classification accuracy of our classifiers (AUC = ∼0.965). Though we expected to further refine classifier performance through discretisation, it slightly lowered the accuracy of our Random Forest classifiers (AUC = ∼0.949), suggesting that our classifiers work best with numeric features rather than nominal features. Besides the advantages of using the Random Forest method, one probable reason for getting such a high performance could be the accuracy of lethal and viable gene assignments in our datasets.

Furthermore, in order to further improve classification performance and to speed up the training process, we used the Information Gain feature selection algorithm which selected a subset of features from training datasets (65, 61 and 55 features from **train–01**, **train–02** and **train–03** datasets, respectively) based on information gain. Further development of Random Forest classifiers with selected features demonstrated the highest prediction accuracy (AUC = ∼0.971). This result suggests that the reduced subset of features is more informative to accurately predict gene essentiality. The information gain score of all selected features showed

that the most informative feature among all is either average shortest path (ASP) length or the closeness centrality within the *Known–Predicted* (**KP**) PPI network. Gene expression level at different developmental stages (including oocyte, zygote, cleavage, morula, egg cylinder, organogenesis, fetus, neonate, adult) was found to be highly informative. The highly relevant features for subcellular localisation were nucleus, extracellular region and plasma membrane. Gene evolutionary age was also found being highly informative. Moreover, almost all PPI network features including degree, betweenness centrality (BC), clustering co–efficient, BottleNeck (BN), Edge Percolation Component (EPC) showed their efficiency in providing more information for gene essentiality. Other informative features were: post–translational modifications, signal peptide, gene length, protein length, molecular weight, proportion of a number of amino acid residues (including W, L, F, K, V, E, non–polar, polar, basic, aromatic, charged), exon length and transcript count. These results generally support what has established in Chapter 2, where these selected features were found to significantly differentiate lethal genes from viable genes. Overall, these results indicate that the integration of feature selection technique in the Random Forest classifier outperforms the predictability of mouse lethal and viable genes by all features.

4.7 Summary

In this study, we proposed a novel Random Forest classifier incorporating feature selection algorithm and missing value correction technique to predict mouse essential genes from their sequence and functional properties. To the best of our knowledge, this is the first study where topological, biological and sequence–based gene properties have been systematically used to develop a classifier that can predict essential genes in the mouse genome. The proposed Random Forest classifier successfully predicted mouse essential genes with high precision. It shows higher prediction capability than other classifiers established in prior studies to predict essential genes in bacteria, yeast and mouse. This study will ultimately lead us to predict essential genes in human due to the high similarity between mouse and human genomes.

Chapter 5

Gene Duplication, Mammalian Essentiality and the Hourglass Model

5.1 Introduction

Gene duplication and mammalian gene essentiality

Gene duplication is a key evolutionary event in multicellular eukaryotes (Lynch and Conery, 2003), which potentially generates new genes (paralogues), with new biological functions (Ohno, 1970; Long et al., 2003). There has been much interest in understanding the roles of duplicate genes and their correlations with phenotypic changes resulting from the gene deletion. As mentioned earlier (section 1.1), genes are considered as essential or lethal if they generate lethal phenotypes when mutated or deleted. However, non–essential genes may be useful but not critical and their deletion produce less deleterious phenotypes. Prior studies reported

that the functional loss of deleting a duplicate gene could be compensated by the existence of its close paralogue in the same genome with overlapping function and expression (Gu, 2003; Gu et al., 2003; Conant and Wagner, 2004; Guan et al., 2007; Dean et al., 2008; DeLuna et al., 2008). Moreover, genome–wide gene knockdown or knockout experiments in *C. elegans* (Kamath et al., 2003; Conant and Wagner, 2004) and *S. cerevisiae* (Gu et al., 2003) showed that duplicate genes are considerably less essential than singletons (single–copy genes). However, studies in mouse knockout phenotypes (∼4000 genes) reported that the proportion of essential genes between singletons and duplicates is similar, therefore, mouse duplicate genes are just as essential as singletons (Liang and Li, 2007; Liao and Zhang, 2007). This trend in mouse was subsequently contradicted in a study where the authors observed an important role of human duplicate genes in genetic robustness (Hsiao and Vitkup, 2008). Overall, these results do not provide any consensus about the relationship between essentiality and gene duplication in mammals.

Liao and Zhang (2007) investigated several factors, including divergence in protein sequence, divergence in expression and evolutionary conservation between a duplicate gene and its closest paralogue that might bias the proportion of essential gene being similar between mouse singletons and duplicates. The authors did not find any significant data bias, which led them to conclude that functional compensation between mouse duplicates is rare. On the contrary, Liang and Li (2007) observed high protein connectivity for mouse duplicate genes compared to mouse singletons. The authors claimed that functionally critical genes are more

likely to be duplicated, since high protein connectivity implies high functional significance, further highlighting mouse duplicate genes being more essential. Their analysis could not justify why functionally essential genes are more likely to be duplicated in mouse, while yeast genes had the opposite trend (Prachumwat and Li, 2006). Moreover, results of these two studies (Liang and Li, 2007; Liao and Zhang, 2007) could be susceptible to potential data biases because researchers prefer to report those genes that show discernible phenotypes in the knockout experiments. Therefore, knockout datasets are likely to under–represent gene knockouts with no phenotypic change even though the experiments have actually taken place.

One study (Makino et al., 2009) found that the mouse knockout dataset is biased towards genes involved in development and genes duplicated by whole genome duplication (WGD) events. The authors further demonstrated that developmental genes tend to be more essential than non–developmental genes, regardless of being singletons or duplicates. WGD duplicates were more likely to be essential than genes derived from single gene duplication (SGD) in this study. In addition, studies (Su and Gu, 2008; Su et al., 2014) found that genes derived from recent duplications were under–represented in the mouse knockout dataset, leading to overestimation of essentiality frequency in duplicates. The overall proportion of mouse essential genes became significantly lower in duplicates compared to that in singletons after correcting these biases (Su and Gu, 2008; Makino et al., 2009).

Furthermore, when evolutionary age was considered, Chen et al. (2012b) observed that older mouse genes were more prone to be essential irrespective of being

singletons or duplicates. The authors also reported that singletons are more likely to be essential than duplicates, presumably because recently duplicated genes are more likely to retain shared functions and expressions. Su et al. (2014) further confirmed that the contribution to functional compensation by mouse duplicate genes is duplication–age dependent. Explanations of all prior analyses could not address why some duplicate pairs are both essential, some are both non–essential and some are mixed. We sought to further investigate this issue to understand the correlation between gene duplication and essentiality in mammals.

The hourglass model during mammalian development

Embryogenesis is the most critical phase of mammalian development, which coordinates the progressive transformation of a single fertilized egg into a complex multicellular organism. It has been observed that embryos from the same phyla are morphologically more divergent at early and late embryogenesis stages but morphologically conserved during mid–embryogenesis. This morphological pattern during embryogenesis is called the hourglass model of development (Duboule, 1994; Raff, 1996). The stage at mid–gestation, where embryos are similar in morphology to other embryos of the same phyla, is known as the *phylotypic stage* (Sander et al., 1983; Elinson, 1987) or *phylotypic period* (Richardson, 1995). Recently much attention has been paid to determine the developmental hourglass model (Abzhanov, 2013; Piasecka et al., 2013; Irie and Kuratani, 2014; Drost et al., 2015). Molecular interpretations of this model suggest that gene expression

patterns between organisms are most similar at the phylotypic stage. Previous studies (Domazet-Lošo and Tautz, 2010; Piasecka et al., 2013; Drost et al., 2015) reported that genes expressed at the phylotypic stage tend to have an older evolutionary origin than genes expressed in early or late development, thus forming an hourglass pattern when transcription age is plotted against developmental time. Support for the hourglass model of developmental gene expression has been shown for multiple organisms, including fungi (Cheng et al., 2015), plants (Quint et al., 2012; Drost et al., 2015), Drosophila (Domazet-Lošo and Tautz, 2010; Kalinka et al., 2010; Ninova et al., 2014), and zebrafish (Domazet-Lošo and Tautz, 2010). Only one study (Irie and Kuratani, 2011) has found the hourglass pattern during mammalian development by investigating the transcriptome. This inspired us to further examine the existence of the developmental hourglass model in mammals.

In this chapter, we investigated the relationships between gene essentiality and gene duplication in mouse. Our analysis found that evolutionary age as well as mode of gene duplication is strongly linked to mouse gene essentiality. It is likely that duplicates with similar developmental expression patterns are more likely to functionally compensate for each other. Hence, we explored the expression profile similarities for duplicates across 13 stages of mouse development. Our hypothesis was that duplicates with similar developmental co–expression are more likely to be viable, whereas duplicates without similar developmental co–expression are more likely to be lethal. Apart from that, the morphological hourglass model in mouse development is also addressed in this chapter. To determine whether the

hourglass pattern exists in mammalian embryos, we assessed the evolutionary age of genes expressed at the early, phylotypic and late stages in mouse development. Prior studies of conserved patterns of gene expression in mouse embryos (Irie and Kuratani, 2011; Bogdanović et al., 2016) have identified the phylotypic period as corresponding to the developmental stages of gastrula and organogenesis, and we adopted this definition for our study. We found evidence that the morphological hourglass pattern does exist in mouse development.

5.2 Results

5.2.1 Datasets

As mentioned in section 3.2, we obtained 1,301 lethal and 3,451 viable mouse genes from the MGI database (Bult et al., 2008) examining the phenotype information of knockout mice. Our dataset shows the same trend of (White et al., 2013) with viable genes being more common than lethal genes in mouse. We further retrieved a total of 22,944 protein–coding mouse genes from the MouseMine system (Motenko et al., 2015). After excluding 4,752 genes of known essentiality status and also the non–mouse genes, we found 16,960 mouse genes with unknown essentiality. Furthermore, we retrieved gene expression data as transcripts per million (TPM) from the UniGene database (Stanton et al., 2003) for 1,301 lethal, 3,409 viable and 14,599 unknown genes over 13 stages of mouse development.

TABLE 5.1: Proportion of mouse genes in different datasets. Here, All(%) refers to the proportion of singletons and duplicates. Lethal(%) and Viable(%) represent proportions of each gene type in singletons and duplicates

Genes	**All**	**All(%)**	**Lethal**	**Lethal(%)**	**Viable**	**Viable(%)**
Singleton	852	17.94	282	33.10	570	66.90
Duplicate	3900	82.12	1019	26.13	2881	73.87
Total	4749		1301		3451	

From the Blastp search (Altschul et al., 1990) and Ensembl gene tree analyses (see section 2.1.3), we obtained 1,019 lethal duplicates and 2,881 viable duplicate genes (Table 5.1). In agreement with (Chen et al., 2012b), we demonstrated that lethal genes originating from duplicates (26.13%) are considerably lower in proportion than those originating from singletons (33.10%), with a p–value = 4.82×10^{-4}. In addition, we obtained 143 lethal–lethal, 962 viable–viable and 491 lethal–viable mouse duplicate pairs within our dataset whose human orthologue duplicate pairs are listed in a previous study (Makino and McLysaght, 2010). Moreover, we found 2,223 small–scale duplicates (lethal: 500; viable: 1,723) and 1,834 whole–genome duplicates (lethal: 489; viable: 1,345).

5.2.2 Lethal Genes and Singleton genes have Older Evolutionary Age

A recent study of Chen et al. (2012b) reported that evolutionary age of genes is greatly linked to gene essentiality. The authors showed that knockout genes with earlier phyletic origin tend to be more essential in yeast and mouse, regardless of being singletons or duplicates. To confirm this result, we analysed the evolutionary age of mouse genes that were expressed over development. We considered two ages

TABLE 5.2: Number of lethal and viable genes in different categories expressed at 13 stages of mouse development

Developmental stages	All		Singletons		Duplicates	
	Lethal	Viable	Lethal	Viable	Lethal	Viable
Oocyte	534	693	162	121	372	572
Unfertilized ovum	336	347	88	57	248	290
Zygote	450	591	125	108	325	483
Cleavage	576	774	163	145	413	629
Morula	553	653	164	114	389	539
Blastocyst	760	1009	221	173	539	836
Egg cylinder	281	284	76	52	205	232
Gastrula	725	974	205	172	520	802
Organogenesis	1070	1805	263	303	807	1502
Fetus	1243	2873	279	461	964	2412
Neonate	1086	2361	255	381	831	1980
Juvenile	1180	2920	275	493	905	2427
Adult	1225	3127	280	513	945	2614

for mouse duplicates: the age of the duplicate common ancestor (DCA) and the age of the most recent duplication (MRD) (section 2.2.1.3). However, the age of non–duplicated genes (singletons) was estimated based on the age of their single common ancestor (SCA). Table 5.2 shows the total numbers of lethal and viable genes that are expressed at different developmental stages. As mentioned in the section 2.2.1.3, mouse gene ages (Table 3.4A) were estimated as millions of years (MYA) from the Ensembl (release 75) gene trees of mouse gene families (Vilella et al., 2009). We found DCA ages for 1,297 (99.69%) lethal and 3,435 (99.54%) viable genes, and MRD ages for a total of 1,276 (98.07%) lethal and 3,357 (97.28%) viable genes.

We compared the differences in different age groups between lethal and viable datasets. We found that majority of lethal singleton genes have evolutionary ages between 1215 and 937 MYA, and that they are older than lethal duplicates, viable singletons and viable duplicates (Figure 5.1). This result was observed from the

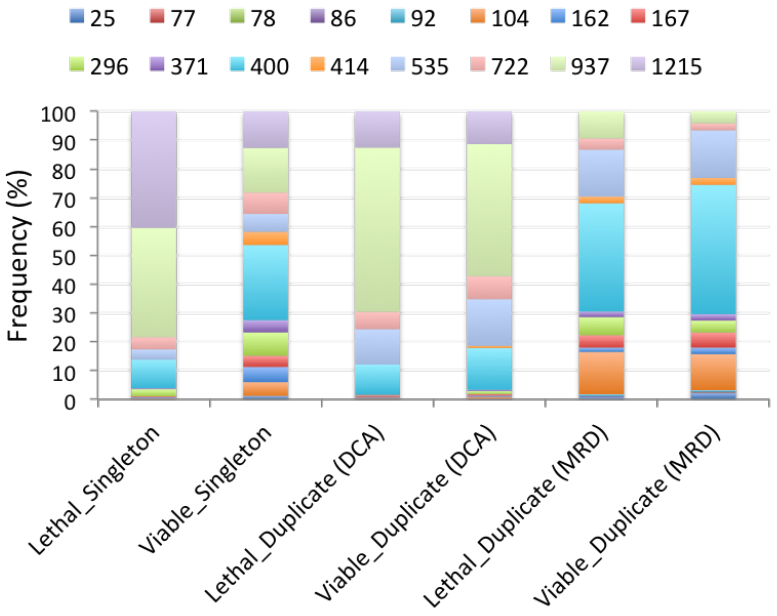


FIGURE 5.1: Percentages of lethal and viable genes for different age groups. Here, ages of mouse duplicates were calculated based on the Duplicate Common Ancestor (DCA) or Most Recent Duplication (MRD). For singleton genes, the age of their single common ancestor (SCA) was used.

analyses of both DCA and MRD age. We further examined proportions of mouse genes in different age groups that are expressed across 13 developmental stages and found that DCA gives much older ages than MRD (Figure 5.2). This is understandable because MRD only considers the most recent duplication event, while DCA accounts for the age of the most distant common ancestor. Moreover, analysing the evolutionary age of all lethal and viable genes expressed at each developmental time point revealed that lethal genes are more likely to be older than viable genes, irrespective of whether the duplicate gene age was calculated based on the DCA (Figure 5.3) or MRD (Figure 5.4). A greater percentage of lethal genes have evolutionary age between 1215 and 937 MYA. However, a greater percentage of viable genes were found to be between 400 and 535 MYA old. Table

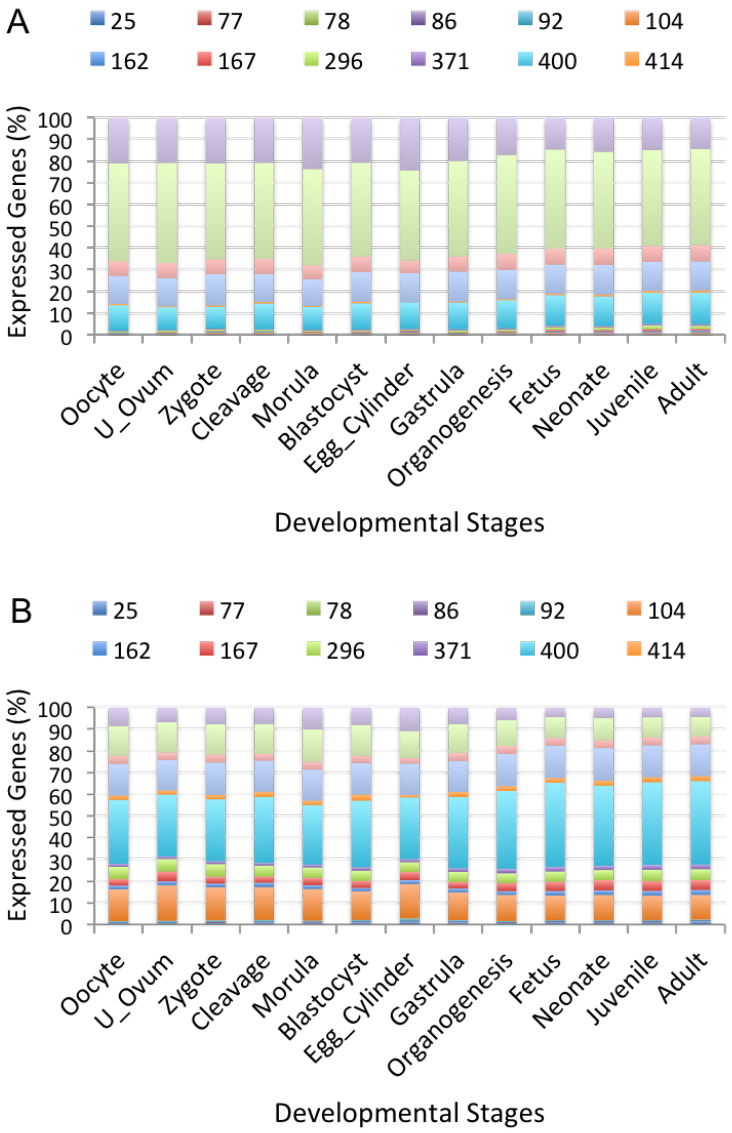


FIGURE 5.2: Percentages of expressed genes for SCA+DCA (A) and SCA+MRD (B) mouse gene ages over 13 stages of mouse development. Here, ages of mouse duplicates were calculated based on the Duplicate Common Ancestor (DCA) or Most Recent Duplication (MRD). For singleton genes, the age of their single common ancestor (SCA) was used.

5.3 shows the statistical significance of these observations.

Furthermore, analysis with singletons and duplicates over developmental stages confirmed that expressed mouse singletons have an older evolutionary age than

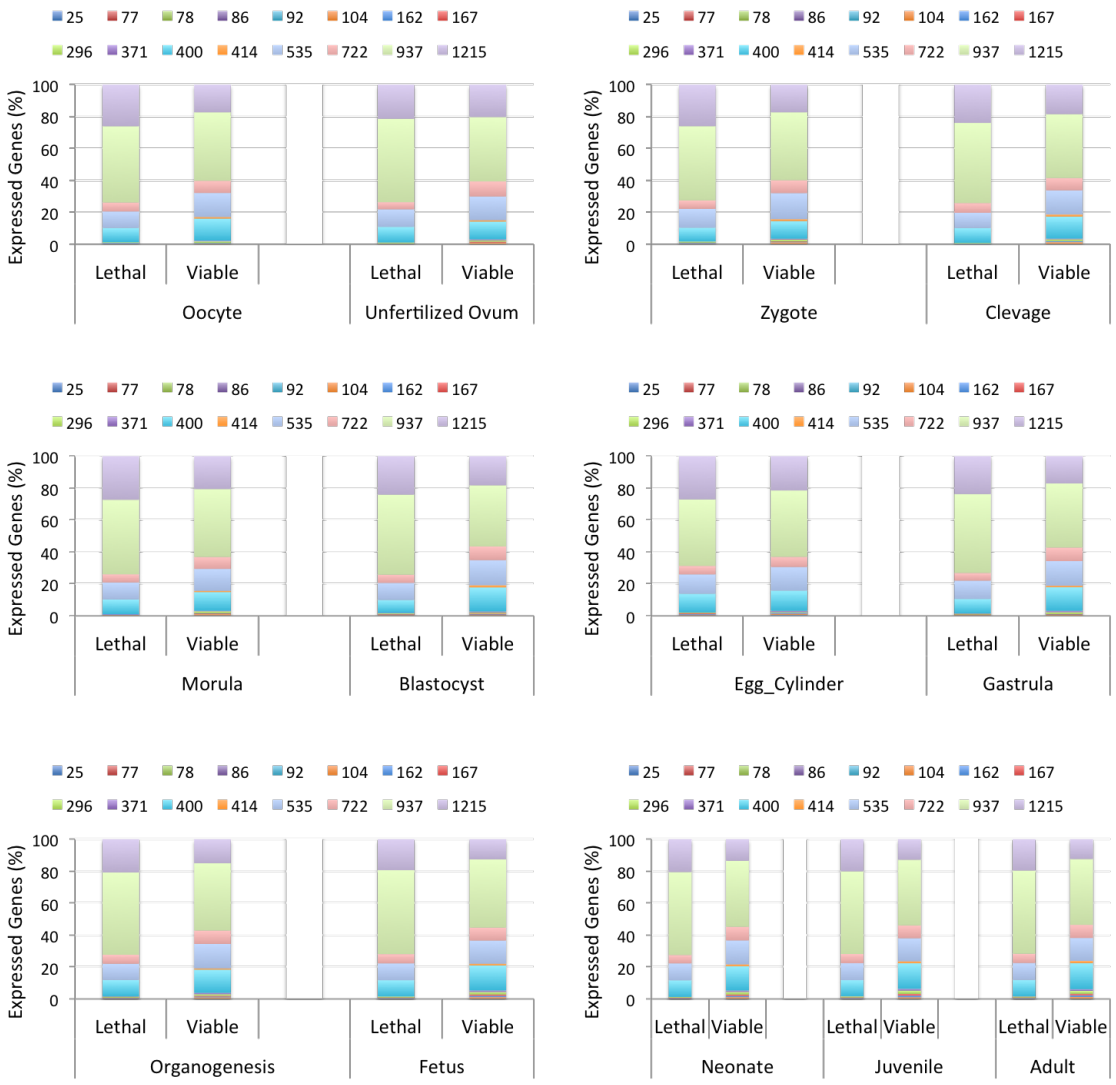


FIGURE 5.3: Percentages of lethal and viable genes for SCA+DCA mouse gene ages that are expressed over 13 developmental stages. Here, age of mouse duplicates and singletons was calculated based on the Duplicate Common Ancestor (DCA) and single common ancestor (SCA), respectively.

mouse duplicates expressed at the same stage of development (Figure 5.5). Moreover, lethal singletons are more likely to be older than viable singletons, lethal duplicates and viable duplicates expressed at the same stage of development.

Overall, our investigation demonstrated that lethal and singleton genes for

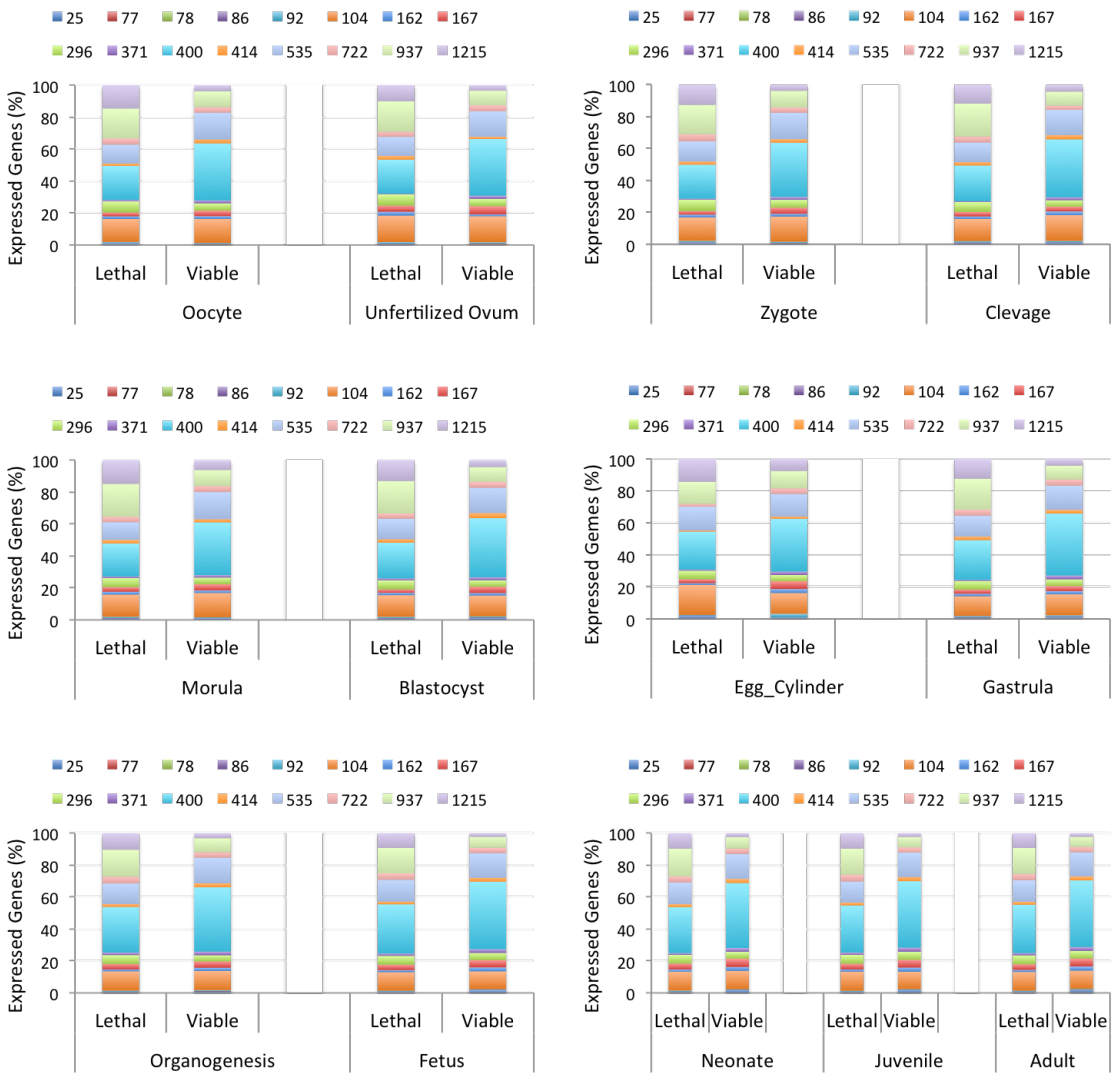
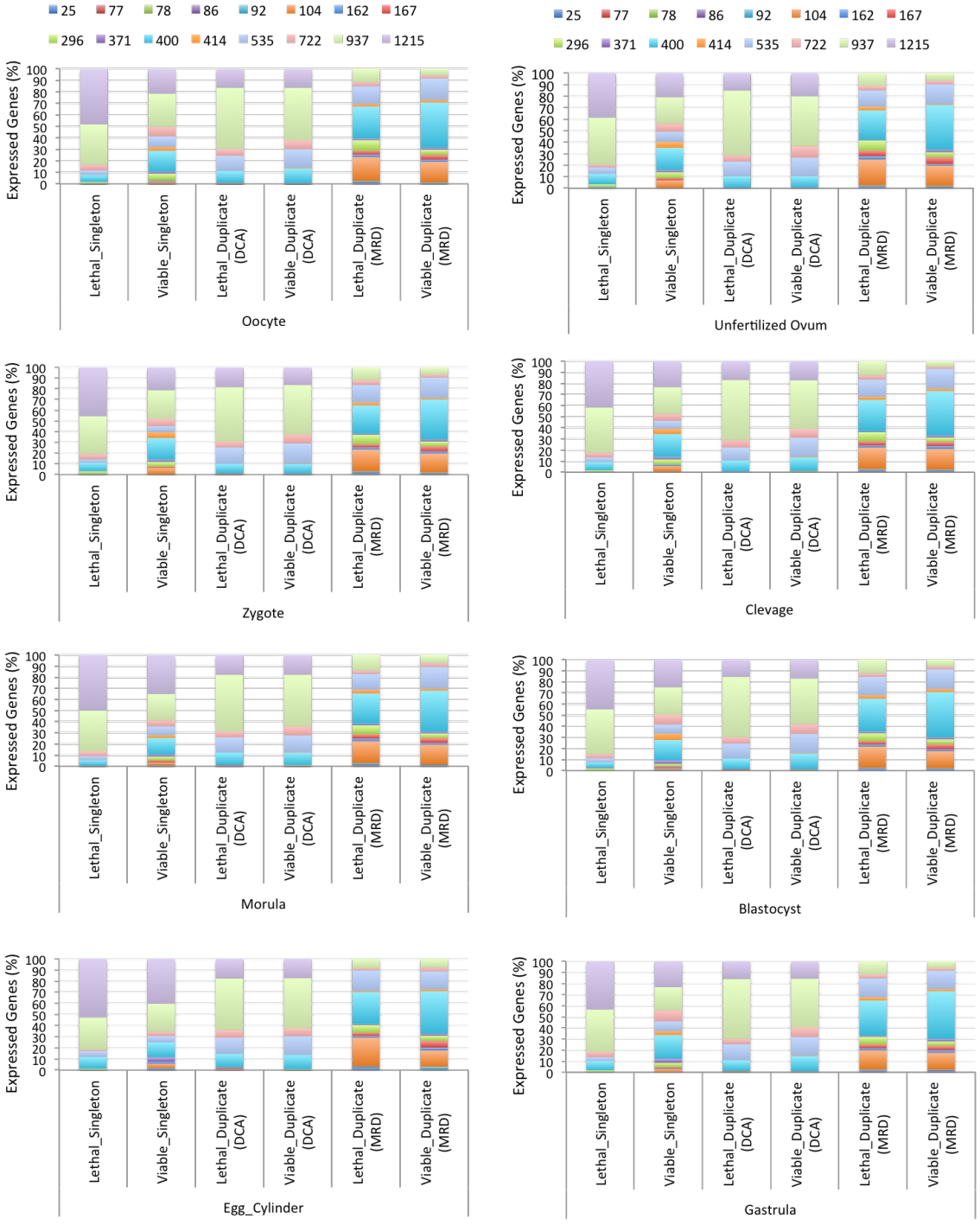


FIGURE 5.4: Percentages of lethal and viable genes for SCA+MRD mouse gene ages that are expressed over 13 developmental stages. Here, age of mouse duplicates and singletons was calculated based on the Most Recent Duplication (MRD) and single common ancestor (SCA), respectively.

mouse development are more ancient than viable and duplicate genes, which re-confirms what Chen et al. (2012b) established.

TABLE 5.3: Chi–squared test results highlighting differences between lethal and viable mouse gene proportions for different age groups at 13 developmental stages. Here, highlighted cells in yellow shows the statistically significant differences.

Developmental stages	Age	400	535	722	937	1215
Oocyte	DCA	0.014	0.023	0.175	0.205	9.9×10^{-04}
	MRD	3.9×10^{-06}	0.025	0.670	4.1×10^{-05}	9.5×10^{-11}
Unfertilized ovum	DCA	0.593	0.162	0.014	0.020	0.743
	MRD	4.8×10^{-04}	0.167	0.742	6.2×10^{-04}	7.9×10^{-04}
Zygote	DCA	0.221	0.056	0.095	0.345	2.4×10^{-03}
	MRD	1.2×10^{-04}	0.105	0.387	4.8×10^{-04}	4.2×10^{-07}
Cleavage	DCA	0.019	0.004	0.266	5.4×10^{-03}	0.028
	MRD	3.8×10^{-06}	0.099	0.203	4.2×10^{-09}	9.6×10^{-07}
Morula	DCA	0.181	0.132	0.099	0.284	0.017
	MRD	3.1×10^{-05}	0.004	0.722	2.5×10^{-06}	2.2×10^{-06}
Blastocyst	DCA	3.3×10^{-05}	4.4×10^{-03}	7.0×10^{-03}	1.7×10^{-04}	7.6×10^{-03}
	MRD	3.4×10^{-08}	0.082	0.755	1.2×10^{-09}	1.2×10^{-10}
Egg cylinder	DCA	0.751	0.384	0.623	0.987	0.179
	MRD	0.028	0.788	0.204	0.317	0.013
Gastrula	DCA	8.8×10^{-04}	0.031	4.5×10^{-03}	4.4×10^{-03}	2.4×10^{-03}
	MRD	5.3×10^{-07}	0.307	0.938	2.1×10^{-09}	1.5×10^{-09}
Organogenesis	DCA	2.0×10^{-03}	3.1×10^{-04}	7.9×10^{-03}	2.8×10^{-04}	3.2×10^{-04}
	MRD	3.1×10^{-07}	0.039	0.265	7.2×10^{-10}	1.0×10^{-14}
Fetus	DCA	1.4×10^{-05}	1.5×10^{-03}	0.013	1.9×10^{-05}	1.7×10^{-07}
	MRD	5.3×10^{-08}	0.183	0.261	1.6×10^{-18}	7.5×10^{-22}
Neonate	DCA	4.0×10^{-04}	1.2×10^{-03}	1.1×10^{-03}	1.5×10^{-05}	1.1×10^{-06}
	MRD	8.1×10^{-08}	0.176	0.530	4.3×10^{-18}	3.1×10^{-19}
Juvenile	DCA	5.3×10^{-06}	2.4×10^{-03}	0.016	3.2×10^{-06}	5.5×10^{-08}
	MRD	5.3×10^{-08}	0.077	0.145	9.5×10^{-21}	7.9×10^{-23}
Adult	DCA	2.6×10^{-06}	2.3×10^{-03}	8.8×10^{-03}	1.3×10^{-06}	1.7×10^{-08}
	MRD	4.8×10^{-08}	0.241	0.405	1.7×10^{-22}	9.1×10^{-24}


25 77 78 86 92 104 162 167
296 371 400 414 535 722 937 1215
Expressed Genes (%)
0 10 20 30 40 50 60 70 80 90 100
Lethal_Singleton
Viable_Singleton
Lethal_Duplicate (DCA)
Viable_Duplicate (DCA)
Lethal_Duplicate (MRD)
Viable_Duplicate (MRD)
Oocyte
Unfertilized Ovum
Zygote
Clevage
Morula
Blastocyst
Egg_Cylinder
Gastrula

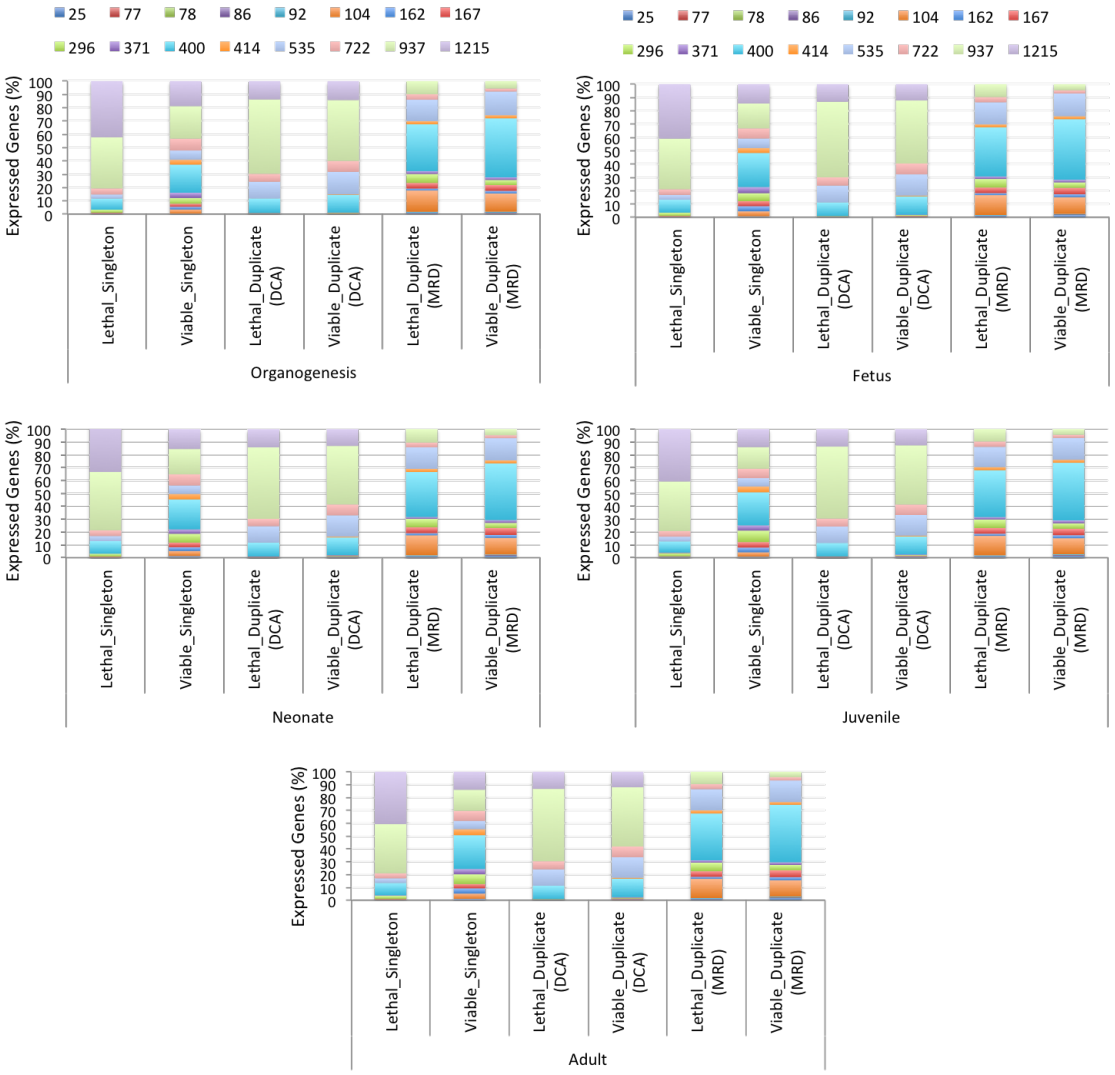


FIGURE 5.5: Percentages of lethal singletons, lethal duplicates, viable singletons and viable duplicates for different age groups across 13 stages of mouse development. Here, age of mouse duplicates and singletons was calculated based on DCA, MRD and SCA ages, respectively.

5.3 Small Scale Duplicates (SSDs) are Older than Whole Genome Duplicates (WGDs)

In a given genome, duplicate genes can be generated either by small–scale (mostly single gene) duplication (SSD) or large–scale duplication events. The most extreme large–scale gene duplication event is the whole genome duplication (WGD) yielding the duplication of the entire genome. Previous studies already revealed a distinct difference between duplicate genes in yeast resulting from SSD and WGD mechanisms (Hakes et al., 2007; Fares et al., 2013), with WGD–derived genes being less essential and functionally more similar than SSD–derived genes. In addition, the authors demonstrated that the deletion of duplicate genes derived through WGD process produces less deleterious effects. We subdivided mouse duplicate genes in our datasets as SSD duplicates and WGD duplicates (mentioned in section 2.1.3) to examine their differences in evolutionary age over development.

Comparing different MRD age groups between small–scale and whole–genome duplicates expressed at each developmental time point revealed that genes duplicated by SSD tend to be older than those duplicated by WGD (Figure 5.6). A significantly greater proportion of genes duplicated by SSD were found to have evolutionary ages between 1215 and 937 MYA. A majority of the WGD genes were 535 and 400 MYA old. Table 5.4 shows the statistical significance of these observations. Moreover, we observed that lethal SSDs are more likely to be older than viable SSDs, lethal WGDs and viable WGDs (Figure 5.7). The similar trend was also observed while considering DCA age for mouse duplicates (Figure 5.8).

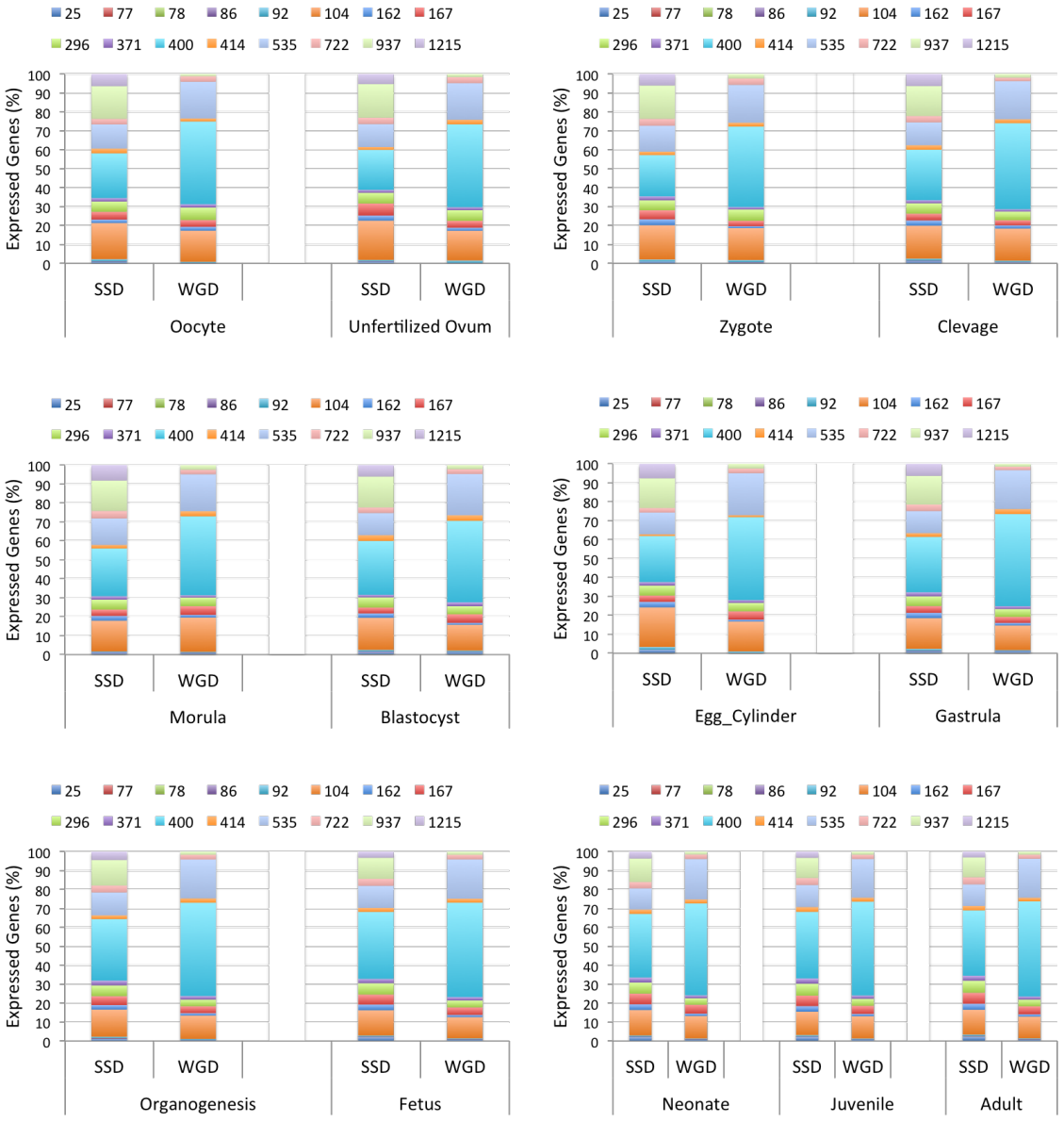
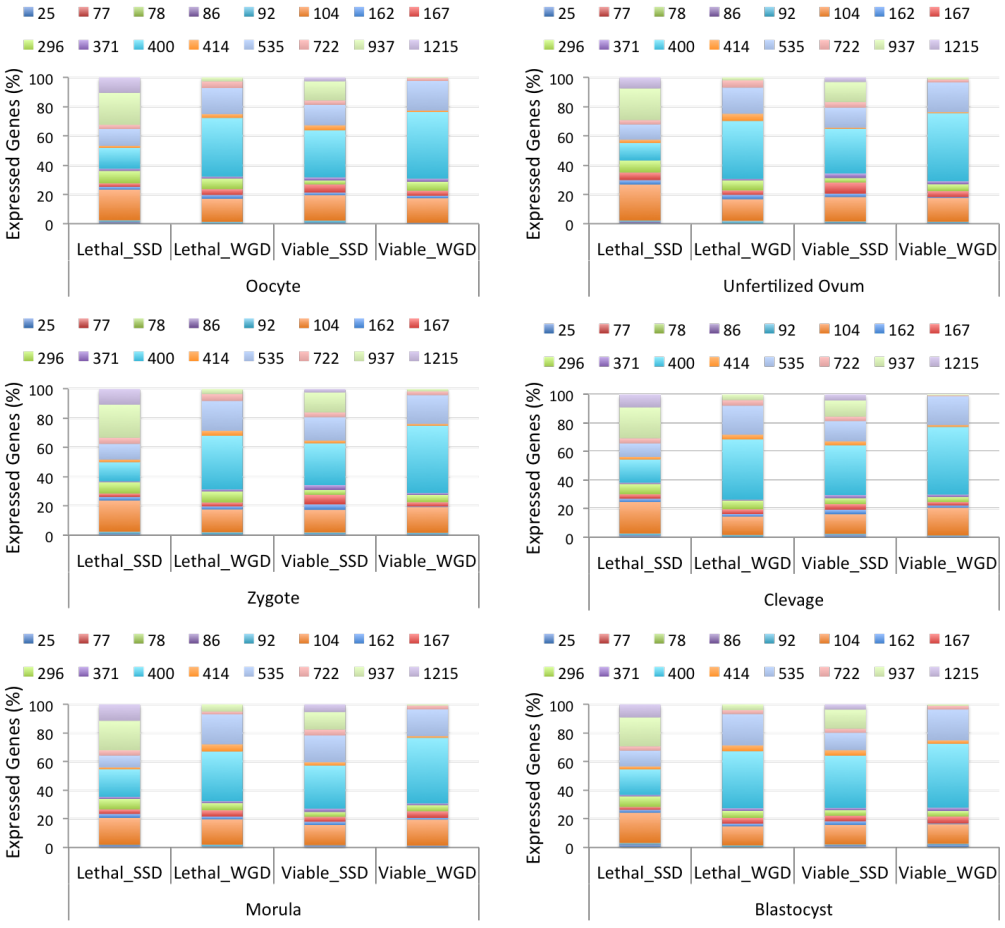


FIGURE 5.6: Percentages of small–scale duplicates (SSD) and whole–genome duplicates (WGD) for different age groups across 13 developmental stages. Here, age of mouse duplicates was calculated based on the Most Recent Duplication (MRD) event.

TABLE 5.4: Chi–squared test results highlighting most significant differences between SSD and WGD gene proportions for MRD age groups across 13 developmental stages.

Developmental stages	Age					
	400	**414**	**535**	**722**	**937**	**1215**
Oocyte	2.8×10^{-7}	0.336	0.014	0.982	2.2×10^{-15}	1.1×10^{-7}
Unfertilized ovum	5.9×10^{-6}	0.465	0.039	0.909	6.6×10^{-9}	2.9×10^{-4}
Zygote	4.4×10^{-7}	0.816	0.033	0.821	1.9×10^{-11}	1.4×10^{-6}
Cleavage	8.5×10^{-7}	0.755	0.002	0.148	2.9×10^{-13}	3.4×10^{-8}
Morula	1.8×10^{-5}	0.439	0.041	0.199	6.9×10^{-11}	6.4×10^{-9}
Blastocyst	1.0×10^{-5}	0.881	7.7×10^{-6}	0.727	1.3×10^{-16}	1.3×10^{-9}
Egg cylinder	6.4×10^{-4}	0.973	0.008	0.957	9.1×10^{-6}	8.4×10^{-5}
Gastrula	3.6×10^{-8}	0.449	8.4×10^{-5}	0.088	1.9×10^{-16}	6.7×10^{-10}
Organogenesis	9.9×10^{-10}	0.544	1.3×10^{-6}	0.136	2.5×10^{-24}	2.9×10^{-12}
Fetus	3.6×10^{-10}	0.977	1.6×10^{-10}	0.031	1.0×10^{-26}	2.7×10^{-13}
Neonate	1.9×10^{-9}	0.766	8.9×10^{-11}	0.149	2.7×10^{-27}	2.5×10^{-12}
Juvenile	4.7×10^{-10}	0.449	6.3×10^{-10}	0.021	2.7×10^{-25}	6.43×10^{-13}
Adult	2.4×10^{-12}	0.501	2.6×10^{-11}	0.007	1.6×10^{-26}	5.4×10^{-13}



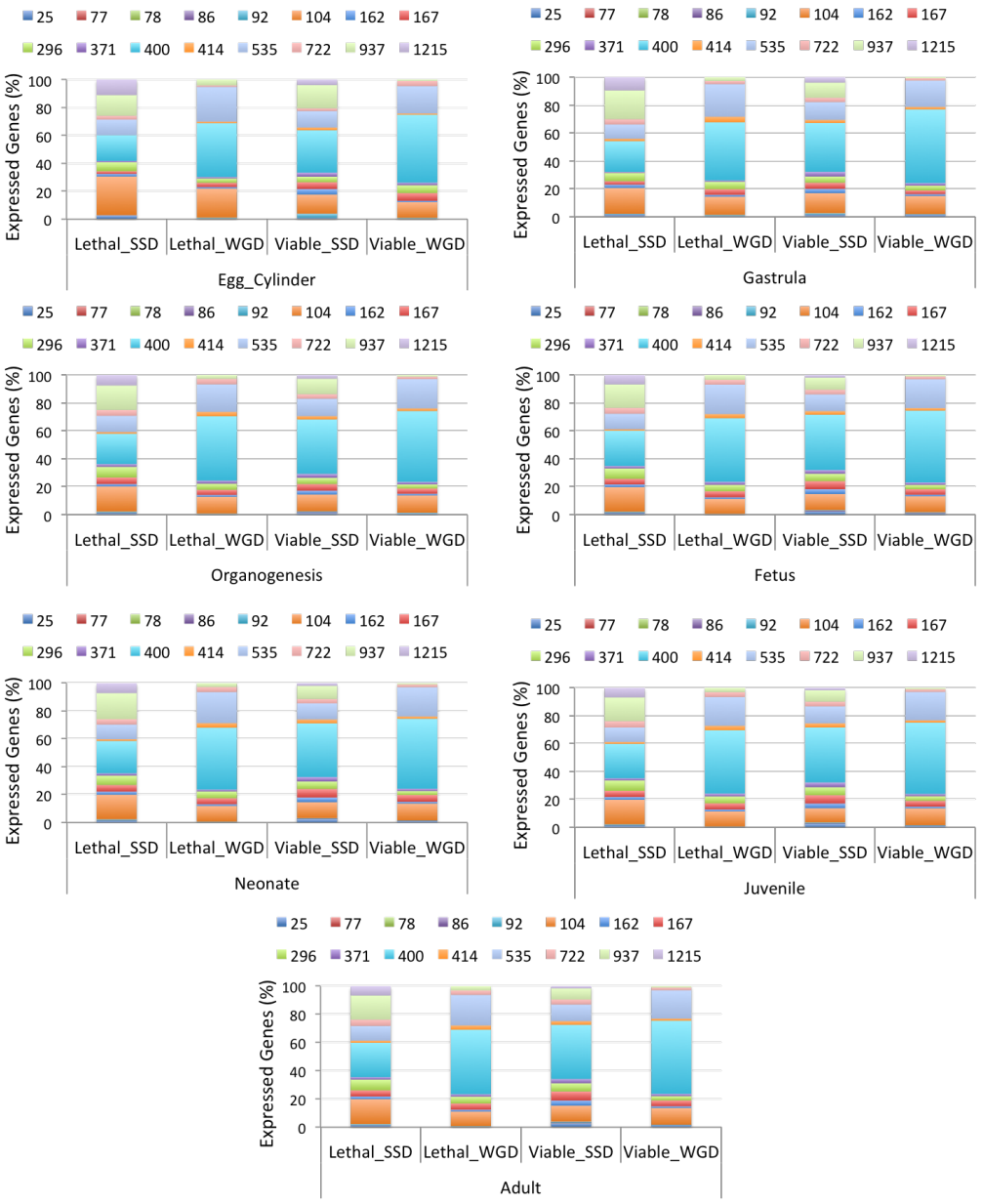
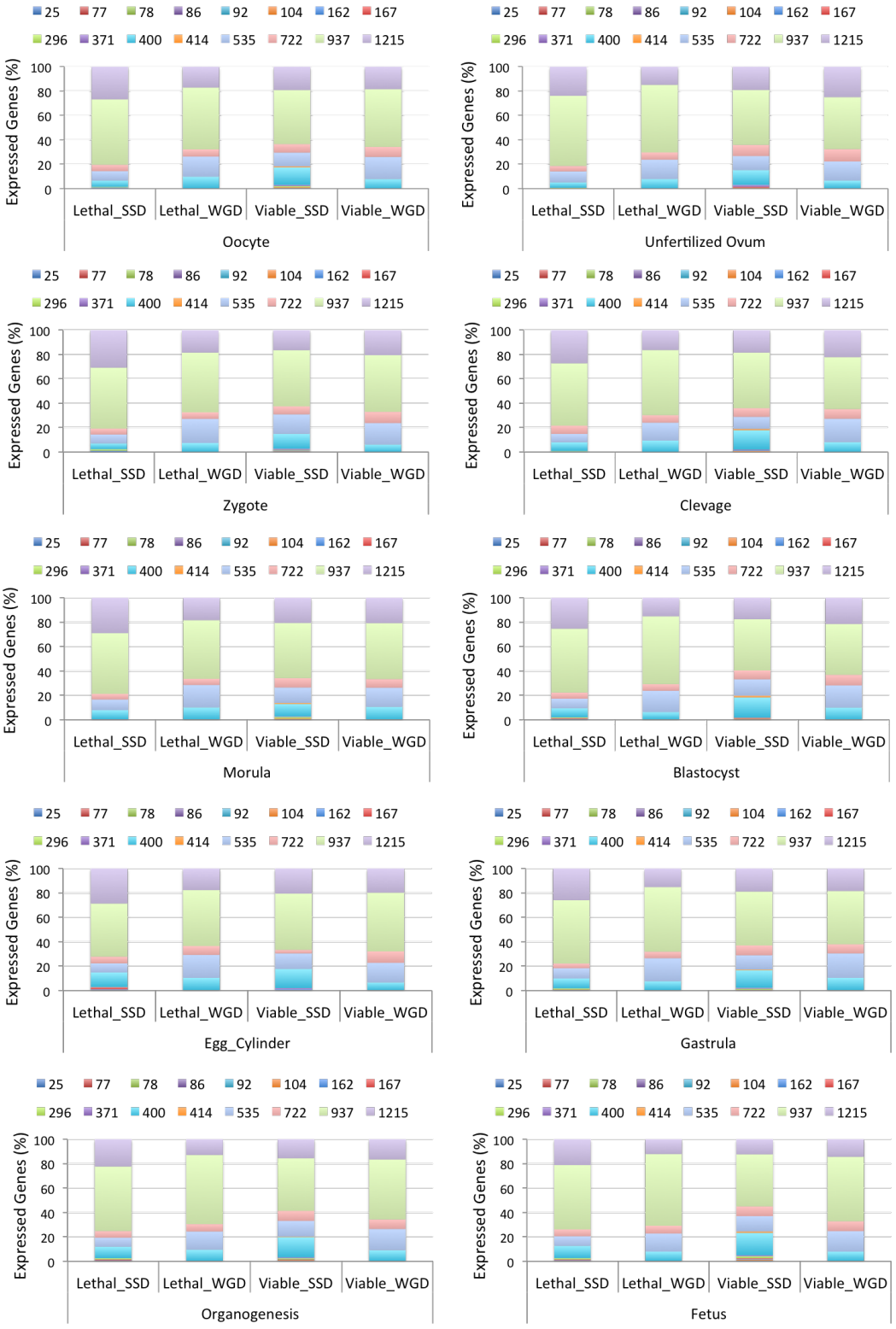


FIGURE 5.7: Percentages of lethal SSD, lethal WGD, viable SSD and viable WGD genes for different age groups across 13 stages of mouse development. Here, age of mouse duplicates was calculated based on the Most Recent Duplication (MRD) event.


25 77 78 86 92 104 162 167
296 371 400 414 535 722 937 1215
Expressed Genes (%)
100 80 60 40 20 0
Lethal_SSD Lethal_WGD Viable_SSD Viable_WGD
Oocyte
Unfertilized Ovum
Zygote
Clevage
Morula
Blastocyst
Egg_Cylinder
Gastrula
Organogenesis
Fetus

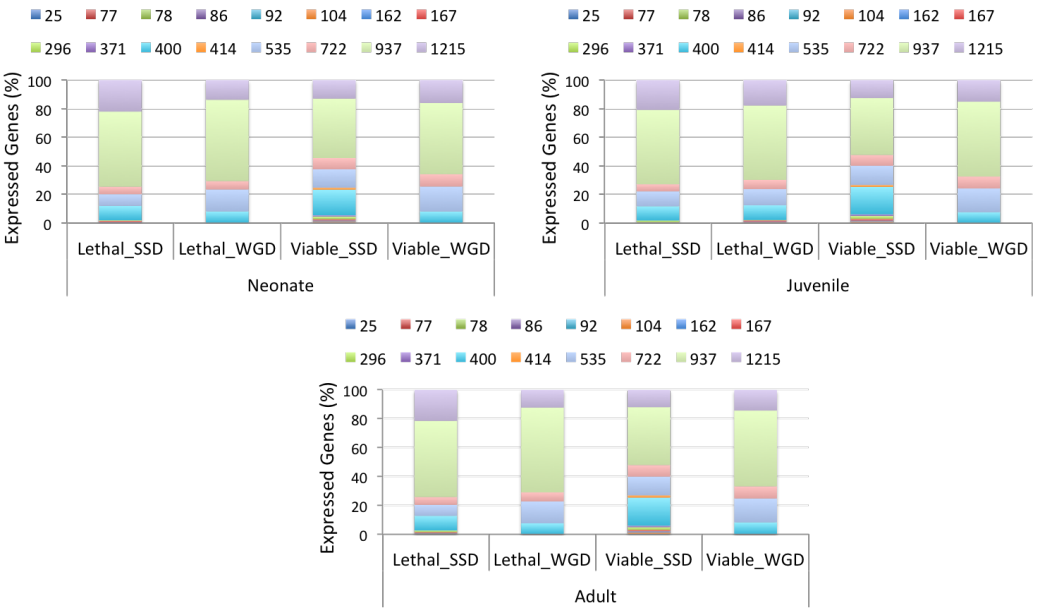


FIGURE 5.8: Percentages of lethal SSD, lethal WGD, viable SSD and viable WGD genes for different age groups across 13 stages of mouse development. Here, age of mouse duplicates was calculated based on the Duplicate Common Ancestor (DCA).

5.4 Developmental Expression Patterns

Differences in gene expression (TPM) patterns of lethal, viable, singleton and duplicate genes were compared over 13 stages of mouse development to address the following questions:

- Do expression patterns of lethal and viable mouse gene vary over embryonic development?
- Do expression patterns of mouse singleton and duplicate vary over embryonic development?
- Do duplicate pairs where both genes are lethal show a greater divergence in developmental expression patterns?

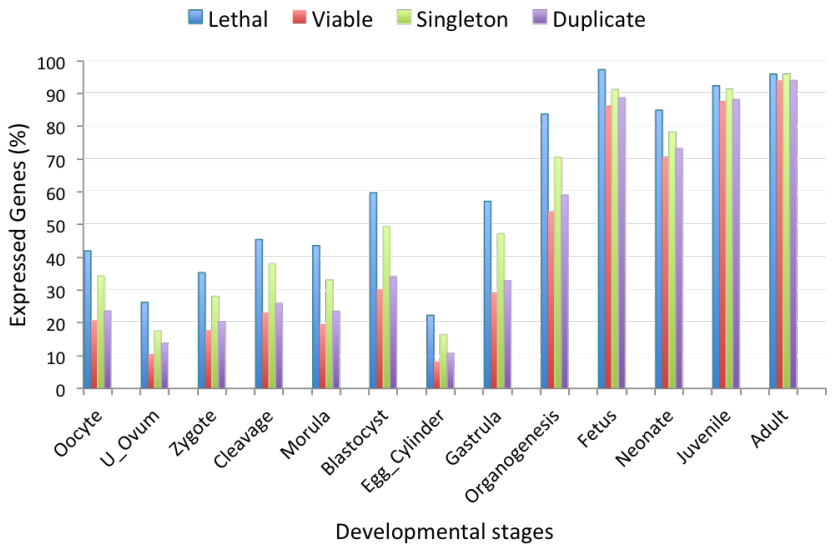


FIGURE 5.9: Frequencies (%) of lethal, viable, singleton and duplicate mouse genes expressed at 13 stages of mouse embryonic development

This analysis revealed that a significantly greater proportion of lethal genes are expressed than viable genes at almost every developmental stage. Similarly, we found a higher proportion of expressed singletons than duplicate genes during development. Figure 5.9 shows the proportions of lethal, viable, singleton and duplicate genes that are expressed at different developmental stages. The Bonferroni corrected Chi–squared test did not find statistical significance of these differences at later developmental stages, presumably because all genes are nearly always expressed at these stages (Table 3.3 and Table 5.5). Viable and duplicate genes were often found in the group with zero transcripts while looking at differences in gene expression distributions.

We further examined the differences in developmental expressions between

TABLE 5.5: Differences in proportions of expressed mouse singleton and duplicate genes. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.00385.

Developmental stages	**Singleton(%)**	**Duplicate(%)**	**p–value**
Oocyte	34.38	24.65	1.5×10^{-10}
Unfertilized ovum	17.57	13.76	2.3×10^{-3}
Zygote	28.15	20.48	6.4×10^{-7}
Cleavage	38.06	25.92	6.1×10^{-12}
Morula	33.11	23.62	1.3×10^{-8}
Blastocyst	49.47	34.12	2.4×10^{-14}
Egg cylinder	16.44	10.68	3.5×10^{-7}
Gastrula	47.22	32.88	2.7×10^{-13}
Organogenesis	70.65	59.01	5.1×10^{-6}
Fetus	91.22	88.69	0.406
Neonate	78.30	73.19	0.071
Juvenile	91.29	88.16	0.315
Adult	95.95	93.99	0.543

TABLE 5.6: Differences in proportions of lethal singleton versus lethal duplicate and viable singleton versus viable duplicate mouse genes. The p–value for the Bonferroni correction is 0.00385. Highlighted cells in yellow represent statistically significant differences based on Bonferroni corrected p–value of 0.00385.

Developmental stages	**Lethal**			**Viable**		
	Singleton(%)	**Duplicate(%)**	**p–value**	**Singleton(%)**	**Duplicate(%)**	**p–value**
Oocyte	50.60	36.63	1.4×10^{-4}	24.58	19.66	0.008
Unfertilized ovum	30.48	23.63	0.018	9.88	10.82	0.479
Zygote	41.04	31.75	0.007	20.72	17.06	0.037
Cleavage	54.38	39.88	1.8×10^{-4}	28.67	21.75	3.7×10^{-4}
Morula	52.39	38.13	1.3×10^{-4}	21.81	19.19	0.146
Blastocyst	72.91	51.50	1.0×10^{-6}	35.90	28.77	1.4×10^{-3}
Egg cylinder	24.70	20.88	0.171	11.45	7.60	9.4×10^{-4}
Gastrula	67.73	50.50	5.8×10^{-5}	34.94	27.61	6.7×10^{-4}
Organogenesis	89.04	80.50	0.101	59.88	52.58	0.014
Fetus	98.01	96.88	0.844	87.59	86.35	0.756
Neonate	89.04	82.38	0.208	72.29	70.53	0.594
Juvenile	96.61	89.75	0.205	88.92	87.44	0.694
Adult	98.80	94.13	0.403	94.22	93.99	0.957

those singleton and duplicate genes that are lethal. We observed a greater percentage of lethal singletons being expressed at different developmental stages (especially during oocyte, cleavage, morula, blastocyst and gastrula) compared to lethal duplicates. In addition, a significantly high percentage of viable singleton genes were shown to be expressed in four crucial stages: cleavage, blastocyst, egg cylinder, and gastrula. We summarise these results in Figure 5.10 and Table 5.6.

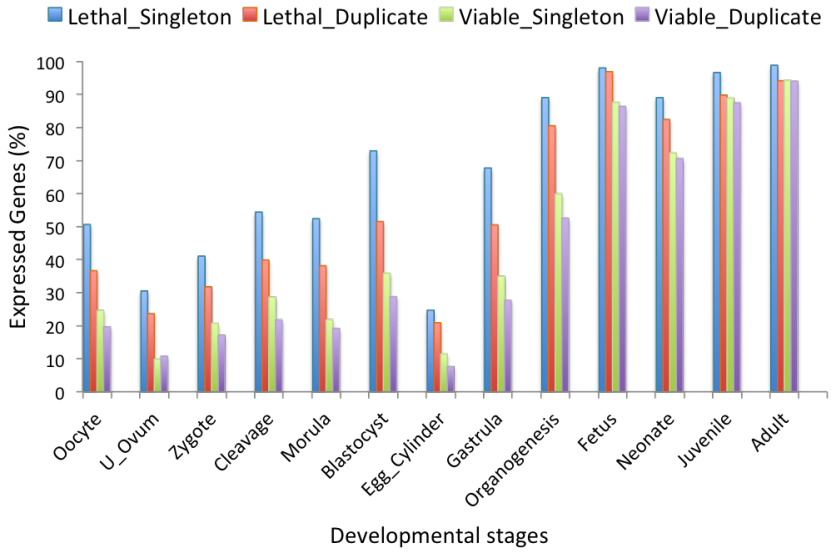


FIGURE 5.10: Frequencies (%) of lethal singleton, lethal duplicate, viable singleton and viable duplicate mouse genes expressed at 13 stages of mouse embryonic development

5.5 Developmental Co–expression Analysis

Since duplicate genes often exhibit different patterns of gene expression (Gu et al., 2002; Makova and Li, 2003), it is likely that the similarity of their expression profiles could mediate the functional compensation. Therefore, we examined the expression profiles of lethal and viable mouse gene pairs in 13 developmental stages to test the hypothesis that duplicates with developmental co–expression are more likely to be viable and duplicates without developmental co–expression are more likely to be lethal. To investigate all plausible duplication paths, we made three distinct gene pairs from our lethal and viable datasets, namely lethal–lethal, lethal–viable and viable–viable gene pairs. We used the Manhattan and Euclidean distance methods (described in section 2.2.1.2) to compute the expression difference (co–expression) between each mouse gene and its pair from their expression

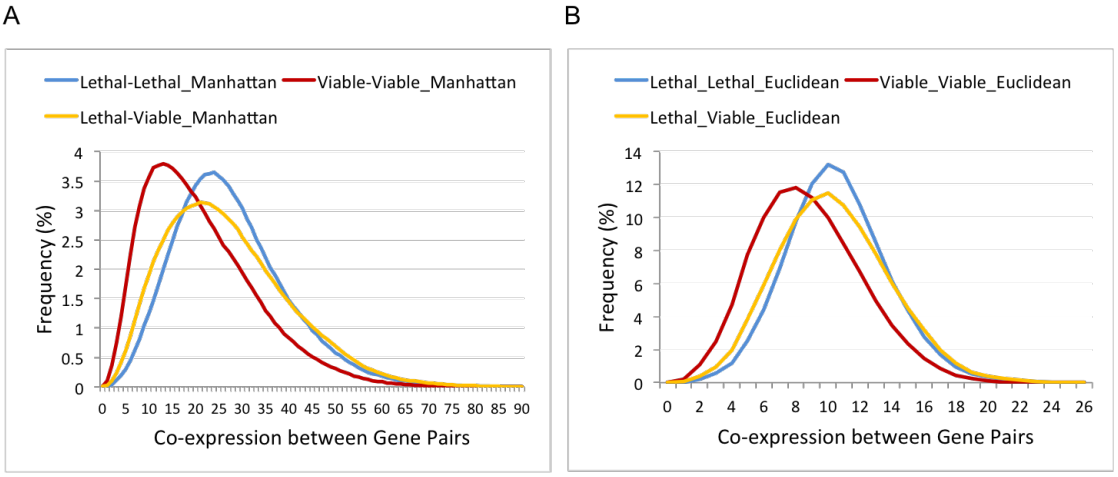


FIGURE 5.11: Differences in co–expression between all gene pairs within a class of essentiality over 13 embryonic developmental stages. Co–expression was computed using the Manhattan (A) and Euclidean (B) distance method.

across 13 developmental stages and consequently, developed three co–expression matrices (lethal–lethal: 1301×1301; viable–viable: 3451×3451; lethal–viable: 1301×3451). Figure 5.11 shows the distributions of lethal–lethal, viable–viable and lethal–viable gene co–expression. As small Manhattan and Euclidean distances refer to higher co–expression, we observed that lethal–lethal gene pairs tend to have lower co–expression compared to viable–viable pairs. Analysis of co–expression between minimum distance gene pairs further showed the low developmental co–expression for lethal–lethal gene pairs (Manhattan distance: p–value $= 5.49 \times 10^{-152}$; Euclidean distance: p–value $= 8.83 \times 10^{-133}$). Figure 5.12 shows the distributions of co–expression between minimum distance pairs.

Furthermore, we obtained lethal–lethal, lethal–viable and viable–viable duplicate gene pairs comparing our lethal and viable gene lists with the human orthologues of mouse duplicates listed by Makino and McLysaght (2010), which include 9,057 human duplicate pairs. All the human duplicate pairs listed in (Makino

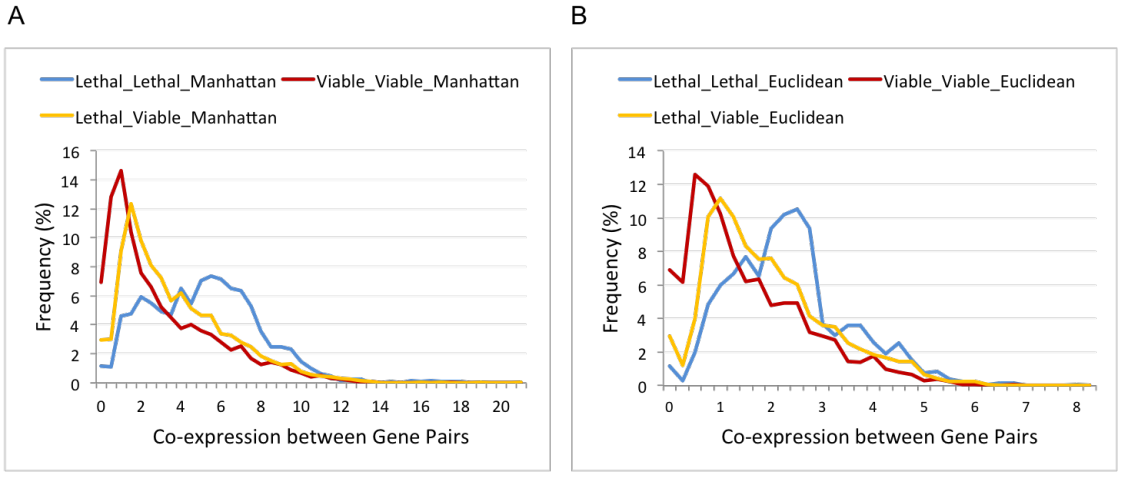


FIGURE 5.12: Differences in co–expression between all minimum distance gene pairs within a class of essentiality over 13 embryonic developmental stages. Co–expression was computed using the Manhattan (A) and Euclidean (B) distance method.

and McLysaght, 2010) were the whole–genome duplicate pairs. We used these duplicate pairs as we assumed that the orthologous mouse genes were duplicates as well, which was then confirmed by our BLAST searches. In addition, it was not possible to directly recompute WGD pairs from the recent release of the Ensembl database. Investigating the Manhattan and Euclidean distances between these gene pairs concluded that developmentally co–expressed duplicates are more likely to be viable (Manhattan distance: p–value $= 2.15 \times 10^{-9}$; Euclidean distance: p–value $= 2.26 \times 10^{-8}$). Figure 5.13 displays the co–expression distribution of this analysis.

In addition, we conducted all–against–all Blastp searches to detect duplicate pairs within our datasets. A mouse gene was considered as a duplicate if it had hits to other mouse genes within our datasets with E–value $< 10^{7}$. For each query protein sequence its closest mouse paralogue was identified. Using lethal

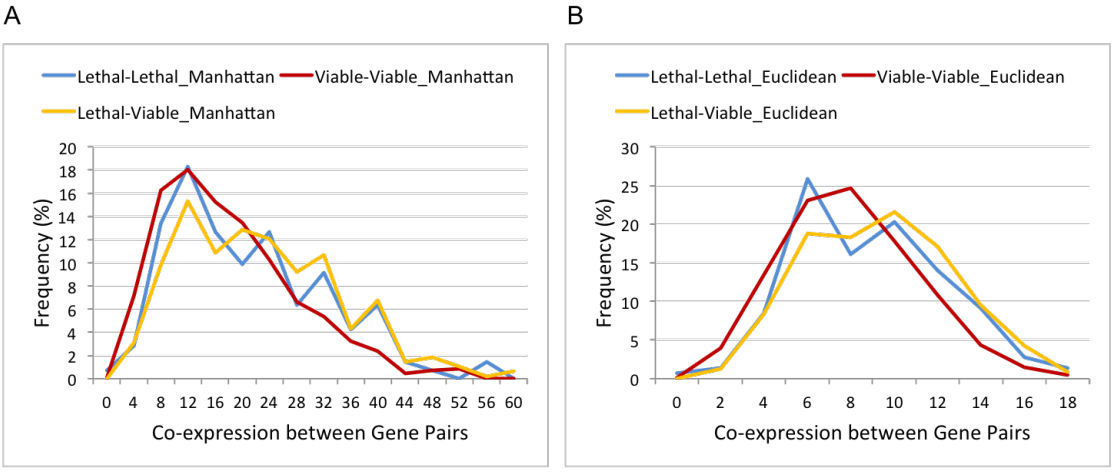


FIGURE 5.13: Differences in developmental co–expression between all gene pairs obtained from (Makino and McLysaght, 2010). These are mouse orthologues of human genes. Co–expression were computed using the Manhattan (A) and Euclidean (B) distance method.

proteins as queries and lethal proteins as a database, we found 535 (41%) lethal–lethal duplicates. Using viable proteins as queries and viable proteins as database, we obtained 1,748 (52%) viable–viable duplicates. Moreover, a total of 2,489 (52%) lethal–viable or viable–lethal duplicates were found from the complete gene lists. Analysis of developmental co–expression within each class of these gene pairs further revealed that viable–viable duplicate pairs tend to have more similar expression during development, whereas lethal–lethal duplicate pairs tend to have greater divergence of expression (Manhattan distance: p–value $= 2.35 \times 10^{-34}$; Euclidean distance: p–value $= 1.57 \times 10^{-30}$), with lethal–viable pairs in between. Figure 5.14 shows this observation.

We further labelled 173 lethal–lethal, 546 lethal–viable and 649 viable–viable gene pairs duplicated by SSD among all duplicate pairs obtained from the Blast search. A total of 194 lethal–lethal, 520 lethal–viable and 343 viable–viable WGD

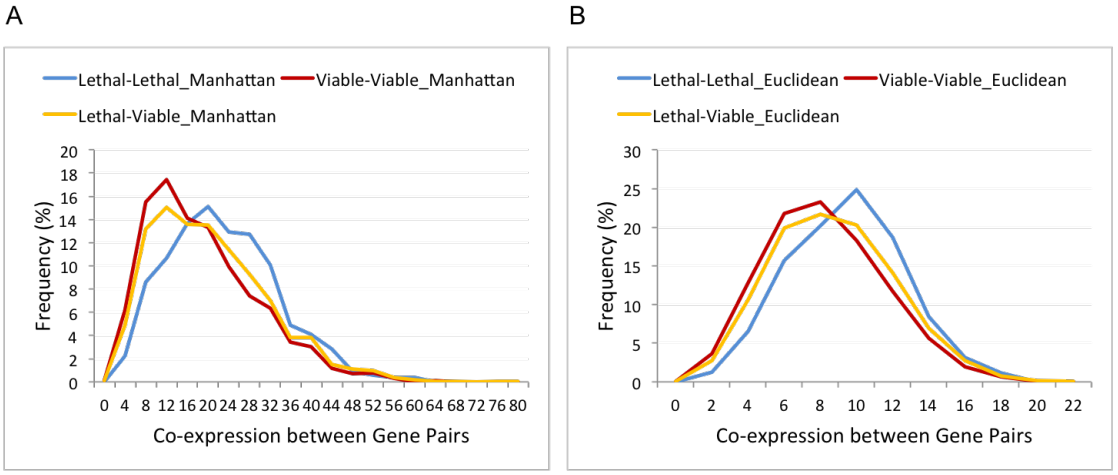


FIGURE 5.14: Differences in co–expression over embryonic developmental stages for duplicate gene pairs obtained by the Blast search. Co–expression were computed using the Manhattan (A) and Euclidean (B) distance method.

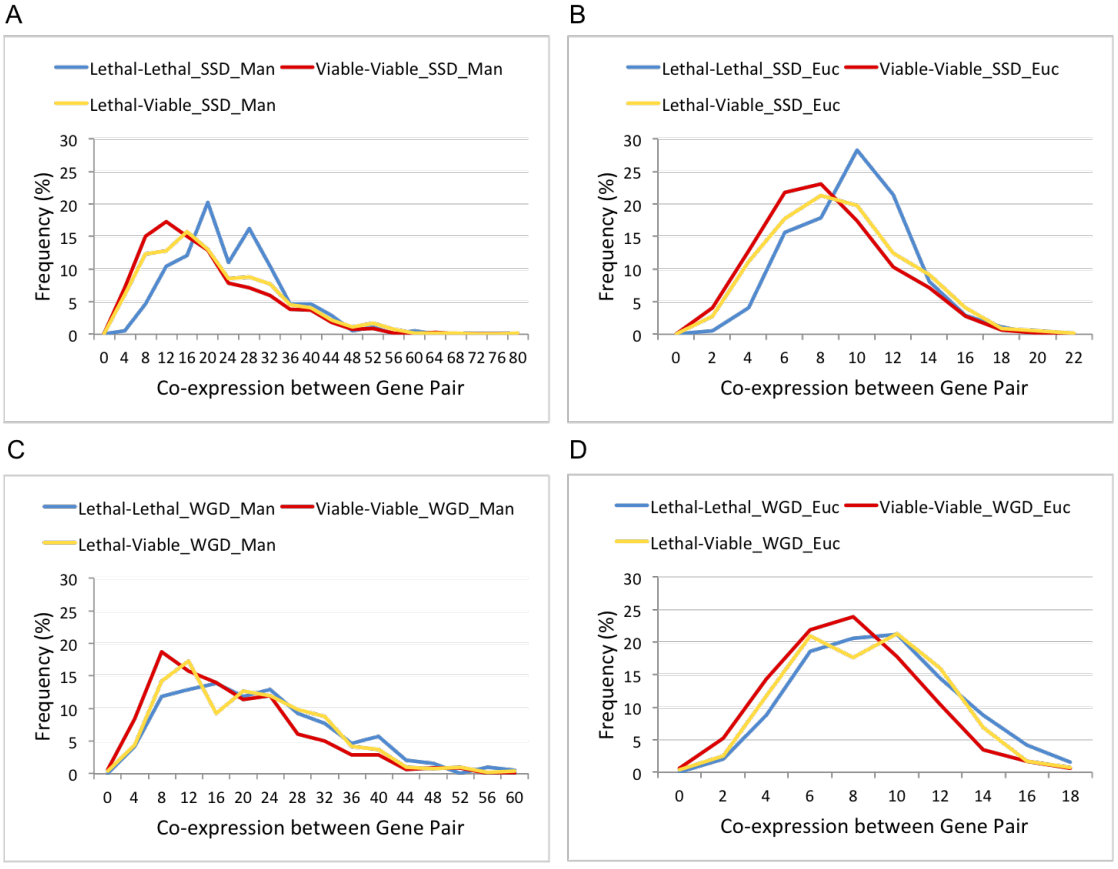


FIGURE 5.15: Differences in co–expression for SSD (A–B) and WGD (C–D) duplicate gene pairs over embryonic developmental stages. Co–expression were computed using the Manhattan (A, C) and Euclidean (B, D) distance method.

TABLE 5.7: Statistical test results signifying similar co–expression between viable–viable duplicate pairs with respect their duplication mode.

Duplicate Gene Pair	p–value	
	Manhattan Distance	**Euclidean Distance**
SSD	5.22×10^{-9}	5.38×10^{-8}
WGD	1.50×10^{-5}	4.70×10^{-5}

gene pairs were identified. Analysis of the Manhattan and Euclidean distances between these gene pairs showed that viable–viable gene pairs have more similar co–expression during development irrespective of their duplication mode (Figure 5.15). Table 5.7 shows the statistical significance of this result. However, SSD pairs are more likely to have greater divergence of expression compared to WGD pairs.

Moreover, Euclidean distance analysis on the normalised expression data (see section 2.2.1.2) further showed that duplicate gene pairs whereby both members are lethal tend to have a lower level of co-expression during development than pairs with at least one viable member (Figure 5.16).

All these analyses validate our hypothesis that duplicate gene pairs with closer developmental co–expression are more likely to be viable.

5.6 Evidence for the Developmental Hourglass Pattern in Mammals

As mentioned earlier (section 2.2.1.2), our analysis exploited experimental mouse expression data from NCBI UniGene database (Stanton et al., 2003) stratified into 13 developmental stages. A total of 19,310 mouse genes (including lethal, viable

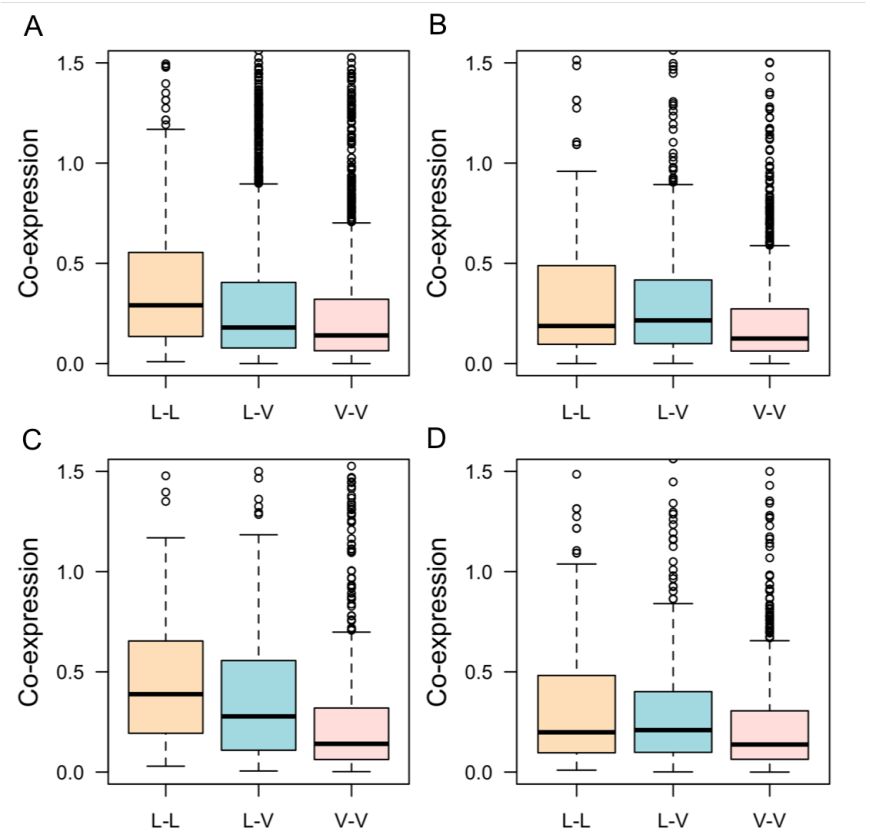


FIGURE 5.16: Differences in developmental co–expression for duplicate gene pairs obtained from the Blast search (A), (Makino and McLysaght, 2010) (B), all SSD pairs (C) and WGD pairs (D). Here, Co–expression were computed using the Euclidean distance method. Euclidean distance were measured considering the normalised expression data

and unknown genes) had UniGene cluster expression data, which were used to define their developmental expression pattern. The evolutionary age reported in millions of years ago (MYA) of the duplicate common ancestor (DCA) or the most recent duplication (MRD) event was assigned to each mouse duplicate gene. Non–duplicated genes were assigned the age of their single common ancestors (SCA).

The oldest genes were approximately 1215 MYA old, whereas youngest genes belonged to the class Murinae and were approximately 25 MYA old. Previous studies (Domazet-Lošo and Tautz, 2010; Piasecka et al., 2013; Drost et al., 2015)

demonstrated that genes expressed in mid–development (the phylotypic stage) are more likely to have an older evolutionary origin than genes expressed in early and late development, thereby reporting an developmental hourglass pattern. We reconfirmed that this pattern is found during mouse development by comparing differences in ages between early, phylotypic and late stages in mouse development. Adopting the definition of the phylotypic period in mouse from previous studies (Irie and Kuratani, 2011; Bogdanović et al., 2016), we labelled all developmental stages before gastrula as the early developmental period and all stages after organogenesis as the late developmental period of mouse. The phylotypic period thus comprised two stages: gastrula and organogenesis. We observed that mouse gene expression patterns do conform to an hourglass shape while plotting the mean evolutionary ages for all mouse genes expressed at each developmental time point, regardless of whether the evolutionary age of duplicated genes is calculated based on the DCA (Figure 5.17A) or MRD (Figure 5.17B).

This initial analysis did not factor in gene expression levels when assigning the mean evolutionary age to each developmental time point. Prior studies (Domazet-Lošo and Tautz, 2010; Drost et al., 2015) of the hourglass model used the transcriptional age index (TAI), which multiplies gene age by expression level at each developmental stage, to produce a weighted index of evolutionary age. TAI was defined as the mean evolutionary age of a transcriptome at a given developmental stage (Domazet-Lošo and Tautz, 2010). We also calculated the TAI values for each stage of mouse development to determine if weighting by expression level

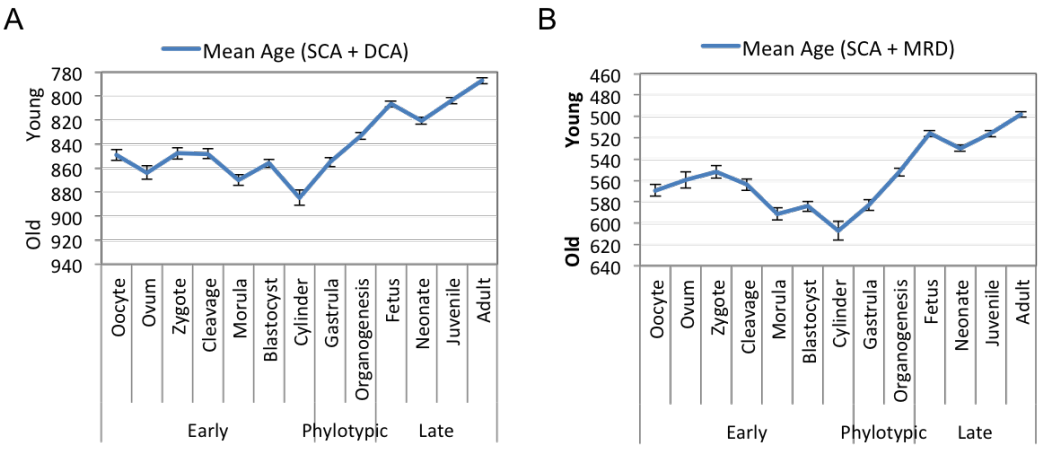


FIGURE 5.17: Mammalian gene expression exhibits a developmental hourglass pattern. Plotting the mean mouse gene age against developmental time points reveals an hourglass pattern. Here, ages of mouse duplicates were calculated based on the Duplicate Common Ancestor (DCA, panel A) or Most Recent Duplication (MRD, panel B). For singleton genes, the age of their single common ancestor (SCA) was used.

would affect our observations. The higher the value of TAI, the older the evolutionary origin of the gene is. Figure 5.18A and Figure 5.18B show TAI profiles for (SCA+DCA) and (SCA+MRD) age along with the corresponding standard deviation estimated by the bootstrap analysis. Use of DCA and MRD ages in the TAI calculation again revealed an hourglass pattern of mammalian gene expression.

Concerns have been raised that calculating the TAI results in a bias towards an older evolutionary age, as housekeeping genes expressed at high levels tend to be more ancient (Piasecka et al., 2013). A method to mitigate this bias involves removing the top 10% of most highly expressed genes from the TAI calculations. When we applied this modification to our calculations, we again observed that both the DCA ages (Figure 5.18C) and MRD ages (Figure 5.18D) conformed to an hourglass pattern.

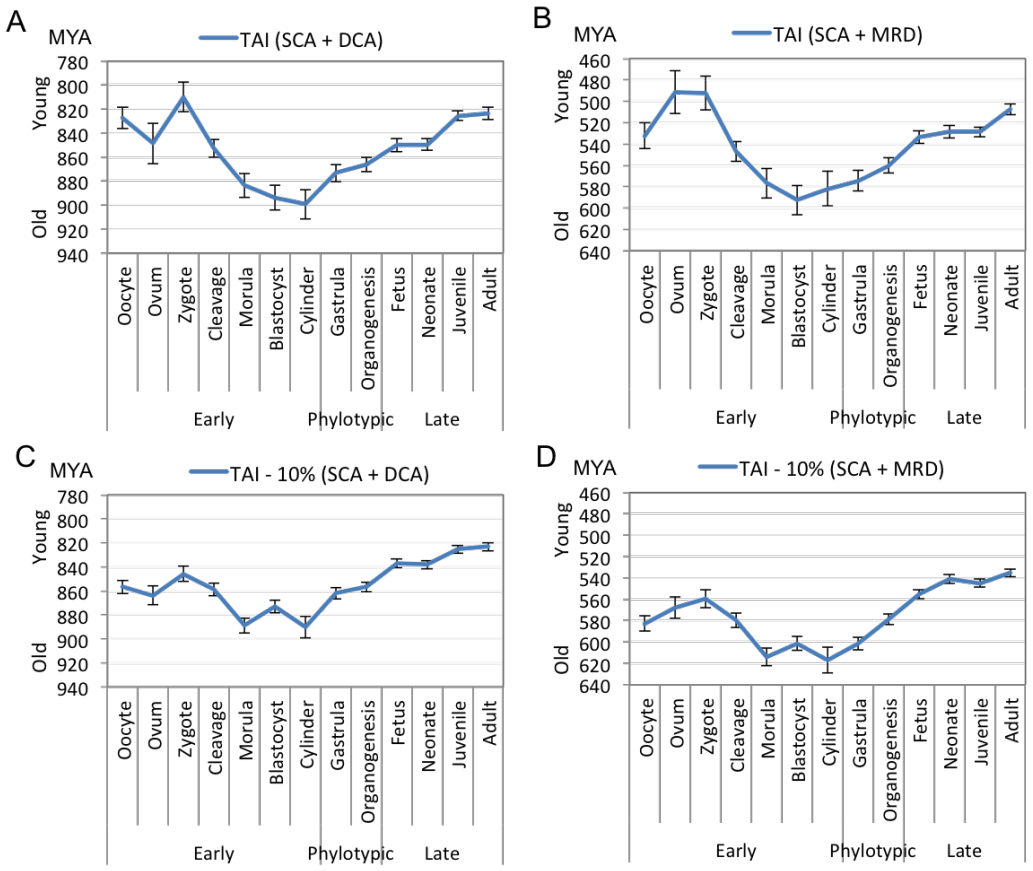


FIGURE 5.18: Transcriptional age index (TAI) analysis. Calculating the transcriptional age index (TAI) reveals a developmental hourglass pattern for SCA+DCA mouse gene ages (A) and for SCA+MRD mouse gene ages (B). Removing the top 10% of the most highly expressed genes does not affect the hourglass pattern for SCA+DCA (C) or SCA+MRD (D) gene ages. Error bars represent the standard deviation of TAI (A–D), estimated by bootstrap error calculations. The y–axis represents taxon gene age groups in millions of years (MYA) before present.

Visually, the overall mean and TAI profile of MRD and DCA age shows an hourglass shape, whereby mouse genes expressed in middle stages of development have older ages compared to genes expressed in the early or later stages. Although all of our analysis showed the existence of an hourglass pattern in the mouse developmental expression data, we were surprised to find that in every analysis the evolutionarily oldest genes are not expressed at the defined phylotypic stage

(gastrula and organogenesis). The mean age analysis demonstrated that genes expressed during organogenesis are likely to be younger than those expressed at the egg cylinder stage (DCA age: ANOVA p–value = 7.50×10^{-13}; MRD age: ANOVA p–value = 1.56×10^{-8}). Using the TAI, we found that older genes were expressed during the egg cylinder stage than at the organogenesis stage (DCA age: ANOVA p–value = 5.44×10^{-92}; MRD age: ANOVA p–value = 1.10×10^{-304}). Similar findings were seen when the top 10% of most highly expressed genes were removed from the TAI dataset (DCA age: ANOVA p–value = 1.18×10^{-242}; MRD age: ANOVA p–value = 5.82×10^{-168}). In addition, plotting the distribution of gene ages at each developmental stage confirms that the egg cylinder stage is enriched for the oldest genes (Figure 5.19 and 5.20). Thus, we find that oldest mouse genes are expressed prior to the phylotypic stage of development.

It has been proposed that genes expressed at the phylotypic stage encode essential functions, and thus will be less tolerant to mutation (Irie and Kuratani, 2014). We, therefore, investigated the proportion of lethal (essential) and viable (non–essential) genes expressed at each stage of mouse development. We found that the developmental stage expressing the greatest proportion of essential genes is the egg cylinder stage, rather than the phylotypic stages of gastrulation and organogenesis (Figure 5.21). Overall, the egg cylinder stage is more likely to be enriched for the oldest and essential genes.

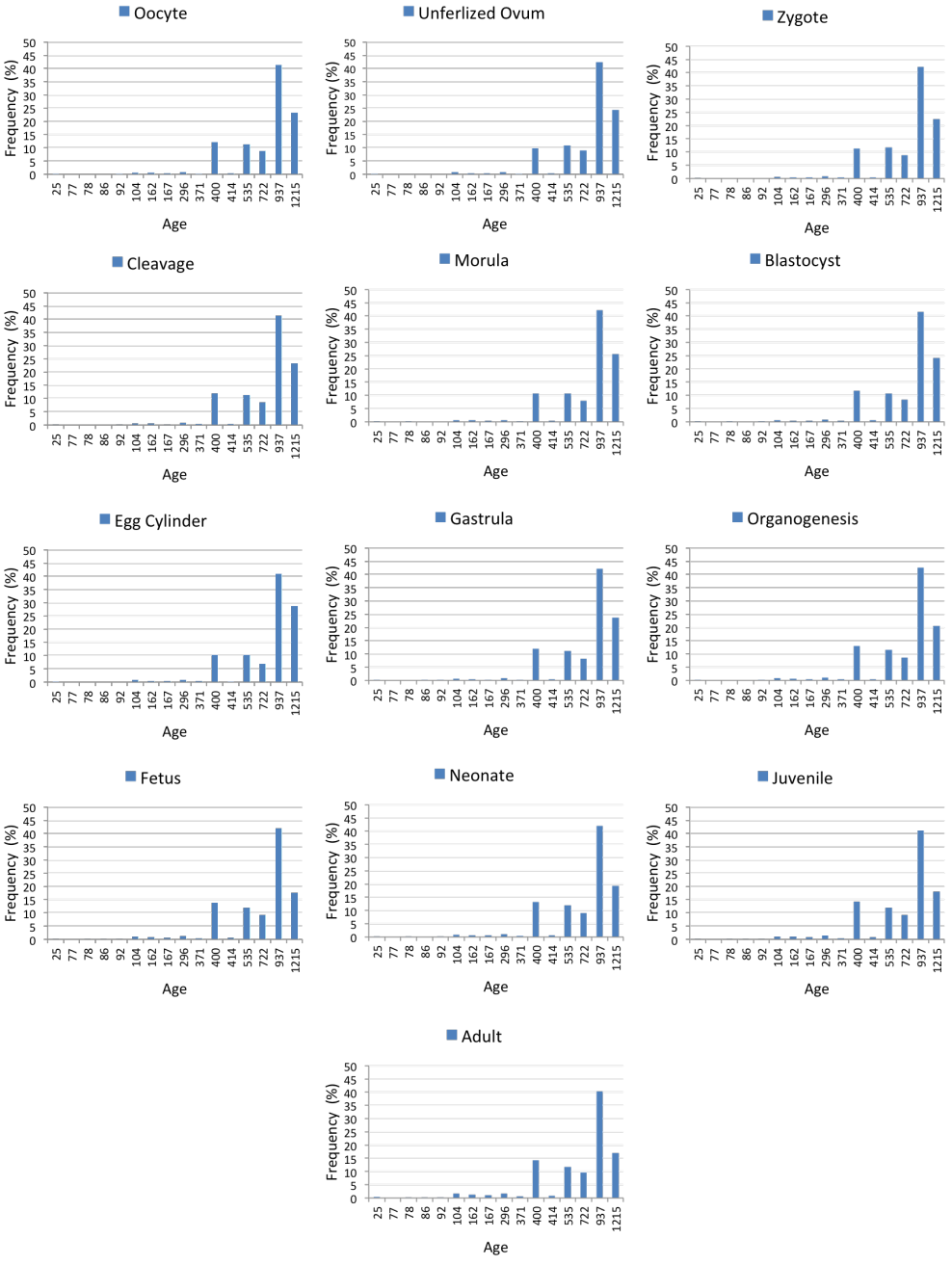


FIGURE 5.19: Distributions of SCA+DCA gene age over 13 stages of mouse development. The youngest and oldest genes are 25 MYA and 1215 MYA old, respectively.

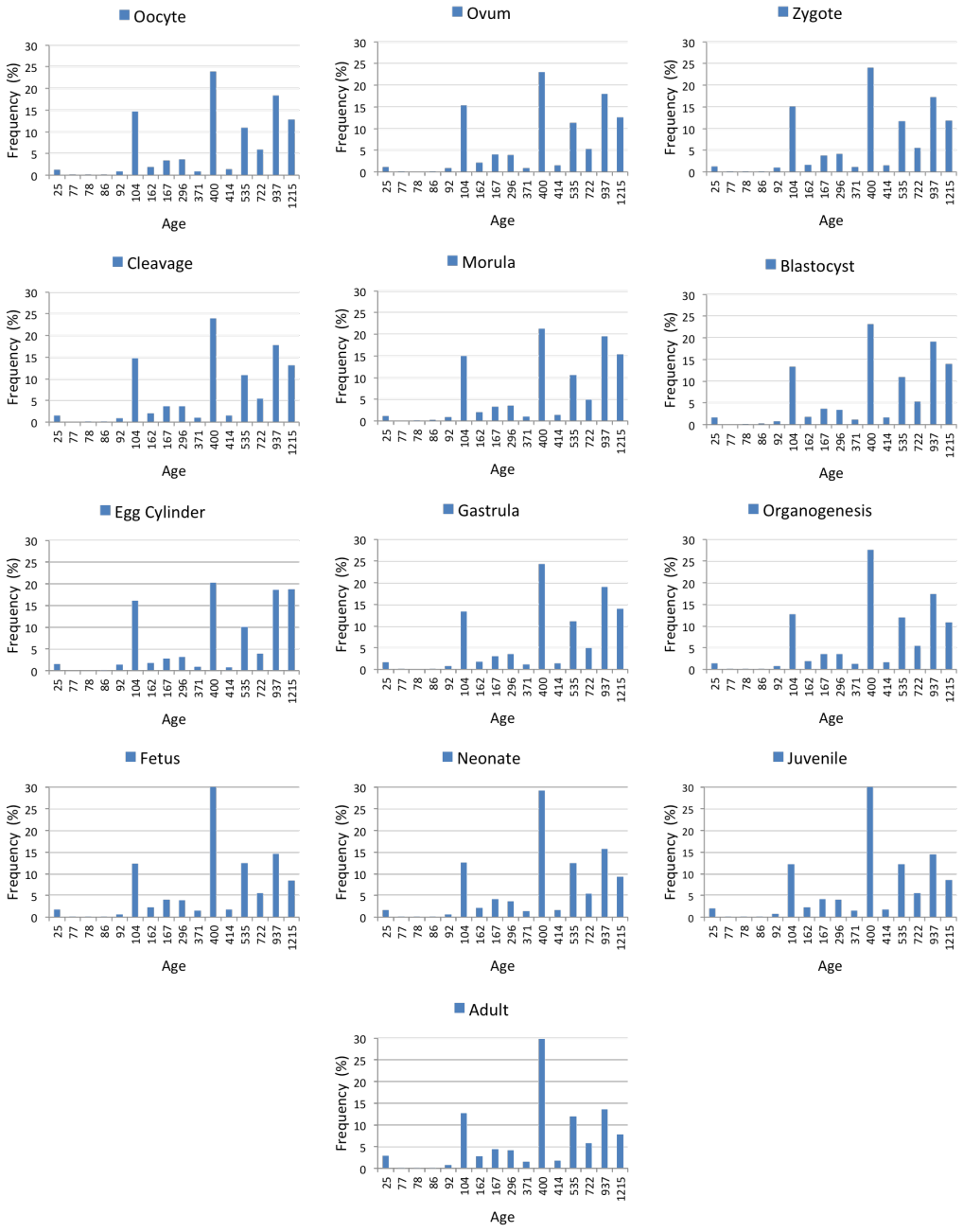


FIGURE 5.20: Distributions of SCA+MRD gene age over 13 stages of mouse development. The youngest and oldest genes are 25 MYA and 1215 MYA old, respectively.

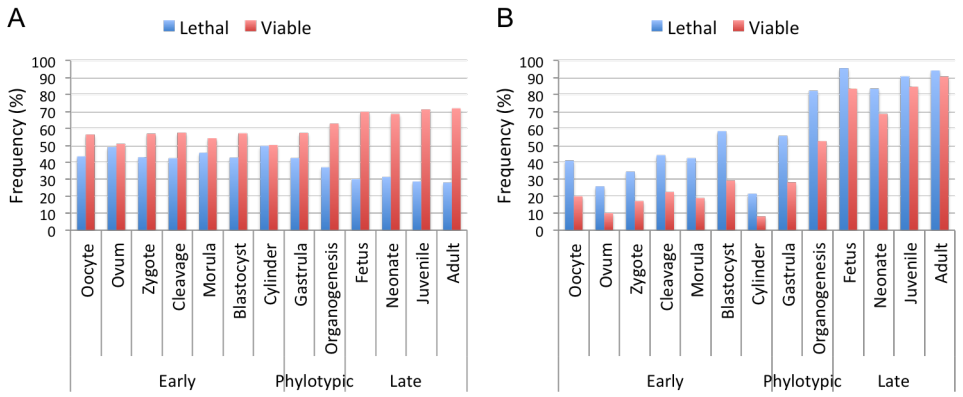


FIGURE 5.21: (A) Percentage of expressed mouse genes that are lethal (essential) and viable (non–essential) for each developmental stage. (B) Proportion of lethal (essential) and viable (non–essential) mouse genes that are expressed at each developmental stage. A gene is considered as expressed at a particular developmental stage if it has a non–zero TPM value for that stage.

5.7 Discussion

The relationship between duplication and essentiality for mammalian genes

In model organisms like worm, yeast and mouse, phenotypic effects of single–gene deletion were examined on a genome–wide scale. Of particular interest are essential genes, whose deletion or mutation results in lethal phenotypes. Many expressed genes, which have critical molecular functionality, are non–essential. For this, it is plausible that the functional loss of gene deletion could be compensated by the presence of a duplicate gene in the same genome with similar expression and function. Gene duplication is a frequent event in eukaryotes, which is the origin of such functional redundancy. There has been enduring interest in understanding the correlations among gene function, phenotypic changes caused by gene deletion,

and gene duplication. A priori, duplicated genes are reported less likely to be essential than non-duplicated genes (singletons) in yeast (Gu et al., 2003). However, studies in mouse knockouts did not confirm this expected trend, where singletons were just as essential as duplicated genes (Liang and Li, 2007; Liao and Zhang, 2007). This contradictory observation can only partly be explained by some experimental biases. It cannot highlight the factors that mediate some duplicates being essential, some being non–essential.

We herein investigated in detail the contribution of mouse duplicate gene to essentiality. We report that duplicate genes in mouse indeed tend to be less essential than singletons. We analysed the evolutionary age of mouse genes to test whether knockout genes with an early evolutionary origin are more likely to be essential. Evolutionary age analysis confirms results from a prior study (Chen et al., 2012b) that lethal (essential) genes are more ancient than viable (non–essential) genes. In addition, it is observed that genes with older evolutionary origins are always more likely to be essential, regardless of being singletons or duplicates. We further report that lethal and singleton genes are more likely to be expressed during development. Expressed singletons were found to have an older evolutionary origin than duplicates expressed at the same stage of development. Moreover, genes duplicated by small–scale duplication (SSD) events were found being more likely to be older than those duplicated by the whole–genome duplication (WGD). We also explored the effects of expression profile similarities between duplicate gene pairs across 13 stages of mouse development on the functional compensation. We

hypothesised that duplicated gene pairs with similar developmental co-expression tend to be viable, otherwise lethal. Consistent with our hypothesis, we found that mouse duplicate gene pairs whereby both members are lethal tend to have divergent expression patterns during development than pairs with at least one viable member. The greater similarity in expression during development between viable duplicate pairs suggests that they contribute more to functional redundancy than lethal duplicates, irrespective of their duplication mode.

However, we have used the Manhattan and Euclidean distance to measure the expression profile similarities between duplicate gene pairs. As we were concerned that the analysis associated with the Euclidean distance could be affected to a great extent by the scale of the data, the distance was calculated based on the logged expression data as well as on the data which were normalised within the range (0, 1) dividing each expression data by the maximum TPM value. As we are comparing between two different dataset comprising of gene expression values, we need to use normalisation techniques that are independent of the values present in the data. That is why, we could not apply other normalisation techniques, such as the correlation coefficient, unit length scaling or subtracting the mean across conditions in our study to scale the expression data. We need to normalise values across both the datasets to keep the scaling right. That is why, the way we scaled our data is a valid normalisation technique for our analysis.

Mammalian hourglass pattern

The hourglass model of development predicts that genes with the oldest evolutionary origin are expressed at the phylotypic stage in many species. Here, we report that in mouse embryos, the oldest genes are not expressed at the morphologically defined phylotypic stage, but instead at an earlier time point. This earlier time point, the egg cylinder stage, also expresses the greatest proportion of essential genes. As essential genes have been noted to possess an older evolutionary origin, it is logical to find that the stage with the oldest genes also contains the greatest proportion of essential genes. Our findings question whether morphological similarities amongst species of the same phyla are universally encoded by the oldest, most conserved and essential functions.

Our findings may be due to specific features of mammalian development. In the early stages of mammalian development, the embryo consists of cells of two lineages: the epiblast, which will form the embryo, and the trophoblast, which will contribute to the placenta. Given that the placenta is a mammalian innovation, genes that are expressed in the trophoblast would be expected to have a younger evolutionary age. Developmental stages up to and including the blastocyst stage include genes expressed in both cell lineages, and thus include genes with a recent evolutionary origin. At the egg cylinder stage we see an increase in the number of expressed genes with the oldest evolutionary age. At this developmental time point the mouse embryo acquires a dramatically different morphological shape, with a simultaneous increase in cell proliferation (Skreb et al., 1991; Bedzhov

et al., 2014b), dependent on the highly conserved mTOR and TGFbeta pathways (Bedzhov et al., 2014a). Thus, the shift in TAI at the egg cylinder stage may arise from a requirement for evolutionarily conserved genes regulating cell proliferation.

We also considered whether genes expressed throughout development (found in all 13 stages analysed) could influence the hourglass pattern. We found a total of 291 genes that are expressed at all 13 stages, with an average age of 912 MYA when using the DCA age and an average age of 591 MYA when using the MRD age. The average age of these genes is beyond the range of variation at most developmental time points, suggesting that these universally expressed genes are not a strong source of the pattern detected.

5.8 Summary

Our analysis demonstrated that developmental co–expression and gene evolutionary age contribute towards determining essentiality of duplicated mouse genes. In addition, we found that singleton genes in mouse are more likely to be essential and older than duplicates. These results reveal new insights into the relationship of gene essentiality, developmental expression, and gene duplication. Moreover, we did find evidence that mouse genes expressed in early and late gestation have a more recent evolutionary origin than those expressed in mid-gestation. However, we found that the genes with the oldest evolutionary origin were not expressed at the defined mammalian phylotypic stage, but instead at the egg cylinder stage, prior to the mammalian phylotypic stage. Future studies on the mechanisms of

generating morphological patterns during development are needed to determine whether mammalian structures are programmed by evolutionarily older genes at early developmental time points, or whether the conserved morphology of the mouse phylotypic stage is not achieved by expression of the most ancient genes.

Chapter 6

Discussion

Identification of essential genes in mammals is one of the central concerns of developmental biology as it assists in the identification of key cellular processes and tissue-specific functions that are crucial for life. Existing methods for mammalian essential gene identification mostly rely on mouse knockout experiments (White et al., 2013), which require great time and cost to carry out. In contrast, computational methods, which use gene attributes to evaluate essentiality, offer a rapid and low-cost means of predicting essential genes (Zhang et al., 2016). In this research, we aimed to characterise and predict essential (**lethal**) and non–essential (**viable**) genes in the mouse in terms of genomic features, protein sequence–based features, gene expression, evolutionary age and protein–protein interaction (PPI) information. Similar studies have been undertaken to determine human disease genes (Kondrashov et al., 2004), human drug targets (Bakheet and Doig, 2009; Bull and Doig, 2015), bacterial drug targets (Bakheet and Doig, 2010), and to

identify features of essential genes in bacteria, yeast and mouse (Saha and Heber, 2006; Gustafson et al., 2006; Seringhaus et al., 2006; Hwang et al., 2009; Acencio and Lemke, 2009; Yuan et al., 2012). Many features are found to be associated with gene essentiality in the mouse. A novel machine learning based computational method is subsequently developed using these gene features which allows for predicting mouse genes that are essential for the survival of embryos during gestation. In addition, we explored the expression profile of mouse singletons and duplicate genes to study their relation with mouse gene essentiality. Our final study aimed to examine the existence of the morphological hourglass pattern during mouse development.

This chapter summarises the overall findings of this thesis and discusses their relevance. This chapter concludes with highlighting some limitations and future research directions.

6.1 Discriminating Features between Mouse Essential and Non–essential Genes

The success of predicting essential genes based on gene features mainly depends on the prediction influence of these quantifiable features. If all features are comparable between the lethal and viable gene groups, then they cannot distinguish gene essentiality. Hence, there needs to be numerous characteristics that are divergent between lethal and viable genes. We identified a total of 75 features that discriminate lethal and viable genes with statistical significance (Chapter 3). These

features, expressing different traits of mouse biology, are interrelated. Many of these features are compatible with those of previous studies on yeast and bacteria, but have not been verified in mammals yet. In addition to previously evidenced features, we found a number of important novel features that are strongly related to essential genes.

We found a number of genomic features for which lethal and viable genes had significantly different values: lethal genes are shown to be significantly longer in length and have a higher number of transcripts, a higher number of exons and longer length of exons compared to viable genes. A probable explanation could be that the functions they perform may involve different complex proteins that could possibly have multiple domains to contribute diverse cellular functionalities (Brocchieri and Karlin, 2005). Lethal genes also tend to have a significantly longer length of introns and a lower GC content. Intron and exon length is known to vary inversely with GC content (Gazave et al., 2007; Zhu et al., 2009). GC content also has correlation with gene length (Duret et al., 1995).

In terms of evolutionary age, we observed that lethal genes are more ancient than viable genes. This result corresponds to a previously reported fact that essential genes are evolutionarily more conserved (Giaever et al., 2002; Jordan et al., 2002). This is because essential genes are likely to be involved in basic cellular processes, therefore the negative selection acting on them is more severe than for non–essential genes. Moreover, a significantly greater proportion of lethal genes are found to be expressed at the earlier stages of mouse development compared

to viable genes. This result makes sense as mouse genes, expressed during earlier development, will be required for further viability of the embryo and therefore might produce more severe phenotypes if mutated. Lethal genes are further shown to exhibit a high level of gene expression. This is logical since more expression means more protein molecules around to have a larger effect. This result is also supported by previous studies which showed that highly expressed genes are likely to be essential and evolve slowly (Pál et al., 2003; Drummond et al., 2005).

We observed that lethal genes are more likely to have critical roles in different cellular processes that are central to life. These include the development of embryo, tissue, heart, nervous system, brain, lung, respiratory tube and blood vessel, cell morphogenesis, cell division, cell proliferation, cell differentiation, DNA replication, DNA repair and transcription regulation. These results are consistent with the finding that lethal proteins are frequently localised to the nucleus. In addition, lethal genes have a tendency to perform functions related to protein binding, DNA binding, ATP binding and nucleic acid binding. In contrast, viable genes are more likely to perform those functions that are related to a cells response to its environment (*e.g.* transporter activity, signal transducer activity, lipid binding and immunoglobulin binding) and to participate in processes like regulation of apoptosis, behaviour, cell communication, ion transport, cell signalling, immune system development, homeostasis and response to stress.

Another set of features that distinguished lethal and viable genes is the cellular components of their protein products. One noticeable trend is the localisation

of a greater proportion of lethal proteins in the nucleus. Results from annotations (from the UniProt database and GO terms analysis) and WoLF PSORT predictions confirmed this. This is reasonable, because proteins involved in essential cellular processes such as DNA replication, transcription and DNA repair mostly locate in the nucleus (Kumar et al., 2002). In contrast, viable proteins are mostly extracellular or membrane–bound. The high proportion of viable genes in the extracellular region is consistent with the presence of signal peptide cleavage sites and fibronectin type III (fn3) domain in their sequence and also by their signal transducer activity. The fn3 protein domain is an evolutionarily conserved domain that is generally found in animal proteins, especially in extracellular proteins. Its main function is to mediate cell–cell signaling or interactions. The high percentage of membrane–bound viable proteins is further confirmed by the presence of a larger number of transmembrane helices. Since most membrane proteins participate in transport and metabolic related processes, this explains why membrane proteins are more likely to be non–essential. These processes are not critical for the survival of embryo. Viable proteins are also more likely have Src Homology 2 (SH2), Src Homology 3 (SH3), ion transport, and Epidermal Growth Factor (EGF) domains. This further establishes the propensity of viable proteins being membrane–bound as these evolutionary conserved protein domains are common constituents of membrane proteins.

We observed many of the protein sequence features were associated with

mouse lethal genes. The significant features were: long protein length; high molecular weight; high frequencies of A, D, E, K, Q and S amino acids; high frequencies of polar, charged and basic residues; low frequencies of aliphatic, aromatic and non-polar residues. Lethal proteins have the tendency of being longer in length because essential proteins are evolutionarily more conserved and conserved proteins are mostly longer (Lipman et al., 2002). In addition, longer proteins mean more possible domains to mediate a wider range of functions and more protein–protein interactions. Moreover, the presence of more polar, charged and basic residues reflects why lethal proteins tend not to be membrane–bound. Furthermore, lethal proteins are more likely to function as Ligase and Transferase enzymes, whereas viable proteins are mostly functionally less critical Hydrolase enzymes. Ligases and Transferases perform more complex chemistry than Hydrolases.

In terms of post–translational modifications, we found that lethal proteins are more likely to be phosphorylated and acetylated. This is reasonable, as phosphoproteins have critical roles in almost all cellular processes and acetylated proteins are important for regulating gene expression and protein–protein interactions (Arnesen, 2011). Lethal proteins are also likely to have the zinc finger, C4 type (zf–C4) domain, which correlates with their likelihood to function as transcription factors. Most occurrences of the zf–C4 domain are found within the DNA–binding regions of many nuclear receptors that function as transcription factors.

Similarly to yeast and bacteria, mouse essential genes are shown to play crucial roles in the protein–protein interaction (PPI) networks. The dominant network

features for lethal proteins are: high degree, short average shortest path (ASP) length, high values of closeness centrality, betweenness centrality, clustering coefficient, BottleNeck (BN), Edge Percolation Component (EPC) and Maximum Neighbourhood Component (MNC). The correlation between highly connected proteins (high degree) and gene essentiality have already been reported in previous research (Yu et al., 2004; Kim et al., 2006). Shorter ASP length and high values of closeness centrality and clustering coefficient shows that lethal proteins can quickly transfer information to other reachable protein nodes in the PPI Network. Moreover, high values of betweenness centrality, BN, EPC and MNC signify the likelihood of lethal proteins to function as hubs or bottleneck.

The presence of similar protein sequences might potentially bias decisions drawn from using a dataset. Therefore, redundant proteins were removed from our lethal and viable datasets using different levels of sequence similarities to verify the importance of the features distinguishing lethal and viable genes in the original dataset. Analysis of non–redundant datasets confirmed the trends seen on the full dataset. Overall, this part of the study is concluded with validating our research hypothesis that, mammalian essential genes are significantly different from non–essential genes by a number of features. This dependency on various features implies that multiple aspects of biology unite to make a gene either essential or non–essential.

6.2 Performance of Essential Gene Prediction

Given a large number of sequence and functional characteristics that vary between lethal and viable genes in mouse, we aimed to use them to construct a supervised machine learning based computational method that could accurately determine whether a mouse gene better fits the profile of a lethal gene or viable gene. Our target was to complement the existing experimental techniques with *in silico* predictions of gene essentiality. Machine learning methods utilise the features of a gene group to learn patterns that are specific to that group and make predictions on the basis of that.

We constructed three balanced training datasets using 102 features with an equal number of lethal and viable genes in order to achieve a better generalisation, as there are fewer lethal genes than viable genes. Imbalanced data, in general, decrease the performance of machine learning algorithms (Visa and Ralescu, 2005). We assessed the potential effectiveness of different candidate machine learning methods to select the one that demonstrates the best performance for our data. A gene essentiality classifier, therefore, has been built using the Random Forest algorithm (Breiman, 2001) as it has the highest level of prediction accuracy (Chapter 4). In order to make an unbiased prediction about a mouse gene, we used 10–fold cross–validation to assess the performance of our classifier. Cross–validation mitigates the potential for overfitting (Kohavi, 1995). Our classifier has achieved an accuracy of ∼91% and AUC (area under the ROC curve) of ∼0.963, proving its competence in predicting genes essential for mouse development. In addition,

this classifier has successfully made predictions for a separate dataset of known mouse genes that were not included in the training datasets. With a classification accuracy of ∼93% and AUC of ∼0.964, our Random Forest classifier has further proven its efficacy. Moreover, adjustments were made on missing PPI network based properties for some mouse genes, consequently this slightly enhanced the classification power of our classifier (AUC = ∼0.965).

The Information Gain feature selection method was further applied to select a subset of most salient features for accurately classifying lethal and viable genes. The use of a subset of features can speed up the classifier building process and can further improve the classification performance. The Random Forest classifier built on these selected features shows the highest accuracy (AUC = ∼0.971) amongst the ones developed using all features. The selected feature set emphasises the importance of PPI network based features, gene expression levels across development, localisation at the nucleus, gene length, exon length, a number of transcripts, protein length, proportions of polar, basic, and charged residues, gene age, acetylation and phosphorylation for the prediction of gene essentiality. All these features have been shown to significantly discriminate between lethal and viable genes in this research.

Our proposed computational method built on combining the Random Forest algorithm, the information gain feature selection filter and data pre–processing techniques, has displayed substantially better performance in predicting essential genes compared to previous studies (Acencio and Lemke, 2009; Deng et al., 2010;

Yuan et al., 2012; Yang et al., 2014). This classifier is robust as it has the ability to accurately classify genes even the data has missing values. In addition, it runs faster and is resistant to overfitting. Our classifier is also inherently adaptive, capable of incorporating any available experimental or sequence–derived properties. Besides the benefits of using the Random Forest and the feature selection technique, the high classification performance could be due to the improved accuracy of the training datasets because each mouse gene was correctly labelled after manual checking. Moreover, the predictability of gene essentiality has increased due to the integration of PPI network based features, developmental gene expression, evolutionary age, post–translational modifications and some other dominant features that have been evidenced to greatly associated with gene essentiality in previous studies. We suggest that future prediction of mouse essential genes, and by extrapolation human essential genes, will stand a higher chance of success if our classifier is used.

We expect our classifier to serve as a valuable resource for the mouse genetics research community in optimising the time–consuming and costly mouse knockout experiments. Due to the genome similarities between mice and humans, our prediction results might facilitate the identification of human genetic disease candidates and potential drug targets. Biologists detect common chromosomal regions in members of a family affected by a genetic disease to confirm genetic linkage to that chromosomal region. This is achieved by comparing the DNA of the people affected by the disease to the DNA of the people not affected by the

disease. These linked chromosomal regions may contain a large number of genes associated with the disease. Experimental verification is necessary to identify with certainty which gene/genes cause the disease progression. But, instead of carrying out knockout experiments for all these genes, biologists may choose to start with a small set of genes that are more likely to be causing the disease. Our model can be used to minimise the set of the candidate genes for experimental verification. If the disease affects development or viability for the affected individuals, then essential genes predicted by our model can be tested first to find their association with the disease. If the disease is mild (not affecting development or viability), non–essential genes can be tested first to find their association with disease. In both cases, the other set of genes can be tried out if the first set of genes does not reveal any association with the disease. This way, essentiality prediction through our model can help reduce the cost of experimental verification in identifying the cause of genetic diseases.

Moreover, our proposed work might allow developmental biologists to study the function of genes to comprehend how an embryo develops into a mature organism. By definition, the genes we predict as essential will be crucial during embryonic development. As deficiencies in developmental genes are often connected to birth defects, our findings will be informative for the biologists to gain insight into different congenital birth defects.

6.3 Correlation between Gene Essentiality and Duplication

Previously, a number of studies have attempted to understand the function of gene duplicates and their associations with phenotypic changes caused by gene deletion. By inference, it is reasonable to guess that genes with duplicates are less likely to be essential due to their analogous function and expression, and thus deleting a duplicate gene could result in mild or even no phenotypic effect. Support for the propensity of gene duplicates to be less essential has been shown for *S. cerevisiae* (Gu et al., 2003) and *C. elegans* (Conant and Wagner, 2004). However, it was observed that duplicate genes and singletons are equally likely to be essential in mouse (Liao and Zhang, 2007; Liang and Li, 2007). Contrarily, it was found that duplicate genes have critical roles in the genetic robustness of human (Hsiao and Vitkup, 2008). This contradiction questions what factors define essentiality in singletons and duplicates.

We herein showed that the proportion of mouse lethal genes among singletons is much higher than among duplicates. This indicates that, similarly to other organisms, duplicate genes in mouse are less likely to be essential than singletons. Further analysis with evolutionary age revealed that lethal genes are predominantly older than viable genes. In particular, genes with older evolutionary origins are more probable to be essential, irrespective of their duplication status. This result matches a recent analysis by (Chen et al., 2012b). In addition, we observed that lethal genes and singletons are more prone to be expressed across

development. These genes tend to have an older evolutionary origin compared to viable and duplicate genes expressed at the same stage of development. After investigating the mode of duplication, we found that duplicate genes generated by the SSD event tend to be more ancient than those generated by the WGD. This is reasonable as WGD–derived genes are functionally more similar and are less critical than SSD–derived genes (Hakes et al., 2007; Fares et al., 2013).

Moreover, by investigating expression level similarities across development, we observed that mouse duplicate pairs with similar developmental co–expression are more likely to be viable and those with divergent expression patterns tend to be lethal. This is reasonable, as duplicate genes often overlap in function and expression. Overall, these results indicate that evolutionary age and expression level similarities over development are crucial for expressing gene essentiality in the mouse. All these findings express new insights into the correlations of gene essentiality, gene expression across embryonic development, and evolution.

6.4 Existence of the Hourglass Model in Mouse Development

In the gene essentiality prediction study, we observed that evolutionary origin and gene expression levels are two critical factors to indicate the potential roles of mouse genes during embryonic development. This promoted us to examine the existence of the hourglass model of morphological divergence for the mouse. The developmental hourglass model proposes that embryos from the same phylum

have a mid–gestation stage (phylotypic stage) where embryos are morphologically conserved. Recent studies on multiple organisms have demonstrated that genes expressed at the phylotypic stage tend to have an older evolutionary origin than genes expressed in early or late development, thus producing an hourglass pattern while plotting mean age of transcripts against developmental time–point (Domazet-Lošo and Tautz, 2010; Drost et al., 2015). We examined the evolutionary age of mouse genes expressed at 13 developmental stages to see if this pattern holds in the mouse. We considered the phylotypic period as corresponding to gastrula and organogenesis stages based on prior evidence.

We found that in mouse embryos, genes expressed in early and late stages of development are evolutionarily younger than those expressed in mid–embryogenesis, thus recapitulating an hourglass pattern. However, the oldest genes are not expressed at the predefined phylotypic stage, but instead at the egg cylinder stage. At this earlier stage we also observed a greater proportion of expressed genes that are essential for mouse embryonic development. It is reasonable that the stage with the oldest genes also includes a higher proportion of essential genes, as essential genes are likely to have an older evolutionary origin. This result raises the question of whether the oldest and evolutionarily most conserved essential genes, in general, encode morphological similarities amongst species of the same phylum.

6.5 Limitations

Despite the promising findings, this research still has some limitations. Our proposed computational method for predicting mouse essential genes produces a small number of false positive and false negative errors despite the very high prediction performance. One expected cause for these errors is that the computational method depends entirely on the feature dataset we compiled that may contain some noise. We assembled these features from the publicly available databases and prediction tools that could have erroneous feature information in them. The performance of the predictor also depends on the quality and completeness of the training dataset. A subset of all mouse genes was used to train the computational model that may not capture the unusual patterns of some of the remaining mouse genes. In these scenarios, our computational model tries to provide prediction based on the available information and may not be 100% accurate as expected. Moreover, the computational method provides a complimentary (rather than alternative) step in deciding the candidate genes for rigorous experimental validation.

Also, we used expressed sequence tag (EST) counts to measure the expression levels of mouse genes in terms of transcripts per million (TPM) across 13 stages of embryonic development. EST data is limited to low coverage of genes in a genome. ESTs only measure the highly expressed genes, are semi-quantitative, and different developmental time points have very different EST library coverage, which impacts the power to measure expression. The RNA–seq data, which quantifies large dynamic range of expression levels, might become a better estimate for expression

profiles if attainable for all 13 developmental stages. To the best of our knowledge, no such RNA–seq dataset of specific developmental stages is available for the mouse at the moment.

6.6 Future Work

Our computational method provides a complimentary step in deciding the candidate genes for the rigorous experimental validation. We tried to make use of all the attainable features related to different aspects of mouse biology for the development of our computational model. It is always possible to add new features (when becomes available) to our existing model that may contribute to improved classification accuracy. In addition to that, changing the feature selection technique might substantially improve the prediction performance. Studies have shown that wrapper methods are best suited for selecting relevant features amongst all the existing feature selection approaches (Maldonado and Weber, 2009; Kursa and Rudnicki, 2011), though these require greater computational efforts. These wrapper methods select useful features in relation to the chosen classifier. We expect to apply this technique in future to verify whether it offers further improvement.

However, only 4,766 mouse genes are annotated as essential or non–essential, leaving approximately 17,000 genes to annotate. Future work may include generating predictions for these mouse genes lacking experimental annotations and performing experimental validation on a small set of predicted genes to verify the performance of our classifier. We will further study these predicted genes to

learn and understand the cellular processes that may be absolutely fundamental for essentiality. We will investigate whether these biological processes are highly conserved across other organisms or are specific to mammals only. The scope our research is not limited to only mouse and can easily be extended to other organisms if the knockout phenotypic data become available for such analyses. Moreover, we will investigate the developmental co–expression of duplicate gene pairs for other vertebrate organisms. It will be interesting to see whether the conclusions made in mouse hold in general for other organisms if data become available. Also, in light of recent availability of RNA–seq data for human (Gerrard et al., 2016), we plan to investigate whether the developmental hourglass pattern exists in human embryos.

Bibliography

Abzhanov, A. (2013). von baer's law for the ages: lost and found principles of developmental evolution. *Trends in Genetics*, 29(12):712–722.

Acencio, M. L. and Lemke, N. (2009). Towards the prediction of essential genes by integration of network topology, cellular localization and biological process information. *BMC Bioinformatics*, 10(1):1–18.

Aha, D. W., Kibler, D., and Albert, M. K. (1991). Instance–based learning algorithms. *Machine Learning*, 6(1):37–66.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410.

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped blast and psi–blast: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402.

Apweiler, R., Bairoch, A., Wu, C. H., Barker, W. C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., and Magrane, M. (2004). Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, 32(suppl 1):D115–D119.

Apweiler, R., Hermjakob, H., and Sharon, N. (1999). On the frequency of protein glycosylation, as deduced from analysis of the swiss-prot database. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1473(1):4–8.

Arnesen, T. (2011). Towards a functional understanding of protein n–terminal acetylation. *PLoS Biology*, 9(5):e1001074.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., and Eppig, J. T. (2000). Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29.

Baer, K. v. (1828). Über entwicklungsgeschichte der thiere. *Beobachtung und Reflexion, Theil*, 1:734.

Bakheet, T. M. and Doig, A. J. (2009). Properties and identification of human protein drug targets. *Bioinformatics*, 25(4):451–457.

Bakheet, T. M. and Doig, A. J. (2010). Properties and identification of antibiotic drug targets. *BMC Bioinformatics*, 11(1):195–214.

Barabasi, A.-L. and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113.

Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., GriffithsJones, S., Khanna, A., Marshall, M., Moxon, S., and Sonnhammer, E. L. (2004). The pfam protein families database. *Nucleic Acids Research*, 32(suppl 1):D138–D141.

Bedzhov, I., Graham, S. J., Leung, C. Y., and Zernicka-Goetz, M. (2014a). Developmental plasticity, cell fate specification and morphogenesis in the early mouse embryo. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 369(1657):20130538.

Bedzhov, I., Leung, C. Y., Bialecka, M., and Zernicka-Goetz, M. (2014b). In vitro culture of mouse blastocysts beyond the implantation stages. *Nature Protocols*, 9(12):2732–2739.

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*, 4(10):e1000173.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The protein data bank. *Nucleic Acids Research*, 28(1):235–242.

Bogdanović, O., Smits, A. H., de la Calle Mustienes, E., Tena, J. J., Ford, E., Williams, R., Senanayake, U., Schultz, M. D., Hontelez, S., and van Kruijsbergen, I. (2016). Active dna demethylation at enhancers during the vertebrate phylotypic period. *Nature Genetics*.

Brandes, U. (2001). A faster algorithm for betweenness centrality*. *Journal of Mathematical Sociology*, 25(2):163–177.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Brocchieri, L. and Karlin, S. (2005). Protein length in eukaryotic and prokaryotic proteomes. *Nucleic Acids Research*, 33(10):3390–3400.

Brown, K. R. and Jurisica, I. (2007). Unequal evolutionary conservation of human protein interactions in interologous networks. *Genome Biology*, 8(5):R95.

Budagyan, B. and Loraine, A. (2004). Gene length and alternative transcription in fruit fly. In *Computational Systems Bioinformatics Conference, 2004. CSB 2004. Proceedings. 2004 IEEE*, pages 515–516. IEEE.

Bull, S. C. and Doig, A. J. (2015). Properties of protein drug target classes. *PloS One*, 10(3):e0117955.

Bull, S. C., Muldoon, M. R., and Doig, A. J. (2013). Maximising the size of non–redundant protein datasets using graph theory. *PloS One*, 8(2):e55484.

Bult, C. J., Eppig, J. T., Kadin, J. A., Richardson, J. E., and Blake, J. A. (2008). The mouse genome database (mgd): mouse biology and model systems. *Nucleic Acids Research*, 36(suppl 1):D724–D728.

Bureau, A., Dupuis, J., Falls, K., Lunetta, K. L., Hayward, B., Keith, T. P., and Van Eerdewegh, P. (2005). Identifying snps predictive of phenotype using random forests. *Genetic Epidemiology*, 28(2):171–182.

Chen, W.-H., Minguez, P., Lercher, M. J., and Bork, P. (2012a). Ogee: an online gene essentiality database. *Nucleic Acids Research*, 40(D1):D901–D906.

Chen, W.-H., Trachana, K., Lercher, M. J., and Bork, P. (2012b). Younger genes are less likely to be essential than older genes, and duplicates are less likely to be essential than singletons of the same age. *Molecular Biology and Evolution*, page mss014.

Chen, Y., Li, Y., Narayan, R., Subramanian, A., and Xie, X. (2016). Gene expression inference with deep learning. *Bioinformatics*, page btw074.

Chen, Y. and Xu, D. (2005). Understanding protein dispensability through machine–learning analysis of high-throughput data. *Bioinformatics*, 21(5):575–581.

Cheng, X., Hui, J. H. L., Lee, Y. Y., Law, P. T. W., and Kwan, H. S. (2015). A developmental hourglass in fungi. *Molecular Biology and Evolution*, page msv047.

Chin, C.-S. and Samanta, M. P. (2003). Global snapshot of a protein interaction network-a percolation based approach. *Bioinformatics*, 19(18):2413–2419.

Comeron, J. M. (2001). What controls the length of noncoding dna? *Current opinion in genetics & development*, 11(6):652–659.

Conant, G. C. and Wagner, A. (2004). Duplicate genes and robustness to transient gene knock-downs in caenorhabditis elegans. *Proceedings of the Royal Society of London B: Biological Sciences*, 271(1534):89–96.

Cooke, J., Nowak, M. A., Boerlijst, M., and Maynard-Smith, J. (1997). Evolutionary origins and maintenance of redundant gene expression during metazoan development. *Trends in Genetics*, 13(9):360–364.

Cord, M. and Cunningham, P. (2008). *Machine learning techniques for multimedia*. Springer, Berlin, Heidelberg.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Coulomb, S., Bauer, M., Bernard, D., and Marsolier-Kergoat, M.-C. (2005). Gene essentiality and the topology of protein interaction networks. *Proceedings of the Royal Society of London B: Biological Sciences*, 272(1573):1721–1725.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27.

Crawley, J. N. (1999). Behavioral phenotyping of transgenic and knockout mice: experimental design and evaluation of general health, sensory functions, motor abilities, and specific behavioral tests. *Brain Research*, 835(1):18–26.

Cullen, L. M. and Arndt, G. M. (2005). Genome-wide screening for gene function using rnai in mammalian cells. *Immunology and Cell Biology*, 83(3):217–223.

Davis, J. C., Brandman, O., and Petrov, D. A. (2005). Protein evolution in the context of drosophila development. *Journal of Molecular Evolution*, 60(6):774–785.

Dean, E. J., Davis, J. C., Davis, R. W., and Petrov, D. A. (2008). Pervasive and persistent redundancy among duplicated genes in yeast. *PLoS Genetics*, 4(7):e1000113.

DeLuna, A., Vetsigian, K., Shoresh, N., Hegreness, M., Colón-González, M., Chao, S., and Kishony, R. (2008). Exposing the fitness contribution of duplicated genes. *Nature Genetics*, 40(5):676–681.

Deng, J., Deng, L., Su, S., Zhang, M., Lin, X., Wei, L., Minai, A. A., Hassett, D. J., and Lu, L. J. (2010). Investigating the predictability of essential genes across distantly related organisms using an integrative approach. *Nucleic Acids Research*, 39(3):1–22.

Dickerson, J. E., Zhu, A., Robertson, D. L., and Hentges, K. E. (2011). Defining the role of essential genes in human disease. *PloS One*, 6(11):e27368.

Ditterrich, T. (1997). Machine learning research: four current direction. *Artificial Intelligence Magzine*, 4:97–136.

Domazet-Lošo, T. and Tautz, D. (2010). A phylogenetically based transcriptome age index mirrors ontogenetic divergence patterns. *Nature*, 468(7325):815–818.

Dougherty, J., Kohavi, R., and Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Machine learning: proceedings of the twelfth international conference*, volume 12, pages 194–202.

Drost, H.-G., Gabel, A., Grosse, I., and Quint, M. (2015). Evidence for active maintenance of phylotranscriptomic hourglass patterns in animal and plant embryogenesis. *Molecular Biology and Evolution*, 32(5):1221–1231.

Drummond, D. A., Bloom, J. D., Adami, C., Wilke, C. O., and Arnold, F. H. (2005). Why highly expressed proteins evolve slowly. *Proceedings of the National Academy of Sciences of the United States of America*, 102(40):14338–14343.

Duboule, D. (1994). Temporal colinearity and the phylotypic progression: a basis for the stability of a vertebrate bauplan and the evolution of morphologies through heterochrony. *Development*, 1994(Supplement):135–142.

Dunn, O. J. (1961). Multiple comparisons among means. *Journal of the American Statistical Association*, 56(293):52–64.

Duret, L., Mouchiroud, D., and Gautier, C. (1995). Statistical analysis of vertebrate sequences reveals that long genes are scarce in gc-rich isochores. *Journal of Molecular Evolution*, 40(3):308–317.

Efron, B. (1981). Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods. *Biometrika*, 68(3):589–599.

Elinson, R. P. (1987). *Change in developmental patterns: embryos of amphibians with large eggs*, volume 8, pages 1–21. Alan R. Liss., p. 121.

Fares, M. A., Keane, O. M., Toft, C., Carretero-Paulet, L., and Jones, G. W. (2013). The roles of whole–genome and small-scale duplications in the functional specialization of saccharomyces cerevisiae genes. *PLoS Genetics*, 9(1):e1003176.

Freund, Y. and Mason, L. (1999). The alternating decision tree learning algorithm. In *Icml*, volume 99, pages 124–133.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Icml*, volume 96, pages 148–156.

Gallagher, L. A., Ramage, E., Jacobs, M. A., Kaul, R., Brittnacher, M., and Manoil, C. (2007). A comprehensive transposon mutant library of francisella novicida, a bioweapon surrogate. *Proceedings of the National Academy of Sciences*, 104(3):1009–1014.

Gardner, R. (2001). The initial phase of embryonic patterning in mammals. *International Review of Cytology*, 203:233–290.

Gazave, E., Marqus-Bonet, T., Fernando, O., Charlesworth, B., and Navarro, A. (2007). Patterns and rates of intron divergence between humans and chimpanzees. *Genome Biology*, 8(2):R21.

Gerrard, D. T., Berry, A. A., Jennings, R. E., Hanley, K. P., Bobola, N., and Hanley, N. A. (2016). An integrative transcriptomic atlas of organogenesis in human embryos. *eLife*, 5:e15657.

Giaever, G., Chu, A. M., Ni, L., Connelly, C., Riles, L., Veronneau, S., Dow, S., Lucau-Danila, A., Anderson, K., and Andre, B. (2002). Functional profiling of the saccharomyces cerevisiae genome. *Nature*, 418(6896):387–391.

Gilbert, S. F. (1994). *Developmental Biology (4th Edition)*. Sinauer Associates, Inc., Sunderland, Massachusetts, fourth edition.

Goh, K.-I., Cusick, M. E., Valle, D., Childs, B., Vidal, M., and Barabsi, A.-L. (2007). The human disease network. *Proceedings of the National Academy of Sciences*, 104(21):8685–8690.

Gu, X. (2003). Evolution of duplicate genes versus genetic robustness against null mutations. *Trends in Genetics*, 19(7):354–356.

Gu, Z., Nicolae, D., Lu, H. H., and Li, W.-H. (2002). Rapid divergence in expression between duplicate genes inferred from microarray data. *Trends in Genetics*, 18(12):609–613.

Gu, Z., Steinmetz, L. M., Gu, X., Scharfe, C., Davis, R. W., and Li, W.-H. (2003). Role of duplicate genes in genetic robustness against null mutations. *Nature*, 421(6918):63–66.

Guan, Y., Dunham, M. J., and Troyanskaya, O. G. (2007). Functional analysis of gene duplications in saccharomyces cerevisiae. *Genetics*, 175(2):933–943.

Guénet, J. L. (2005). The mouse genome. *Genome Research*, 15(12):1729–1740.

Gustafson, A. M., Snitkin, E. S., Parker, S. C., DeLisi, C., and Kasif, S. (2006). Towards the identification of essential genes using targeted genome sequencing and comparative analysis. *BMC Genomics*, 7(1):265.

Hagan, M. T., Demuth, H. B., Beale, M. H., and De Jess, O. (1996). *Neural network design*, volume 20. PWS publishing company Boston.

Hahn, M. W. and Kern, A. D. (2005). Comparative genomics of centrality and essentiality in three eukaryotic protein–interaction networks. *Molecular Biology and Evolution*, 22(4):803–806.

Hakes, L., Pinney, J. W., Lovell, S. C., Oliver, S. G., and Robertson, D. L. (2007). All duplicates are not equal: the difference between small–scale and genome duplication. *Genome Biology*, 8(10):R209.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Haselbeck, R., Wall, D., Jiang, B., Ketela, T., Zyskind, J., Bussey, H., Foulkes, J., and Roemer, T. (2002). Comprehensive essential gene identification as a platform for novel antiinfective drug discovery. *Current Pharmaceutical Design*, 8(13):1155–1172.

Hazkani-Covo, E., Wool, D., and Graur, D. (2005). In search of the vertebrate phylotypic stage: a molecular examination of the developmental hourglass model and von baer's third law. *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, 304(2):150–158.

He, X. and Zhang, J. (2006). Why do hubs tend to be essential in protein networks? *PLoS Genetics*, 2(6):e88.

Henriksen, P., Wagner, S. A., Weinert, B. T., Sharma, S., Bainskaja, G., Rehman, M., Juffer, A. H., Walther, T. C., Lisby, M., and Choudhary, C. (2012). Proteome–wide analysis of lysine acetylation suggests its broad regulatory scope in saccharomyces cerevisiae. *Molecular & Cellular Proteomics*, 11(11):1510–1522.

Hentges, K. E., Pollock, D. D., Liu, B., and Justice, M. J. (2007). Regional variation in the density of essential genes in mice. *PLoS Genetics*, 3(5):e72.

Horton, P., Park, K.-J., Obayashi, T., Fujita, N., Harada, H., Adams-Collier, C., and Nakai, K. (2007). Wolf psort: protein localization predictor. *Nucleic Acids Research*, 35(suppl 2):W585–W587.

Hsiao, T.-L. and Vitkup, D. (2008). Role of duplicate genes in robustness against deleterious human mutations. *PLoS Genetics*, 4(3):e1000014.

Huang, D. W., Sherman, B. T., Tan, Q., Kir, J., Liu, D., Bryant, D., Guo, Y., Stephens, R., Baseler, M. W., and Lane, H. C. (2007). David bioinformatics resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic Acids Research*, 35(suppl 2):W169–W175.

Huang, H., Winter, E. E., Wang, H., Weinstock, K. G., Xing, H., Goodstadt, L., Stenson, P. D., Cooper, D. N., Smith, D., and Alb, M. M. (2004). Evolutionary conservation and selection of human disease gene orthologs in the rat and mouse genomes. *Genome Biology*, 5(7):1.

Huang, J. and Ling, C. X. (2005). Using auc and accuracy in evaluating learning algorithms. *IEEE Transactions on knowledge and Data Engineering*, 17(3):299–310.

Hubbard, T., Barker, D., Birney, E., Cameron, G., Chen, Y., Clark, L., Cox, T., Cuff, J., Curwen, V., Down, T., Durbin, R., Eyras, E., Gilbert, J., Hammond, M., Huminiecki, L., Kasprzyk, A., Lehvaslaiho, H., Lijnzaad, P., Melsopp, C., Mongin, E., Pettett, R., Pocock, M., Potter, S., Rust, A., Schmidt, E., Searle, S., Slater, G., Smith, J., Spooner, W., Stabenau, A., Stalker, J., Stupka, E., Ureta-Vidal, A., Vastrik, I., and Clamp, M. (2002). The ensembl genome database project. *Nucleic Acids Research*, 30(1):38–41.

Hughes, A. (2003). Comparative genomics: Genomes of mice and men. *Heredity*, 90(2):115–116.

Hwang, Y.-C., Lin, C.-C., Chang, J.-Y., Mori, H., Juan, H.-F., and Huang, H.-C. (2009). Predicting essential genes based on network and sequence analysis. *Molecular BioSystems*, 5(12):1672–1678.

Information, S. C. (2001). Appendix a: Early development.

Irie, N. and Kuratani, S. (2011). Comparative transcriptome analysis reveals vertebrate phylotypic period during organogenesis. *Nature Communications*, 2:248.

Irie, N. and Kuratani, S. (2014). The developmental hourglass model: a predictor of the basic body plan? *Development*, 141(24):4649–4655.

Irie, N. and Sehara-Fujisawa, A. (2007). The vertebrate phylotypic stage and an early bilaterian-related stage in mouse embryogenesis defined by genomic information. *BMC Biology*, 5(1):1.

Jensen, L. J., Gupta, R., Staerfeldt, H.-H., and Brunak, S. (2003). Prediction of human protein function according to gene ontology categories. *Bioinformatics*, 19(5):635–642.

Jeong, H., Mason, S. P., Barabsi, A.-L., and Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature*, 411(6833):41–42.

Jeong, H., Oltvai, Z. N., and Barabsi, A.-L. (2003). Prediction of protein essentiality based on genomic data. *ComPlexUs*, 1(1):19–28.

Jo, T., Hou, J., Eickholt, J., and Cheng, J. (2015). Improving protein fold recognition by deep learning networks. *Scientific reports*, 5.

Jordan, I. K., Rogozin, I. B., Wolf, Y. I., and Koonin, E. V. (2002). Essential genes are more evolutionarily conserved than are nonessential genes in bacteria. *Genome Research*, 12(6):962–968.

Joy, M. P., Brock, A., Ingber, D. E., and Huang, S. (2005). High-betweenness proteins in the yeast protein interaction network. *BioMed Research International*, 2005(2):96–103.

Joyner, A. L., Herrup, K., Auerbach, B. A., Davis, C. A., and Rossant, J. (1991). Subtle cerebellar phenotype in mice homozygous for a targeted deletion of the en-2 homeobox. *Science*, 251(4998):1239–1243.

Juhas, M., Eberl, L., and Glass, J. I. (2011). Essence of life: essential genes of minimal genomes. *Trends in Cell Biology*, 21(10):562–568.

Kalinka, A. T. and Tomancak, P. (2012). The evolution of early animal embryos: conservation or divergence? *Trends in Ecology & Evolution*, 27(7):385–393.

Kalinka, A. T., Varga, K. M., Gerrard, D. T., Preibisch, S., Corcoran, D. L., Jarrells, J., Ohler, U., Bergman, C. M., and Tomancak, P. (2010). Gene expression divergence recapitulates the developmental hourglass model. *Nature*, 468(7325):811–814.

Kamath, R. S., Fraser, A. G., Dong, Y., Poulin, G., Durbin, R., Gotta, M., Kanapin, A., Le Bot, N., Moreno, S., and Sohrmann, M. (2003). Systematic functional analysis of the caenorhabditis elegans genome using rnai. *Nature*, 421(6920):231–237.

Kim, P. M., Lu, L. J., Xia, Y., and Gerstein, M. B. (2006). Relating three–dimensional structures to protein networks provides evolutionary insights. *Science*, 314(5807):1938–1941.

Kira, K. and Rendell, L. A. (1992). A practical approach to feature selection. In *Proceedings of the ninth international workshop on Machine learning*, pages 249–256.

Kirkpatrick, S. (1984). Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, 34(5-6):975–986.

Knight, R. D., Freeland, S. J., and Landweber, L. F. (2001). A simple model based on mutation and selection explains trends in codon and amino–acid usage and gc composition within and across genomes. *Genome Biology*, 2(4):1–13.

Kobayashi, K., Ehrlich, S. D., Albertini, A., Amati, G., Andersen, K., Arnaud, M., Asai, K., Ashikaga, S., Aymerich, S., and Bessieres, P. (2003). Essential *Bacillus subtilis* genes. *Proceedings of the National Academy of Sciences*, 100(8):4678–4683.

Kohavi, R. (1995). A study of cross–validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324.

Kondrashov, F. A., Ogurtsov, A. Y., and Kondrashov, A. S. (2004). Bioinformatical assay of human gene morbidity. *Nucleic Acids Research*, 32(5):1731–1737.

Kotsiantis, S. B., Zaharakis, I., and Pintelas, P. (2007). Supervised machine learning: A review of classification techniques.

Kruskal, W. H. and Wallis, W. A. (1952). Use of ranks in one–criterion variance analysis. *Journal of the American Statistical Association*, 47(260):583–621.

Kumar, A., Agarwal, S., Heyman, J. A., Matson, S., Heidtman, M., Piccirillo, S., Umansky, L., Drawid, A., Jansen, R., and Liu, Y. (2002). Subcellular localization of the yeast proteome. *Genes & Development*, 16(6):707–719.

Kursa, M. B. and Rudnicki, W. R. (2011). The all relevant feature selection using random forest. *arXiv preprint arXiv:1106.5112*.

Landwehr, N., Hall, M., and Frank, E. (2005). Logistic model trees. *Machine Learning*, 59(1-2):161–205.

Le Cessie, S. and Van Houwelingen, J. C. (1992). Ridge estimators in logistic regression. *Applied Statistics*, pages 191–201.

Lee, C. and Lee, G. G. (2006). Information gain and divergence–based feature selection for machine learning-based text categorization. *Information Processing & Management*, 42(1):155–165.

Li, H., Coghlan, A., Ruan, J., Coin, L. J., Heriche, J.-K., Osmotherly, L., Li, R., Liu, T., Zhang, Z., and Bolund, L. (2006). Treefam: a curated database of phylogenetic trees of animal gene families. *Nucleic Acids Research*, 34(suppl 1):D572–D580.

Liang, H. and Li, W.-H. (2007). Gene essentiality, gene duplicability and protein connectivity in human and mouse. *Trends in Genetics*, 23(8):375–378.

Liao, B.-Y. and Zhang, J. (2006). Low rates of expression profile divergence in highly expressed genes and tissue-specific genes during mammalian evolution. *Molecular Biology and Evolution*, 23(6):1119–1128.

Liao, B.-Y. and Zhang, J. (2007). Mouse duplicate genes are as essential as singletons. *Trends in Genetics*, 23(8):378–381.

Lin, C.-Y., Chin, C.-H., Wu, H.-H., Chen, S.-H., Ho, C.-W., and Ko, M.-T. (2008). Hubba: hub objects analyser-a framework of interactome hubs identification for network biology. *Nucleic Acids Research*, 36(suppl 2):W438–W443.

Lipman, D. J., Souvorov, A., Koonin, E. V., Panchenko, A. R., and Tatusova, T. A. (2002). The relationship of protein conservation and sequence length. *BMC Evolutionary Biology*, 2(1):20.

Liu, H., Hussain, F., Tan, C. L., and Dash, M. (2002). Discretisation: An enabling technique. *Data Mining and Knowledge Discovery*, 6(4):393–423.

Liu, J., Yakar, S., and LeRoith, D. (2000). Conditional knockout of mouse insulin-like growth factor1 gene using the cre/loxp system. *Proceedings of the Society for Experimental Biology and Medicine*, 223(4):344–351.

Long, M., Betrn, E., Thornton, K., and Wang, W. (2003). The origin of new genes: glimpses from the young and old. *Nature Reviews Genetics*, 4(11):865–875.

López-Bigas, N. and Ouzounis, C. A. (2004). Genome–wide identification of genes likely to be involved in human genetic disease. *Nucleic Acids Research*, 32(10):3108–3114.

Lynch, M. and Conery, J. S. (2003). The origins of genome complexity. *Science*, 302(5649):1401–1404.

Makino, T., Hokamp, K., and McLysaght, A. (2009). The complex relationship of gene duplication and essentiality. *Trends in Genetics*, 25(4):152–155.

Makino, T. and McLysaght, A. (2010). Ohnologs in the human genome are dosage balanced and frequently associated with disease. *Proceedings of the National Academy of Sciences*, 107(20):9270–9274.

Makova, K. D. and Li, W.-H. (2003). Divergence in the spatial pattern of gene expression between human duplicate genes. *Genome Research*, 13(7):1638–1645.

Maldonado, S. and Weber, R. (2009). A wrapper method for feature selection using support vector machines. *Information Sciences*, 179(13):2208–2217.

Mann, H. B. and Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, pages 50–60.

Mann, M., Ong, S.-E., Grnborg, M., Steen, H., Jensen, O. N., and Pandey, A. (2002). Analysis of protein phosphorylation using mass spectrometry: deciphering the phosphoproteome. *Trends in Biotechnology*, 20(6):261–268.

Massey Jr, F. J. (1951). The kolmogorov–smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78.

Montoya-Burgos, J. I., Boursot, P., and Galtier, N. (2003). Recombination explains isochores in mammalian genomes. *Trends in Genetics*, 19(3):128–130.

Motenko, H., Neuhauser, S. B., OKeefe, M., and Richardson, J. E. (2015). Mousemine: a new data warehouse for mgi. *Mammalian Genome*, 26(7-8):325–330.

Newman, M. E. (2005). A measure of betweenness centrality based on random walks. *Social Networks*, 27(1):39–54.

Ninova, M., Ronshaugen, M., and Griffiths-Jones, S. (2014). Conserved temporal patterns of microrna expression in drosophila support a developmental hourglass model. *Genome Biology and Evolution*, 6(9):2459–2467.

Norusis, M. J. (1985). *SPSS–X advanced statistics guide*.

Ohno, S. (1970). Evolution by gene duplication. *Springer-Verlag*.

Oliver, P. L., Bitoun, E., and Davies, K. E. (2007). Comparative genetic analysis: the utility of mouse genetic systems for studying human monogenic disease. *Mammalian Genome*, 18(6-7):412–424.

Oppenheimer, S. and Lefevre, G. (1984). *Introduction to embryonic development*. Allyn and Bacon, Inc.

Pál, C., Papp, B., and Hurst, L. D. (2003). Genomic function (communication arising): Rate of evolution and gene dispensability. *Nature*, 421(6922):496–497.

Palaniappan, K. and Mukherjee, S. (2011). Predicting" essential" genes across microbial genomes: A machine learning approach. In *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, volume 2, pages 189–194. IEEE.

Petersen, T. N., Brunak, S., von Heijne, G., and Nielsen, H. (2011). Signalp 4.0: discriminating signal peptides from transmembrane regions. *Nature Methods*, 8(10):785–786.

Piasecka, B., Lichocki, P., Moretti, S., Bergmann, S., and Robinson-Rechavi, M. (2013). The hourglass and the early conservation modelsco-existing patterns of developmental constraints in vertebrates. *PLoS Genetics*, 9(4):e1003476.

Prachumwat, A. and Li, W.-H. (2006). Protein function, connectivity, and duplicability in yeast. *Molecular Biology and Evolution*, 23(1):30–39.

Prud'Homme, B. and Gompel, N. (2010). Evolutionary biology: Genomic hourglass. *Nature*, 468(7325):768–769.

Pržulj, N., Wigle, D. A., and Jurisica, I. (2004). Functional topology in a network of protein interactions. *Bioinformatics*, 20(3):340–348.

Puente, L. G., Borris, D. J., Carrire, J.-F., Kelly, J. F., and Megeney, L. A. (2006). Identification of candidate regulators of embryonic stem cell differentiation by comparative phosphoprotein affinity profiling. *Molecular & Cellular Proteomics*, 5(1):57–67.

Queisser-Luft, A., Stolz, G., Wiesel, A., Schlaefer, K., and Spranger, J. (2002). Malformations in newborn: results based on 30940 infants and fetuses from the mainz congenital birth defect monitoring system (1990–1998). *Archives of Gynecology and Obstetrics*, 266(3):163–167.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. Morgan Kaufmann.

Quint, M., Drost, H.-G., Gabel, A., Ullrich, K. K., Bnn, M., and Grosse, I. (2012). A transcriptomic hourglass in plant embryogenesis. *Nature*, 490(7418):98–101.

Raff, R. A. (1996). *The shape of evolutionary developmental biology*. Chicago: University Chicago Press.

Rangarajan, A. and Weinberg, R. A. (2003). Comparative biology of mouse versus human cells: modelling human cancer in mice. *Nature Reviews Cancer*, 3(12):952–959.

Rice, P., Longden, I., and Bleasby, A. (2000). Emboss: the european molecular biology open software suite. *Trends in Genetics*, 16(6):276–277.

Richardson, M. K. (1995). Heterochrony and the phylotypic period. *Developmental Biology*, 172(2):412–421.

Richardson, M. K. and Keuck, G. (2002). Haeckel's abc of evolution and development. *Biological Reviews of the Cambridge Philosophical Society*, 77(04):495–528.

Rish, I. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46. IBM New York.

Roemer, T., Jiang, B., Davison, J., Ketela, T., Veillette, K., Breton, A., Tandia, F., Linteau, A., Sillaots, S., and Marta, C. (2003). Large–scale essential gene identification in candida albicans and applications to antifungal drug discovery. *Molecular Microbiology*, 50(1):167–181.

Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Engineering*, 12(2):85–94.

Russell, S. J., Norvig, P., Canny, J. F., Malik, J. M., and Edwards, D. D. (2003). *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River.

Saga, Y., Yagi, T., Ikawa, Y., Sakakura, T., and Aizawa, S. (1992). Mice develop normally without tenascin. *Genes & Development*, 6(10):1821–1831.

Saha, S. and Heber, S. (2006). In silico prediction of yeast deletion phenotypes. *Genetics & Molecular Research*, 5(1):224–232.

Sander, K. et al. (1983). The evolution of patterning mechanisms: gleanings from insect embryogenesis and spermatogenesis. *Development and Evolution*, pages 137–159.

Seringhaus, M., Paccanaro, A., Borneman, A., Snyder, M., and Gerstein, M. (2006). Predicting essential genes in fungal genomes. *Genome Research*, 16(9):1126–1135.

Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55.

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):2498–2504.

Shi, H. (2007). *Best-first decision tree learning*. PhD thesis.

Shindyalov, I. N. and Bourne, P. E. (1998). Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein Engineering*, 11(9):739–747.

Simmer, F., Moorman, C., van der Linden, A. M., Kuijk, E., van den Berghe, P. V., Kamath, R. S., Fraser, A. G., Ahringer, J., and Plasterk, R. H. (2003). Genome-wide rnai of c. elegans using the hypersensitive rrf-3 strain reveals novel gene functions. *PLoS Biology*, 1(1):e12.

Skreb, N., Solter, D., and Damjanov, I. (1991). Developmental biology of the murine egg cylinder. *The International Journal of Developmental Biology*, 35(3):161.

Smedley, D., Haider, S., Ballester, B., Holland, R., London, D., Thorisson, G., and Kasprzyk, A. (2009). Biomart–biological queries made easy. *BMC Genomics*, 10(1):1.

Spencer, M., Eickholt, J., and Cheng, J. (2015). A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 12(1):103–112.

Stanton, J.-A. L., Macgregor, A. B., and Green, D. P. (2003). Identifying tissue-enriched gene expression in mouse tissues using the nih unigene database. *Applied Bioinformatics*, 2:S65–S74.

Steinmetz, L. M., Scharfe, C., Deutschbauer, A. M., Mokranjac, D., Herman, Z. S., Jones, T., Chu, A. M., Giaever, G., Prokisch, H., and Oefner, P. J. (2002). Systematic screen for human disease genes in yeast. *Nature Genetics*, 31(4):400–404.

Su, Z. and Gu, X. (2008). Predicting the proportion of essential genes in mouse duplicates based on biased mouse knockout genes. *Journal of Molecular Evolution*, 67(6):705–709.

Su, Z., Wang, J., and Gu, X. (2014). Effect of duplicate genes on mouse genetic robustness: An update. *BioMed Research International*, 2014.

Thomas, J. H. (1993). Thinking about genetic redundancy. *Trends in Genetics*, 9(11):395–399.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Vilella, A. J., Severin, J., Ureta-Vidal, A., Heng, L., Durbin, R., and Birney, E. (2009). Ensemblcompara genetrees: Complete, duplication-aware phylogenetic trees in vertebrates. *Genome Research*, 19(2):327–335.

Vinogradov, A. E. (2004). Compactness of human housekeeping genes: selection for economy or genomic design? *Trends in Genetics*, 20(5):248–253.

Visa, S. and Ralescu, A. (2005). Issues in mining imbalanced data sets-a review paper. In *Proceedings of the sixteen midwest artificial intelligence and cognitive science conference*, pages 67–73. sn.

Vitter, J. S. (1985). Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, 11(1):37–57.

Wang, G. and Dunbrack, R. L. (2005). Pisces: recent improvements to a pdb sequence culling server. *Nucleic Acids Research*, 33(suppl 2):W94–W98.

Weng, T.-Y., Chiu, W.-T., Liu, H.-S., Cheng, H.-C., Shen, M.-R., Mount, D. B., and Chou, C.-Y. (2013). Glycosylation regulates the function and membrane localization of kcc4. *Biochimica et Biophysica Acta (BBA)-Molecular Cell Research*, 1833(5):1133–1146.

White, J. K., Gerdin, A.-K., Karp, N. A., Ryder, E., Buljan, M., Bussell, J. N., Salisbury, J., Clare, S., Ingham, N. J., and Podrini, C. (2013). Genome–wide generation and systematic phenotyping of knockout mice reveals new roles for many genes. *Cell*, 154(2):452–464.

Wilcox, A. J., Baird, D. D., and Weinberg, C. R. (1999). Time of implantation of the conceptus and loss of pregnancy. *New England Journal of Medicine*, 340(23):1796–1799.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wolfer, D. P., Crusio, W. E., and Lipp, H.-P. (2002). Knockout mice: simple solutions to the problems of genetic background and flanking genes. *Trends in Neurosciences*, 25(7):336–340.

Wolpert, L., Tickle, C., and Arias, A. M. (2015). *Principles of development*. Oxford University Press, USA.

Wood, T. C. and Pearson, W. R. (1999). Evolution of protein sequences and structures. *Journal of Molecular Biology*, 291(4):977–995.

Yan, K., Khoshnoodi, J., Ruotsalainen, V., and Tryggvason, K. (2002). N–linked glycosylation is critical for the plasma membrane localization of nephrin. *Journal of the American Society of Nephrology*, 13(5):1385–1389.

Yang, L., Wang, J., Wang, H., Lv, Y., Zuo, Y., Li, X., and Jiang, W. (2014). Analysis and identification of essential genes in humans using topological properties and biological information. *Gene*, 551(2):138–151.

Yu, H., Greenbaum, D., Lu, H. X., Zhu, X., and Gerstein, M. (2004). Genomic analysis of essentiality within protein networks. *Trends in Genetics*, 20(6):227–231.

Yuan, Y., Xu, Y., Xu, J., Ball, R. L., and Liang, H. (2012). Predicting the lethal phenotype of the knockout mouse by integrating comprehensive genomic data. *Bioinformatics*, 28(9):1246–1252.

Zaret, K. S. (2001). Hepatocyte differentiation: from the endoderm and beyond. *Current Opinion in Genetics and Development*, 11(5):568–574.

Zhang, C.-T. and Zhang, R. (2008). Gene essentiality analysis based on deg, a database of essential genes. *Microbial Gene Essentiality: Protocols and Bioinformatics*, pages 391–400.

Zhang, J. (2003). Evolution by gene duplication: an update. *Trends in Ecology & Evolution*, 18(6):292–298.

Zhang, R. and Lin, Y. (2009). Deg 5.0, a database of essential genes in both prokaryotes and eukaryotes. *Nucleic Acids Research*, 37:D455–D458.

Zhang, X., Acencio, M. L., and Lemke, N. (2016). Predicting essential genes and proteins based on machine learning and network topological features: a comprehensive review. *Frontiers in Physiology*, 7.

Zhang, X., Xu, J., and Xiao, W.-x. (2013). A new method for the discovery of essential proteins. *PloS One*, 8(3):e58763.

Zhong, J., Wang, J., Peng, W., Zhang, Z., and Pan, Y. (2013). Prediction of essential proteins based on gene expression programming. *BMC Genomics*, 14(Suppl 4):S7.

Zhu, L., Zhang, Y., Zhang, W., Yang, S., Chen, J.-Q., and Tian, D. (2009). Patterns of exon-intron architecture variation of genes in eukaryotic genomes. *BMC Genomics*, 10(1):1.

AppendixA

Table A.1: Differences in gene and protein sequence-based features of lethal and viable genes in the **test–new** dataset.

Properties	**Lethal(median)**	**Viable(median)**	**p–value**
Gene Length	22458	25039	0.297
GC content (%)	45.69	46.30	0.15
Transcript count	4	3	5.0×10^{-03}
Exon count	11	9	4.7×10^{-05}
Exon length	2909	2867	0.204
Intron length	18849	21863	0.265
Age	937	400	1.9×10^{-24}
MW	55591.59	48584.91	0.075
Protein length	508	434	0.078
Aliphatic	28.61	28.04	0.267
Aromatic	10.12	10.71	0.013
NonPolar	52.52	52.97	0.115
Polar	47.48	47.03	0.117
Charged	26.81	25.55	0.001
Basic	14.11	13.70	0.020
Acidic	12.17	11.48	0.001
A	7.14	6.74	0.039
C	1.85	1.97	0.003
D	5.05	4.71	0.006
E	7.08	6.49	0.001
F	3.56	3.75	0.067
G	5.83	6.23	0.056
H	2.41	2.48	0.032
I	4.38	4.26	0.603
K	5.74	5.26	4.1×10^{-04}
L	10.06	10.17	0.296
M	2.27	2.06	0.001
N	3.33	3.33	0.976
P	5.12	5.48	0.049
Q	4.37	4.37	0.336
R	5.60	5.59	0.756
S	7.33	7.99	0.001
T	4.90	5.06	0.174
V	6.29	6.11	0.324
W	0.97	1.25	9.7×10^{-05}
Y	2.80	2.68	0.484

Table A.2: Differences in proportions of lethal and viable mouse genes in the **test–new** dataset expressed across 13 embryonic developmental stages.

Developmental stages	**Lethal(%)**	**Viable(%)**	**p–value**
Oocyte	48.03	25.90	1.0×10^{-07}
U_Ovum	29.69	12.83	3.1×10^{-08}
Zygote	42.36	20.05	2.5×10^{-09}
Cleavage	55.90	28.27	3.2×10^{-10}
Morula	58.95	24.16	1.9×10^{-16}
Blastocyst	70.31	34.99	5.9×10^{-13}
Egg_Cylinder	26.20	9.09	1.9×10^{-10}
Gastrula	66.81	31.76	9.8×10^{-14}
Organogenesis	91.70	60.40	3.5×10^{-07}
Fetus	97.38	85.68	0.096
Neonate	89.52	67.87	1.0×10^{-07}
Juveline	96.94	83.56	0.055
Adult	97.82	94.40	0.640

Table A.3: Distributions of network properties between lethal and viable genes in the **test–new** dataset.

Network properties	**Lethal(median)**	**Viable(median)**	**p–value**
ASP (Known)	4.13	4.67	0.003
BC (Known)	0.01	0	0.022
Closeness centrality (Known)	0.22	0.17	0.005
Clustering Coefficient (Known)	0	0	0.005
Degree (Known)	2	1	0.002
BN (Known)	1	0	0.030
EPC (Known)	1.86	1.42	6.7×10^{-04}
MNC (Known)	1	1	0.118
DMNC (Known)	0	0	0.005
ASP (Known-Predicted)	2.98	3.23	4.9×10^{-49}
BC (Known-Predicted)	0.00	0.00	2.6×10^{-14}
Closeness centrality (Known-Predicted)	0.34	0.31	4.9×10^{-50}
Clustering coefficient (Known-Predicted)	0	0	2.6×10^{-15}
Degree (Known-Predicted)	20	5	1.0×10^{-17}
BN (Known-Predicted)	6	2	1.4×10^{-12}
EPC (Known-Predicted)	23.40	43.66	3.6×10^{-03}
MNC (Known-Predicted)	1	1	8.7×10^{-05}
DMNC (Known-Predicted)	0	0	5.9×10^{-12}

Table A.4: Differences in the frequencies of different keywords and enzyme classes observed between lethal and viable mouse proteins in the **test–new** dataset.

Properties	**Lethal(median)**	**Viable(median)**	**p–value**
Glycoprotein	6.11	16.81	1.7×10^{-04}
Phosphoprotein	39.74	21.17	8.3×10^{-07}
Acetylation	30.57	9.22	2.3×10^{-14}
Transcription	10.48	6.48	0.049
Signal peptide	3.49	14.94	1.4×10^{-05}
Transmembrane domain	13.54	21.67	0.014
Oxidoreductase	4.80	3.61	0.419
Transferase	6.99	5.98	0.588
Hydrolase	9.61	8.72	0.691
Lyase	0.44	0.50	0.906
Isomerase	0	0.37	0.355
Ligase	5.68	1.87	0.002

Table A.5: Subcellular locations of all lethal and viable mouse proteins in the **test–new** dataset.

Cellular components	**Lethal(median)**	**Viable(median)**	**p–value**
Nucleus (UniProt)	28.82	16.69	6.3×10^{-16}
Cytoplasm (UniProt)	25.76	20.80	8.2×10^{-12}
Plasma (UniProt)	4.80	10.34	1.3×10^{-18}
Membrane (UniProt)	6.55	11.71	2.6×10^{-13}
Extracellular (UniProt)	2.62	6.85	1.5×10^{-12}
Mitochondrion (UniProt)	12.66	4.36	0.015
ER (UniProt)	3.93	4.73	8.2×10^{-06}
Golgi (UniProt)	5.68	3.11	0.016
Lysosome (UniProt)	0.87	1.99	1.1×10^{-04}
Peroxisome (UniProt)	0.44	0.87	4.4×10^{-04}
CellJunction (UniProt)	1.75	2.86	0.021
CellProjection (UniProt)	3.06	2.62	0.720
Nuclues (WoLF PSORT)	71.62	59.78	0.045
Cytoplasm (WoLF PSORT)	69.43	58.53	0.062
Plasma (WoLF PSORT)	23.14	30.39	0.072
Extracellular (WoLF PSORT)	24.02	34.99	0.010
Golgi (WoLF PSORT)	2.62	2.74	0.923
ER (WoLF PSORT)	11.35	13.33	0.464
Mitochondria (WoLF PSORT)	38.43	33.87	0.303
Peroxisome (WoLF PSORT)	10.04	13.95	0.150
Lysosome (WoLF PSORT)	3.06	7.10	0.030

AppendixB

All the associated source codes and data will be made publicly available once the manuscripts describing the experimental results will get accepted for the publication.

Data Files Submitted with this Thesis in a CD

File Name	Description
Lethal_GeneList.xlsx	All lethal (essential) genes
Viable_GeneList.xlsx	All viable (non–essential) genes
Singleton–Duplicate_GeneLists.xlsx	All singleton and duplicate genes labelled as lethal or viable
SSD–WGD_GeneLists.xlsx	All small–scale and whole–genome duplicated genes labelled as lethal or viable
DCA_Age_AllGene.xlsx	Genes expressed at each stage and their DCA age
MRD_Age_AllGene.xlsx	Genes expressed at each stage and their MRD age
GenesExpressedAllStage.xlsx	Genes expressed at all stages
Top10%Genes_Removed_DCAAnalysis.xlsx	Top 10% of most highly expressed genes removed for analysis shown in Figure 5.17C
Top10%Genes_Removed_MRDAnalysis.xlsx	Top 10% of most highly expressed genes removed for analysis shown in Figure 5.17D
train-01Dataset.csv	Dataset used to train **RF–1** classifier
train-02Dataset.csv	Dataset used to train **RF–2** classifier
train-03Dataset.csv	Dataset used to train **RF–3** classifier
test-bDataset.csv	Dataset used to assess the efficacy of **RF–1** classifier
test-u01Dataset.csv	Dataset used to assess the efficacy of **RF–2** classifier
test-u02Dataset.csv	Dataset used to assess the efficacy of **RF–3** classifier
test-newDataset.csv	Newly annotated lethal and viable mouse genes published by the IMPC