DATA MINING METHODOLOGIES FOR GENE EXPRESSION ANALYSIS: APPLICATION TO STRAIN IMPROVEMENT

JONNALAGADDA SUDHAKAR

(B.Tech, National Institute of Technology, Warangal, India)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF CHEMICAL AND BIOMOLECULAR ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
2008

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor Prof. Rajagopalan Srinivasan for his excellent guidance and support throughout the course of my research. His wealth of knowledge and innovative thinking stimulated me in developing novel ideas in my research. I am indebted to him for his care and advice not only in my academic research but also in my daily life. Without him, my research would not be successful.

I sincerely thank Prof. I. A. Karimi, Dr. Lakshminarayanan S. and Prof. Low Boon Chuan (Department of Biological Sciences, NUS) for their helpful suggestions.

Special thanks to our collaborators at Bioprocessing Technology Institute (BTI), Dr. Steve Oh and Dr. Ow Siak Wei Dave for their help in providing gene expression data.

I would like to thank all my lab mates Ng Yew Seng, Mohammad Iftekhar Hossain, Arief Adhitya, Manish Mishra, Nguyen Trong Nhan and Mukta Bansal for maintaining pleasant working environment. The discussions I had with my lab mates especially with Ng Yew Seng helped me in getting new ideas for my research.

I would like to thank my flat mates and friends Mekapati Srinivas, Velu Perumal, Sukumar Balaji, and Selvarasu Suresh for making my off campus stay as peaceful and memorable.

Last but not the least, I thank my friends in National University of Singapore, Guntuka Sathish, Yelneedi Sreenivas, Yelchuru Ramprasad, and Konda Murthy for making my journey as pleasurable and memorable.

TABLE OF CONTENTS

To my Mother Vijayalakshmi

and

Brother Suresh

SUMMARY

Biological strains are increasingly used to produce amino acids, vitamins, antibiotics, metabolites, enzymes, solvents, organic acids and bulk chemicals. Millions of tons of biotechnology products are produced each year for a multi-billion dollar market. Considering the depletion of fossil fuels, environmental issues and increase in use of therapeutic proteins, the number and scale of bioprocesses will significantly increase in the future. Improvement of strains by modifying genetic targets to increase yield of desired products is the key issue for the successful and economical operation of bioprocesses.

The advent of microarray technology has created a deluge of gene expression data by virtue of its ability to measure the expression levels of thousands of genes simultaneously. This data, when suitably mined, can provide understanding of the physiological state of cells and thus enable the identification of genetic targets for strain improvement.

In this thesis, a data-driven framework is proposed for identifying genetic targets for strain improvement. The framework contains different methods for identifying differentially expressed genes, clustering of genes, cluster validation, and integration of complementary datasets to identify genetic targets for strain improvement. Novel methods based on multivariate statistics are proposed for each step of the proposed framework. In the first step, a method using Principal Components Analysis is proposed to discover the genes differently expressed between wild-type strain and the strain pro-

ducing desired product. These differently expressed genes shed light on the changes in the cellular processes due to genetic modifications done to strains and hence provide the clues to manipulate the genotype of cells to have desired phenotype.

In the second step, clustering and cluster validation algorithms to group genes into disjoint and homogenous clusters based on their similarity in their expression profiles are proposed. Since genes within a cluster are more similarly expressed, the potential roles of uncharacterized genes can be hypothesized based on the expression similarity with the other known genes. In contrast to the generally used clustering algorithms that induce a fixed topological structure on cluster, the proposed algorithm takes into the consideration the actual geometric shape of the gene clusters in the expression space. It is devised to work effectively even if some of the clusters lie in subspaces due to the inter-dependency of the different time-points. Then, methods based on an evolutionary approach for spherical clusters and PCA subspace similarity metric for ellipsoidal clusters are proposed to find the number of clusters in the expression dataset.

In the last step, a Bayesian method is introduced to integrate the gene expression data with the genome-wide Transcription Factor-DNA interaction data in order to reliably identify TFs that are targeted for strain improvement. All the methods proposed in this thesis are tested with artificial as well as expression data from different organisms. A real case study involving improvement of *Escherichia coli K12* strain producing recombinant protein by identifying genetic targets is used to illustrate the integration of the above steps.

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

AcCoA	Acetyl Coenzyme A
ANCOVA	Analysis of covariance
ANOVA	Analysis of variance
BIC	Bayesian Information Criterion
cDNA	complementary DNA
DEG	Differentially Expressed Genes
DNA	Deoxyribonucleic Acid
F6P	Fructose-6-Phosphate
FBA	Flux Balance Analysis
GA	Genetic Algorithms
GG	Gath and Geva clustering
GK	Gustafson and Kessel clustering
G3P	Glyceraldehyde-3-Phosphate
G6P	Glucose-6-Phosphate
HEC	Hyper Ellipsoidal Clustering
HSF1	Heat-shock transcription factor 1
JC	Jaccard Coefficient
mRNA	messenger Ribonucleic Acid
MS	Mass Spectrometry
NADPH	Nicotinamide adenine dinucleotide phosphate
NEPSI	Net Principal Subspace Information Index

NIFTI	Net InFormation Transfer Index
OD	Optical Density
PB	Plasmid Bearing strain
PCA	Principal Component Analysis
PCs	Principal Components
PPP	Phosphate Pentose Pathway
PYR	Pyruvate
RMSECV	Root-Mean Square Error of Cross-Validation
SIMCA	Soft Independent Method of Class Analogy
SOM	Self-Organizing Map
TCA	Tricarboxylic Acid cycle
TFs	Transcription Factors
TIC	Total Information Content
WT	Wild-type strain
2DE	Two Dimensional Gel Electrophoresis

NOMENCLATURE

Chapter 2

C_i	The i^{th} cluster in a partition
d	Distance metric used for clustering
d_2	Euclidean distance
DB	Davies-Bouldin Index
DI	Dunn's Index
D_m	Sum of distances of all pairs of objects
$f_i(x_r\|\theta_i)$	Density function
I	Index of partition quality
J_k	Objective function for clustering
k	Number of clusters
k_{min}	Minimum number of clusters
k_{max}	Maximum number of clusters
k_{opt}	Optimal or correct number of clusters
L	Likelihood function
m_i	Centroid of cluster
n	Number of replicates
n_m	Number of objects in C_m
N	Number of objects to be clustered
p	Dimensionality of feature space
s_i	Standard deviation of replicates of i^{th} gene

S	Silhouette Width of partition
t_i	The t-statistic for i^{th} gene
W_k	Within-cluster dispersion
$\overline{x}_i$	Mean of expression replicates of i^{th} gene
x,y	Objects for clustering
X	Data matrix for clustering
τ_r^i	Probability of r^{th} object belongs to i^{th} component
δ	Inter cluster distance
Δ	Intra cluster distance
μ	Mean of Gaussian distribution
Σ	Covariance matrix of Gaussian distribution

Chapter 4

E	Residual matrix
g_i	i^{th} Gene in expression data
k	Number of PCs used for modeling expression data
MD	Mahalanobis Distance
n	Number of genes
$\mathbf{p_i}$	Loading vectors of PCA
P_i	p-value of gene i
S	Covariance matrix of gene expression data
t	number of time-points
X	Gene expression data

$\mathbf{z_i}$	Scores vectors
z_i^{Δ}	Difference of scores for i^{th} gene
Z^{Δ}	Difference of scores matrix
$\overline{Z}$	Mean of difference of scores
λ_i	Eigenvalues of Covariance matrix
Σ	Covariance matrix of difference of scores

Chapter 5

A	Norm matrix
C	Clustering partition
D	Distance metric used for clustering
J	Objective function for clustering
k	Number of clusters
l_j	Number of PCs used for j^{th} cluster
M	Population size for GA
n_j	Number of genes in j^{th} cluster
N	Number of generations for GA
p	Number of time-points
p_m	Mutation probability for GA
p_r	Probability function for reassignment in GA
P	Population of GA
Q	Q statistic
Q_α	Confidence limit for Q statistic

Q_r	Residual Q statistic
s_i i^{th}	Clustering solution
T^2	Hotelling's T^2
T^2_α	Confidence limit for Hotelling's T^2
T^2_r	Residual Hotelling's T^2
v_j	Centroid of j^{th} cluster
x_{ij}	Expression level of i^{th} gene in j^{th} time-point
X	Gene expression data
ρ	Volume of cluster
ρ_{pca}	Volume of cluster in PCA subspace
ρ_{res}	Volume of cluster in residual subspace
μ_{ij}	Cluster membership term
λ	Eigenvalue of Covariance matrix
Σ	Covariance matrix

Chapter 6

C^i_k	i^{th} cluster in k^{th} generation
d	Distance metric used for clustering
D^i_k	Direction of change of information for i^{th} cluster
g^i_k	Change of information for i^{th} cluster
G_k	Generation k
G_{max}	Maximum number of generations
IC	Information content of a partition

k	Number of clusters
$k_{optimal}$	Optimal number of cluster
m	Number of features or assays
M_k^i	Magnitude of information change of i^{th} cluster
n	Number of genes in a cluster
N	Number of genes in the dataset
p^{ij}	Fraction of objects inherited by j^{th} offspring from i^{th} parent
r	Number of offspring of a parent cluster
v_X,v_Y	centroids of dominant offsprings X and Y
X,Y	Dominant offspring of a parent clusters
$\|X\|$,$\|Y\|$	Number of objects in X and Y
Z	Gene expression data
δ_{xy}	Centroid distance between X and Y
Δ_X,Δ_X	Radii of X and Y
μ	Mean of a Gaussian distribution
Σ	Covariance matrix os Gaussian distribution

Chapter 7

A,B	Clusters or groups of genes
C	Clustering partition
C_{opt}	Partition with optimal number of clusters
d_i	Distinctness of cluster
$E(R)$	Entropy of random variable R

$E(C_i)$	Entropy of i^{th} cluster in a partition
I	Identity matrix
k	Number of clusters
k_{opt}	Optimal number of clusters
l	Number of PCs
L,M	Eigenvector matrices of clusters A and B
m	Number of variables or assays
N	Number of genes in gene expression dataset
$NEPSI_{kopt}$	NEPSI for optimal k
p_i	Fraction of total genes or objects in cluster
R	Random variable
S	Sample covariance matrix of a cluster
S_{PCA}	PCA similarity factor
S^{λ}_{PCA}	Eigenvalue modified PCA similarity factor
v_j	Variables or assays
w_j	Coefficient vectors in PCA
x_{ij}	Expression level of genes x_i in j^{th} assay
X	Gene expression data matrix
z_j	Principal Components or Scores vectors
λ_i	Eigenvalues
θ_{ij}	Angle between i^{th} and j^{th} PCs
θ_T	Threshold for distinct of clusters
Σ	Covariance matrix of Gaussian distribution

Chapter 8

b_{ij}	p-value for TF-gene interaction
B	Genome-wide TF-gene interaction data
m	Number of genes
n	Number of time-points
$p(x)$	Probability density function of x
$p(x/w_i)$	Probability density function of x given class w_i
$P(w_i))$	a priori probability
$P(w_i/x)$	a posterior probability of x belonging to class w_i
t	Number of TFs
w_i	Gene class i
X	Gene expression data
μ	Mean of class
Σ	Covariance of class

1. INTRODUCTION

Bioprocesses using microbial strains for producing metabolites, proteins and vitamins are becoming prominent in many industries including chemical, pharmaceutical, health care, food, and agriculture industries. Approximately, one million tons of amino acids with market value over $3 billion dollars are being produced every year through fermentation processes (Demain, 2000). Currently, 5% of all chemicals produced including fuels, polymers, and specialty chemicals are through the bioprocess route. The share of bioprocesses in chemical production is expected to increase to 10-20% by 2010 (Bachmann, 2005). The use of microorganisms for the production of pharmaceutical drugs is enormous. Approximately, 165 bio-pharmaceutical drugs are currently in use worth approximately $30 billion; this market is expected to increase to $70 billion by 2010 (Walsh, 2006). Considering the depletion of fossil fuels, the use of microorganisms for the conversion of biomass to useful products is also of great importance for a sustainable future.

The importance of fermentation processes is due to the ability of microorganisms to accept a variety of carbon sources and the diversity of chemical reactions they are capable of carrying out. However, natural microorganisms produce no (or at best in small amounts) compounds of interest. Increase in yield and productivity of desired compounds is essential for successful and economical viability of bioprocess industries. The improvement of yield is generally achieved by developing improved strains (Stephanopoulos, 2002).

1.1 Strain Improvement

Biological production of chemicals and proteins starts with the identification of strains that are suitable for the production of desired products. The next stage is the optimization of bioprocess for economical production of desired product. Optimization of bioprocess can be achieved either at the process level or through strain improvement (Lee *et al.*, 2005). Since the improvement of strains that yield more desired products has greater impact on economics, strain improvement programs have attracted more interest recently (Lee *et al.*, 2005).

Initially, approaches for strain improvement were greatly dependent on mutagenesis and screening. The development of recombinant DNA technology has revolutionized the strain improvement process by enabling modifications at genetic level. Now, researchers are using directed approaches for strain improvement through modification of genes. The first step in strain improvement program is to select genetic targets for modification that results in higher yield of desired product (Nielsen, 1998). However, it is very difficult to identify such genetic targets due to the complexity and redundancy of cellular processes. Understanding the interactions among different compounds inside the cells is essential to successfully identify genetic targets.

The classical way of identifying gene targets relied on biochemistry literature and knowledge about the organism. However, this approach is limited by the availability of literature. Recently, this approach has been complemented by constraints-based analysis of cellular metabolism, called Flux Balance Analysis (FBA) (Price *et al.*, 2004;

Edwards and Palsson, 2000). FBA is a constrained optimization procedure to identify the flux distribution through different pathways in a metabolic network. The genetic targets are identified such that more flux is directed towards desired pathway. Though FBA has been successful in some cases, it requires a mathematical model of metabolic reactions. Such a model is laborious to develop and specific to the organism. Also, the FBA approach uses only known biological processes and interactions. With the progress in molecular biology and advent of new technologies, it is now possible to collect comprehensive data even at the molecular level. These data capture the internal state of cells and hence useful for understanding the functioning of cell. The DNA microarray technology is one such technique.

The DNA microarray allows measurement of expression levels of genes at the genome-scale. The data contain information about almost all the molecules expressed in the cells during the bioprocess. There is a lot of potential to use this data to identify the genetic targets for improving microbial strains (Van der Werf, 2005). In contrast to model based approaches, data-driven methods make fewer assumptions and are not limited by known interactions. However, suitable statistical data-mining approaches are essential to extract useful information from these data.

In this thesis, a data-driven framework is proposed for genetic target selection for improving biological strains. Novel data-mining methods suitable for gene expression data mining are proposed and validated using artificial and real expression datasets. In the following sections, gene expression data generation and challenges in mining these data are described followed by the overview of thesis.

1.2 Large Scale Data Generation: Microarrays

The central dogma of biology is that genes are first transcribed to messenger RNA (mRNA) and mRNA is translated to proteins as shown in Figure 1.1. Measurement of internal and external variables that determine the behaviour of cells is important for understanding cell functioning. The internal state and response of cells to changes in environment are sensitively reflected in the mRNA levels of all genes (Lander, 1996). Hence, simultaneous monitoring the expression levels, *i.e.* mRNA levels, of the genes is essential. Initially measurement of mRNA levels was limited to a handful of genes. The development of DNA microarray technology enables the simultaneous measurement of mRNA levels of all the genes at the genome-scale (Schena *et al.*, 1995).

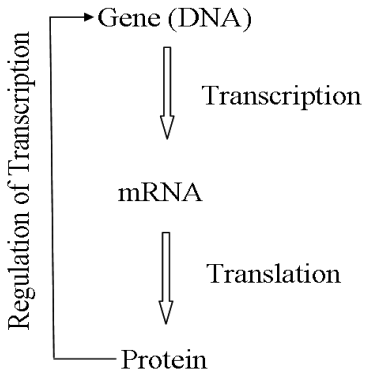

Fig. 1.1. The central dogma of biology. Genes are first transcribed to mRNA and then translated to proteins.

Microarrays exploit the capacity of nucleic acid sequence to recognize the complementary sequence through base-pairing. The process of recognition, called as hybridization, is extremely parallel—every sequence in the mixture can identify its complementary sequence. A microarray slide consists of large number of DNA sequences (for example, all the 6200 known and predicted genes of *Saccharomyces cerevisiae*

called probes. The extracted mRNA from the cell is labeled with fluorescent dye, reverse transcribed to produce the DNA sequence complimentary to the sequence attached to the slide and hybridized with these probes. The slides are excited with light and amount of fluorescence at each probe is measured. The amount of florescence is proportional to the amount of specific mRNA present, thus the level of expression of the corresponding gene can be inferred.

Now, it is possible to spot thousands of genes on a single microscope slide and quantify the expression levels of each gene. DNA microarrays thus provide a natural vehicle for systematic and comprehensive exploration of genomes (Brown and Botstein, 1999). Recently, the gene expression data is complimented by proteomics data, *i.e.* the proteins produced by cell, since mRNA levels do not always correlate with the protein concentrations (Ideker *et al.*, 2001). Two dimensional gel electrophoresis (2DE) and mass spectrometry (MS) are the two important techniques generally used to identification and quantification of proteins present in cell. However, there are several limitations for protein quantification. The accuracy of protein measurement is low due to the complex and dynamic nature of proteins. Also, large-scale protein quantification is not possible with the 2DE and MS techniques (Beranova-Giorgianni, 2003).

1.3 Time-Course Gene Expression Data

In the early days of microarray experiments, expression data was measured for a given condition. These provide a snapshot of the expression levels at that particular condition and hence it does not provide any insights into the dynamics of the biological process. Current gene expression research focuses on time-course experiments where

expression levels are measured at multiple time-points. The time-course gene expression data capture the changes happening within the cells during a bioprocess. The rest of the thesis focuses mainly on time-course gene expression data analysis.

1.4 Challenges in Gene Expression Data-mining

Though gene expression data provides the state of a cell by measuring the expression levels of almost all its genes, the information is hidden in the data. We need efficient data-mining methodologies to uncover the hidden patterns and identify the genetic targets for strain improvement. There are several challenges for the analysis of gene expression data. Some of the important ones are given below:

1. The main question in microarray experiments is that which genes are differentially expressed between two or more conditions, say between a wild-type and an organism producing desired product or normal cells and cancerous cells? Differentially expressed genes are the ones which explain the difference in molecular mechanisms leading to the phenotypic changes. Currently available techniques for identifying differentially expressed genes (DEG) are not suitable for time-course datasets.

2. Clustering of genes into different clusters such that genes within a cluster are more similar in expression is an important challenge in gene expression data analysis. This organization of genes into clusters reveals the broad organization of genetic programs and execution of the regulatory program in the cells. The understanding of cell function facilitates the identification of genetic targets.

The currently available algorithms for clustering identify only spherical clusters where as clusters can be of different geometrical shapes.

3. Another important challenge in gene expression data analysis is the identification of number of clusters in a dataset. Number of clusters is one of the key parameters that has to be specified a priori to many clustering algorithms. The results with different number of clusters varies significantly (Bezdek and Pal, 1998). Though there exists a lot of literature on finding number of clusters, they are dependent on the characteristics of the data. Methods that work on a particular type of data may not be suitable for another kind of data. So, methods specifically suited for gene expression data are needed.

4. Another important challenge is integration of multiple and complementary genomic datasets in order to increase the reliability of predictions. Though gene expression data provide the expression levels (mRNA levels) of thousands of genes, it does not provide any information about the regulation of expression. Specific kind of proteins, called Transcription Factors (TFs), bind to genes and regulate their expression according to the cell's requirement. To understand the functioning of cells and to modify them, it is essential to find which TF regulates which genes. Fortunately, there is a genome-scale technique, called Genome-Wide Location experiments, for identification of TF-gene binding. However, as other genome-scale techniques, the genome-wide location data contain noise. To enhance the reliability of TF-gene interactions, it is necessary to combine complementary datasets such as gene expression and genome-wide location data.

1.5 Thesis Overview

In this thesis, novel methods are proposed for identifying DEG, clustering genes and finding number of clusters. A Bayesian approach for combining gene expression data with genome-wide location data is proposed. A systematic framework that combines these methods in a principled way to identify the genetic targets for strain improvement is also proposed.

In Chapter 2, several methods currently available for identifying DEG, clustering and finding number of clusters are reviewed. In Chapter 3, a data-driven framework that combines several data-mining techniques to identify targets for strain improvement is proposed. A Principal Component Analysis (PCA) based approach for identifying DEG in time-course data is presented and validated in Chapter 4. A novel clustering method that identifies ellipsoidal clusters in gene expression data is proposed in chapter 5. An evolutionary approach for finding number of clusters in gene expression data is proposed in Chapter 6. In Chapter 7, a method for finding distinct clusters in gene expression data through comparing clusters in PCA subspaces is presented. A Bayesian approach for combining complementary genomic datasets is proposed in Chapter 8.

In Chapter 9, a complete case study is provided where the proposed data-driven framework is used for identifying genetic targets for improvement of *Escherichia coli* strain. Conclusions and suggestions for future work are provided in Chapter 10.

2. LITERATURE REVIEW

Microarray technology has transformed the genomic research from studying handful of genes to genome-scale by facilitating the measurement of expression levels of thousands of genes simultaneously (Schena *et al.*, 1995). There are two different types of microarray technologies commonly used in genomic experiments, namely cDNA microarray and Oligonucleotide arrays. cDNA microarray is a specially coated glass microscope slide to which DNA sequences are printed at fixed locations, called spots, using a robotic arrayer (Brown and Botstein, 1999). With up-to-date computer controlled high-speed robots, more than 20000 spots can be printed on a single slide, each representing a single gene. Affymetrix is one of the main promoters of oligonucleotide arrays. Affymetrix's 'GeneChip' arrays consist of small glass plates with thousands of oligonucleotide DNA probes (short stretches of nucleotides, typically 25-mers) attached to their surface (Lipshutz *et al.*, 1999). The oligonucleotides are synthesized directly onto the surface using a combination of semiconductor-based photolithography and light-directed chemical synthesis. With this high-tech approach, very large numbers of mRNAs can be probed at the same time.

To measure the expression levels of genes, the total mRNA from the cells is extracted, labeled using fluorescent dyes and reverse transcribed to cDNA. The sample is then hybridized with the arrayed DNA spots. After hybridization, a laser microscope illuminates each spot and measures fluorescence intensities. The gene expression levels are estimated from the fluorescence intensities. For measuring the relative expression of a gene in two cell populations (control and sample), different fluorescent dyes (gen-

erally red for sample and green for control) are used.

Many researchers explored the gene expression at genome-scale with microarrays. DeRisi *et al.* (1997) published the first whole-genome gene expression measurements (approximately 6400 distinct cDNA sequences), a seven-point time series on the diauxic shift (transition from sugar metabolism to ethanol metabolism) in yeast using cDNA microarrays. Recently, Alizadeh *et al.* (2000) used cDNA microarray data to discover previously unknown sub-types within Diffuse Large B-cell Lymphomas, associated with significantly different survival of patients. Wodicka *et al.* (1997) used oligonucleotide chip technology to do genome-wide analysis on yeast gene expression. Cho *et al.* (1998) employed oligonucleotide microarrays to query the abundances of 6220 mRNA species in synchronized *Saccharomyces cerevisiae* batch cultures.

There is a huge potential to use these large scale gene expression data for understanding functioning of cells and identifying genetic targets for strain improvement. However, identification of genetic targets for strain improvement requires extraction of information from gene expression datasets using statistical data-mining techniques. This includes identification of differentially expressed genes, clustering of genes, finding number of clusters, etc. Also, it is essential to integrate multiple and complementary genomic datasets due to the inherent limitations of individual datasets to provide all the information about the cell. Here, currently available data-mining techniques are reviewed.

2.1 Identifying Differentially Expressed Genes

Microarray expression profiling is often carried out to identify genes whose expression change across biological conditions (Slonim, 2002). This includes comparison of gene expressions from one group with another group and delineate a list of genes ranked according to their respective differential expression (Steinhoffand and Vingron, 2006). Two types of expression profiling can be differentiated, static and time-course. In the static type, snapshots of gene expression levels are measured in two different cell populations, such as normal and diseased (Alizadeh *et al.*, 2000). Genes that are differentially expressed in the diseased cells, compared to normal cell population, disclose pathways related to the disease and also serve as signature of the disease. However, measuring expression levels irrespective of time does not provide information about the dynamic interactions that characterize the cellular processes (Fielden *et al.*, 2002). This necessitates time-course experiments where gene expression levels are measured at different time-points and across biological conditions such as wild-type and gene-knockout (Zhu *et al.*, 2000), normal and stimulated cells (Calvano *et al.*, 2005), etc.

Several methods have been proposed in literature to identify DEG in static experiments. The simplest technique is the calculation of fold change of gene expression between normal and diseased states. Genes with fold change above a user-defined threshold (say 2-fold) may be considered as differently expressed (DeRisi *et al.*, 1997). The fold change approach results in poor results since it does not consider the natural variation in gene expression levels (Kerr and Churchill, 2001). This necessitates the use of replicates in microarray experiments. Availability of replicates enable the applica-

tion of statistical methods for identifying DEG. Ranking of genes based on differential expression can be done based on t-statistic for each gene if replicates are available in both groups. The gene specific t-statistic is given by:

$$t_i = \frac{\overline{x}_i^1 - \overline{x}_i^2}{\sqrt{\frac{s_i^1}{n^1} + \frac{s_i^2}{n^2}}} \tag{2.1}$$

where $\overline{x}_i$, s_i are the mean and standard deviation of replicates of i^{th} gene, n is the number of replicates. The superscripts indicate the conditions 1 and 2. Problems arise when the denominator of Equation 2.1 becomes very small due to the small expression levels. Several penalizing factors that artificially increase the variation are proposed to circumvent this problem (Tusher *et al.*, 2001; Efron *et al.*, 2001; Pan *et al.*, 2003). More details and comparison of several methods for identifying DEG in static case are available (Pan, 2002; Troyanskaya *et al.*, 2002). These methods are not directly applicable for time-course experiments where differential expression has to be calculated globally in the temporal space and not just between corresponding time points (Storey *et al.*, 2005).

Recently, several methods have been proposed to identify the differentially expressed genes in time-course data. Bar-Joseph *et al.* (2003a) proposed a method that represents expression profiles as continuous curves and then uses a global difference between the curves to identify differentially expressed genes. In their approach, clustering of genes is used as a preprocessing step; although simple, this makes the method computationally expensive for large datasets. Storey *et al.* (2005) proposed a method that measures the improvement in goodness-of-fit when a single curve is used to fit the data from both conditions compared to fitting a separate curves for each condition. If

the improvement in goodness-of-fit is significant then that particular gene is considered as differentially expressed. Their approach treats all genes as equal irrespective of their expressions levels in the experiments. This leads to the identification of genes with low expression in both conditions as differentially expressed genes. Conesa *et al.* (2006) proposed a regression-based approach that models the expression profile of each gene with time as regressor and tests the hypothesis on the equality of regression coefficients. A similar method is proposed by Vinciotti *et al.* (2006) where the expression profiles are fitted using cubic polynomials and tested for similarity of coefficients. Modeling individual genes is generally not recommended due to noise in the microarray data (Bar-Joseph *et al.*, 2003c). Cheng *et al.* (2006) proposed an approach that represents the time-course data from both conditions as two different gene relationship networks where each node is a gene and each edge links two genes. Differentially expressed genes are identified by comparing the neighborhood, genes that have very similar and very dissimilar expression profiles in both networks. Genes with dramatic change in neighborhood are deemed as differentially expressed. Since the actual expression of gene is not directly compared in both conditions, genes similarly expressed in both conditions can be declared as differentially expressed if their neighbors are changed. Reverter *et al.* (2006) proposed a method that identifies genes that are simultaneously differentially expressed and differentially connected. However, they quantify the difference in expression of a gene as the sum of differences in individual time-points which may not capture systematic variations. Methods based on Analysis of Variance (ANOVA) (Park *et al.*, 2003) and Analysis of Covariance (ANCOVA) (Tabibiazar *et al.*, 2005) models have also been proposed specifically for replicated time-course data.

Each one of the currently available methods for identifying differentially expressed genes in time-course data have particular drawbacks associated to them. They do not consider natural dependencies among different time-points and the noise in the data. A novel statistical method for identifying differentially expressed genes in time-course data is proposed in this thesis. The proposed method uses PCA that considers the correlation among different time-points and identifies fundamental patterns in the data that are independent of each other. The scores of genes on these fundamental patterns are used to identify the differentially expressed genes. The noise is discounted from the analysis by considering only the most significant Principal Components (PCs) in the analysis.

2.2 Clustering Expression Profiles

Microarrays provide a deluge of gene expression data by simultaneous measurement of expression levels of thousands of genes. This large amount of gene expression data necessitates use of data-mining techniques to organize and extract useful information from these data. Clustering is one such technique widely used for gene expression data analysis.

The objective of clustering is to separate a finite set of objects into a few discrete groups, called clusters, with high internal homogeneity and external separation (Hartigan, 1975). Internal homogeneity and external separation means that objects within a cluster are similar to each other and dissimilar to objects in other clusters. The similarity between the objects is measured in the feature space using a suitable distance

metric. The most widely used distance metric is the Euclidean distance. The Euclidean distance, d_2, between two objects, x and y, is given by:

$$d_2 = (\sum_{i=1}^{p}(x_i - y_i)^2)^{(1/2)} \tag{2.2}$$

where p is the dimensionality of feature space. Other well known distance metrics are Pearson correlation, standard correlation coefficient, mutual information, etc (Jiang *et al.*, 2004).

In gene expression data analysis, genes or samples (assays) are clustered based on similarity in expression. Two way clustering, *i.e.* simultaneous clustering of both genes and samples is also possible (Eisen *et al.*, 1998). Clustering of genes results in clusters of co-expressed genes whereas sample clustering results in correlated samples. Clustering of genes has several benefits: (1) Genes having similar expression profiles often function together, hence, clustering of genes leads to the identification of gene functions. (2) Similarly expressed genes are often regulated by the same TFs leading to identification of TFs. (3) In case of sample clustering, new subtypes of diseases or molecular level signatures of diseases can be identified which enables development of customized diagnostic procedures.

Clustering methods can be broadly classified as hierarchical and partitional approaches based on the type of results from these algorithms (Jain *et al.*, 1999). Hierarchical clustering method arranges the objects into a hierarchy based on their similarity to each other. The partitional clustering algorithms create a predefined number of

disjoint clusters that optimizes the given objective function (generally the sum of distance of objects to cluster centroids). In the following sections, the hierarchical clustering and two partitional clustering methods—k-means and model-based clustering—methods are described. These clustering methods are widely used for gene expression data analysis.

2.2.1 Hierarchical clustering

Hierarchical clustering generates a hierarchical series of nested clusters which can be visualized as a tree generally called as *dendrogram*. The branches of the dendrogram represent clusters and the branch length represents the similarity between clusters. The clusters are extracted from the dendrogram by cutting the it at different levels (Jiang *et al.*, 2004). Hierarchical clustering can be further divided as *agglomerative* and *divisive* approaches. Agglomerative hierarchical clustering starts by considering each object as single cluster. In each iteration, the distance between all pairs of clusters is calculated and the pair with the smallest distance is merged. The merger of clusters continues till the last pair and the algorithm terminates. The divisive approach considers all the objects as single cluster initially and sequentially splits clusters till only singleton clusters with one object remains.

For agglomerative clustering, different approaches are available for measuring the cluster proximity for merging clusters. In *single linkage* clustering, the minimal distance between the clusters is used. The maximum distance between clusters is used for *complete linkage* clustering. Average distance between the all objects in pair of clusters is used for *average linkage* clustering. For divisive clustering approach heuristics

based on graph theory are generally used for spiting clusters (Jiang *et al.*, 2004). Eisen *et al.* (1998) pioneered the use of hierarchical clustering for gene expression data. They developed a software for hierarchical clustering and visualization of results.

2.2.2 k-means clustering

k-means is a partitional clustering technique that partitions a dataset into specified number of clusters while minimizing an objective function (MacQueen, 1967; Hartigan, 1975). The general objective function is the sum, J_k, over all k clusters, of the within-cluster sums of object-to-cluster-centroid distances given by:

$$J_k = \sum_{i=1}^{k} \sum_{x \in C_i} d(x, m_i) \tag{2.3}$$

where d is the distance metric used for clustering, x represents the object belonging to cluster C_i and m_i is the centroid of cluster C_i.

k-means starts with the random assignment of data objects to k clusters. Then the loop containing the calculation of cluster centroids and redistribution of data objects to clusters that minimizes objective function is activated and carried out till the objective function reaches a local minimum. Since k-means terminates at a local minima, it generally gives different results for different initiations. To overcome this problem, the multi-start method is employed in this thesis. For a given value of k, the procedure is repeated with different initial guesses (replicates) and the partition with the lowest J_k is selected as the best partition. Tavazoie *et al.* (1999) used k-means clustering for clustering yeast *Saccharomyces cerevisiae* cell-cycle data and identified novel TFs.

2.2.3 Model-based clustering

Model-based clustering approach assumes that the data to be clustered is generated from a finite mixture of underlying probability distributions, where each component in this mixture corresponds to a different cluster (Banfield and Raftery, 1993). Given a dataset X consisting of N objects $\{x_1, x_2...x_N\}$, and number of components (clusters), k , the objective is to estimate the parameters $\Theta = \{\theta_i | 1 \leq i \leq k\}$ and $\Gamma = \{\tau_r^i | 1 \leq i \leq k, 1 \leq r \leq N\}$ that maximize the likelihood function given by:

$$L(\Theta, \Gamma) = \sum_{r=1}^{N} \sum_{i=1}^{k} \tau_r^i f_i(x_r|\theta_i) \tag{2.4}$$

where τ_r^i is the probability that a data object x_r belonging to i^{th} component, and $f_i(x_r|\theta_i)$ is the density function of x_r of i^{th} component.

In Gaussian mixture models, each component is represented by a multivariate normal distribution with mean μ_i and covariance matrix Σ_i. The parameters are generally estimated using the Expectation Maximization (EM) algorithm (Redner and Walker, 1984). Banfield and Raftery (1993) proposed a general framework for identifying clusters of different shapes using model-based clustering. With this framework, by relaxing some of the parameters in covariance matrix, clusters of different geometrical shapes can be identified: *Equal-volume spherical (EI), Unequal-volume spherical (VI), and elliptical models*. More details of these schemes are available in Yeung *et al.* (2001). Fraley and Raftery (1999) implemented these schemes as a $Matlab^{TM}$ toolbox. The model-based clustering with Equal-volume spherical scheme (EI) is used in this thesis since it requires estimation of least number of parameters from data compared to other

schemes. Also, the EI scheme is more accurate and can be used with large number of clusters.

Hierarchical clustering is widely used for gene expression data clustering due to its ability to visualize the clustering results (Eisen *et al.*, 1998). But, there are several problems with hierarchical clustering. Hierarchical clustering follows a series of cluster merge/split based on local decision not on global objective. Hence, any bad merge or split occurred at any step cannot be corrected in later steps. Besides this, hierarchical clustering generates several singleton clusters and hence difficult to extract meaningful clusters from the dendrogram (Leach and Hunter, 2000). Several other clustering approaches such as Self-Organizing Maps (SOM) (Tamayo *et al.*, 1999), graph-theoretic (Sharan *et al.*, 2003), fuzzy clustering algorithms (Dembele and Kastner, 2003) and density based clustering (Wicker *et al.*, 2002) are proposed for gene expression data analysis. Gibbons and Roth (2002) compared different clustering algorithms using real gene expression data with functional enrichment of clusters as objective. The study shows that the performance of hierarchical clustering is more or less equal to random clustering. It also shows that partitional clustering perform better than hierarchical clustering.

Although the partitional clustering methods are successful in some cases, they have the following drawbacks. Any partitional clustering algorithm has two critical components: (1) the distance metric used for measuring the similarity of expression profiles, and (2) an algorithm for assigning each gene to a cluster. There is a challenge associated with each component:

1. The generally used Euclidean distance metric identifies spherical clusters whereas the objective of clustering is to identify the natural structure in the data.

2. The optimization algorithm for assigning genes to clusters generally leads only to a local minima whereas reaching global minimum is preferred.

Methods have been proposed with adaptive distance metrics that can identify clusters of different shapes. Gustafson and Kessel proposed a new clustering method known as GK clustering with adaptive distance metric (Gustafson and Kessel, 1979). Gath and Geva proposed similar approach for identifying clusters of different shapes (Gath and Geva, 1989). A Self-organizing neural-network based algorithm is proposed by Mao and Jain (1996) for identifying hyper-ellipsoidal clusters (HEC) in the data. The common feature of these methods is the adaptation of distance metric to the shape of cluster by estimating the covariance matrix of cluster. Problems arise in estimating the covariance matrix when the number of objects in the cluster are smaller than number of features or due to the linear correlation among features or objects (Babuska *et al.*, 2002). In such cases, the covariance matrix becomes singular or close to singular and cannot be inverted for calculation of adaptive distance metric.

The singularity of covariance matrix is common in time-course gene expression data analysis as different time-points are correlated to each other (Schafer and Strimmer, 2005). So, the methods based on adaptive distance metric such as GK, GG and HEC are not suited for gene expression data. In this thesis, a distance metric is proposed which takes the natural structure, *i.e.* 'shape', of the cluster into consideration while calculating the distance and able to identify clusters even the covariance matrix

becomes singular. To address the issues with local minima, the proposed method uses a Genetic Algorithm (GA) to optimize the objective function..

2.3 Finding Number of Clusters in Expression Data

The selection of clustering parameters affects, directly or indirectly, the resulting partition. In many cases, the optimal specification of number of clusters, k, is difficult especially if there is inadequate biological understanding of the system. A suboptimal specification of number of clusters can generally result in misleading results — either all classes may not be identified or spurious classes may be generated (Bezdek and Pal, 1998).

Several methods have been proposed for finding the 'best' number of clusters. Comprehensive reviews of several methods are available in Milligan and Cooper (1985) and Halkidi *et al.* (2001). Methods for finding the number of clusters in a dataset can be classified as global or local methods (Gordon, 1999). Global methods evaluate the clustering results by calculating some measure over the entire dataset. Local methods considers the pairs of clusters and test whether they should be amalgamated. The disadvantage of the global methods is that there is no definition for the measure for $k = 1$, *i.e.*, the global methods do not provide any clue whether the data should be clustered or not. Since the local methods consider the pair of clusters, they can be used to know whether data should be clustered or not. The disadvantage of local methods is that they need a threshold value or significance level to decide whether the clusters are amalgamated. The threshold or significance level is generally depends on the actual data and may not be available *a priori*. Apart from this, local methods are only suitable for eval-

uating the results from hierarchical clustering approaches where as the global methods are independent of clustering techniques.

In practice, global cluster validation methods are more popular than local methods as they can be applied to several clustering methods. The general procedure for finding the number of clusters requires the evaluation of quality of clusters generated by the clustering algorithm using an index, I. The procedure consists of the following steps (Halkidi *et al.*, 2001):

1. Run the clustering algorithm for each value of number of clusters, k, between $[k_{min}\, k_{max}]$. k_{min} is generally set to 2 and k_{max} is decided by the user.

2. For each value of k, calculate the index, I, using the compactness of individual clusters and the separation from other clusters.

3. Plot the index value as a function of k.

4. Identify the optimal value for number of clusters, k_{opt}, for which the index value is optimal (maximum or minimum).

In the next section, a brief description about methods that are frequently used for finding number of clusters is given.

2.3.1 Silhouette index

The Silhouette index (Rousseeuw, 1987) assigns a measure called Silhouette width to every object in the clustered partition. The value of the Silhouette width of an object is based on the average distance of that object to its own cluster and the minimum

of average distances to all other clusters in the partition. Consider a dataset with N objects. Let $a(i)$ be the average distance of the i^{th} object to its own cluster and $b(i)$ the minimum of average distances to other clusters. Then the Silhouette width of the i^{th} object is given by:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{2.5}$$

From the above equation, it is clear that $s(i)$ takes a value between [-1 1]. A value around 1 $(a(i) << b(i))$ indicates that the object is 'well-clustered', a value around 0 $(a(i) \approx b(i))$ indicates that the object can be assigned to other cluster as well (i.e. marginally classified), and around -1 $(a(i) >> b(i))$ indicates that the object is 'mis-classified'. The Silhouette index, S, for a given partition is the average Silhouette width over all the objects in the dataset.

$$S = \frac{1}{N} \sum_{i=1}^{N} s(i) \tag{2.6}$$

Given several partitions of the dataset, the partition with the highest Silhouette index is selected as the optimal one.

2.3.2 Dunn's index

The Dunn's index (Dunn, 1974) identifies the partition that maximizes the ratio of the minimum inter-cluster distance to the maximum intra-cluster distance. This index

finds the partition with compact and well separated clusters. For a given partition with k clusters $(C_1, C_2...C_k)$, the Dunn's index, DI, is given by:

$$DI = \min_{1 \leq i \leq k} \left\{ 1 \leq \min_{j \neq i} j \leq k \left\{ \frac{\delta(C_i, C_j)}{\max_{1 \leq l \leq k} \{\Delta(C_l)\}} \right\} \right\} \tag{2.7}$$

where $\delta(C_i, C_j)$ is the inter-cluster distance between clusters C_i and C_j, and $\Delta(C_l)$ is the intra-cluster distance of cluster C_l. The value of the index depends on the definition of $\delta()$ and $\Delta()$. Six different variants of inter- and three variants of intra-clusters distances have been proposed, thus leading to 18 generalized Dunn's indices (Bezdek and Pal, 1998). An averaging scheme that combines these 18 generalized indices into a single normalized Dunn's index value for a given partition is available (Bolshakova and Azuaje, 2003). A partition with compact, well-separated clusters results in high minimum inter-cluster distance and low maximum intra-cluster distance and has the highest Dunn's index.

2.3.3 Davies-Bouldin index

The Davies-Bouldin index (Davies and Bouldin, 1979) also finds the clusters that are compact and well-separated. The Davies-Bouldin index, DB, for a partition with k clusters is given by:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} \left\{ \frac{\Delta(C_i) + \Delta(C_j)}{\delta(C_i, C_j)} \right\} \tag{2.8}$$

A partition with the lowest Davies-Bouldin index is selected as optimal one. Similar to Dunn's index, generalized and normalized forms of the Davies-Bouldin index can also

be developed; the latter is used in this thesis.

All the methods that are described above use either or both intra- and inter-clusters distances to identify the best partition. These methods are suitable for cases where the clusters are compact and well-separated from each other. These methods fail to detect the 'correct' number of clusters when the intra- and inter-cluster distances vary widely (Jonnalagadda and Srinivasan, 2004). Gene expression data contains clusters of different sizes, shapes, and there exist smaller clusters within the larger well-separated cluster (Jiang *et al.*, 2003). Hence, these distance based methods have limited applicability for gene expression data.

2.3.4 Other Methods

Recently, several methods have been proposed for finding number of clusters in gene expression datasets. Tibshirani *et al.* (2001) proposed the gap statistic that measures the difference between within-cluster dispersion and its expected value under the null hypothesis of uniform distribution *i.e. no clusters exit in data*. The k that maximizes the difference is selected. The gap statistic is given by:

$$Gap(k) = E(\ln(W_k)) - \ln(W_k) \tag{2.9}$$

where W_k is the within-cluster dispersion, and E indicates the expected within-cluster dispersion based on uniform distribution. The within-cluster dispersion is given by:

$$W_k = \sum_{m=1}^{k} \frac{1}{2n_m} D_m \tag{2.10}$$

$$D_m = \sum_{i,j \in C_m} d(x_i, x_j) \tag{2.11}$$

where D_m is sum of distances between all pairs of objects in cluster C_m measured using distance metric d, n_m is the number of objects in C_m.

Since the gap statistic uses within-cluster sum of squares around the cluster means to evaluate the within-cluster dispersion, this method is suitable for compact, well separated clusters. Dudoit and Fridlyand (2002) proposed a prediction based re-sampling method for finding the number of clusters. For each value of k, the original data is randomly divided into training and testing sets. The training data is used to build a predictor for predicting the class labels of the test set. The predicted class labels are compared to that obtained by clustering of test data using a similarity metric. This value is compared to that expected under an appropriate null distribution. The k for which the evidence of significance is the largest is selected. Ben-Hur *et al.* (2002) proposed a similar re-sampling approach where two random subsets (possibly overlapping) are selected from the data. The two random subsets are subsequently clustered independently and the similarity between the resulting partitions is measured. The distribution of this similarity (from multiple runs) is visualized for each k and the optimal number of clusters is selected where transition from high to low similarity occurs. The approach of Dudoit and Fridlyand as well as Ben-Hur et al. assume that the sample subset can represent the inherent structure in the original data which may not be true for small clusters. Furthermore, the user has to manually locate the transition in Ben-Hur et al. approach.

In this thesis, two different methods for identifying number of clusters in gene expression data are proposed. The first method named Net InFormation Transfer Index (NIFTI) evaluates a cluster partition based on separability of resultant clusters. A statistical test is proposed for testing the separability of clusters. NIFTI increases if clusters are separable and decreases otherwise. In contrast to other methods, NIFTI gives no weightage for larger inter-cluster distances and hence suitable for identifying number of clusters in complex data with varying inter-cluster distances. The partition with the largest value of NIFTI is identified as the optimal partition. The second method, called NEPSI, finds the maximum number of distinct clusters in the data. NEPSI evaluates the quality of partition using the distinctness of clusters. NEPSI increases with increase in number of distinct clusters and decreases if clusters are similar to each other. A similarity metric based on PCA is used for determining whether a cluster is distinct or not. A partition corresponding to the maximum value of NEPSI is selected as the best partition.

2.4 Integration of Genomic Datasets

Cells carry-out their complex functions by temporally altering the transcription rates of specific genes. The transcription rate of a gene is precisely regulated by the combinatorial action of activator and repressor proteins called Transcription Factors (TFs) that bind to the promoter regions of genes and regulate the expression of genes (Lee and Young, 2000). Strain improvement through modification of genetic targets requires the understanding of gene regulation. Primarily, we need to know which TFs regulate which genes. Even though analysis of genome-wide expression profiles enhances our understanding of cellular processes, individual datasets provide only a part of informa-

tion about the cell.

Attempts were made to identify TFs and their target regulated genes exclusively from gene expression data. Segal *et al.* (2003) proposed a method to identify the targets of regulators using gene expression data. Their procedure first identifies clusters of similarly expressed genes from gene expression data. The expression similarity of known and putative regulators (TFs) to these clusters establishes the link between TFs and their target regulated genes. There are several drawbacks in this approach. Microarray expression profiles do not distinguish between effect of direct binding of TF to a target gene and the indirect effect caused by intermediate TFs. So genes can have similar expression profile even though their regulators are different. Hence clustering of co-expressed genes is of limited use for TF assignment (Bar-Joseph *et al.*, 2003b). The approach of Segal *et al.* (2003) also assumes that expression profile of regulated genes depend on expression of their regulators. This assumption is not always valid. For example, during post-transcriptional modifications of TFs the expression of regulator does not change appropriately. Hence expression data alone is not adequate for identifying the regulators for genes.The remedy is to integrate different and complementary datasets to enhance the TF-gene interactions.

There are other genomic data sources that provide complementary information about TF-gene interactions. For example, the genome-wide location analysis method identifies the direct TF-gene physical interactions at genome-scale by combining the chromatin immunoprecipitation (ChiP) procedure with microarrays (Ren *et al.*, 2000). Though location data is highly useful, false positives and false negatives hinder the assignment

of TFs to genes. For instance, there is only moderate agreement between the genome-wide location studies of *Saccharomyces Cerevisiae* by Iyer *et al.* (2001) and Simon *et al.* (2001) for the same TFs (Futcher, 2002). However, by integrating gene expression and genome-wide location data one can extract useful and reliable information about regulation of genes.

Two different approaches have been proposed to combine these two datasets. In the first approach, a Bayesian network approach is proposed to combine gene expression data and location data (Hartemink *et al.*, 2001). A Bayesian network is a representation of joint probability distribution of several random variables (genes), expressed in the form of a directed a-cyclic graph and a conditional distribution for each variable (Friedman *et al.*, 2000). The genes make up the vertices of the acyclic graph. Hartemink *et al.* (2001) uses Bayesian networks with the location data influencing the model prior and the expression data influencing the likelihood. The identified network provides the links between TFs and their target genes. The Bayesian network makes assumption that the expression level of a genes is not dependent on the expression of descendent genes in the directed graph which is not reasonable. As another approach, Bar-Joseph *et al.* (2003b) proposed a method that combines the expression data with location data. In their approach, location data is used to classify genes into different sets such that genes in each set are bound by the same TFs. Then for each set, a minimum radius sphere (capturing the genes within the set) is found in gene expression data. Genes without any regulators (false negatives) in location data are classified into these sets if they fall in the sphere and have the combined probability of regulatory interactions lesser than a predefined threshold. One of the limitations of their method is

the computational complexity of finding the minimum radius sphere in the high dimensional expression data. The predictions that can be made from this approach are also limited due to the strict criteria of minimal radius sphere. Furthermore, this method is not extendable to other datasets such as gene / promoter sequences.

A Bayesian approach that reliably assigns TFs to genes by combining genome-wide location data with gene expression is proposed in this thesis. The proposed method is based on statistical theory and can be extended to new types of data. The proposed method uses genome-wide location data and gene expression data in an incremental way to reliably assign regulators to genes. A model is first developed using genes for which high-confidence TFs are available in the location data. This model is then used for assigning TFs to the remaining genes (i.e. those without reliable TF information) using expression similarity.

2.5 Gene Expression Data for Strain Improvement

Since the advent of microarrays, several researchers employed them to explore the cell functioning and identification of targets using DEG between wild-type (WT) and recombinant strains. Choi *et al.* (2003) used microarrays to compare the transcriptome profiles between WT *Escherichia coli* and recombinant strain producing Insulin-Like Growth Factor I Fusion Protein (IGF-If) and identified 600 DEG. Genes *prsA* (encoding a phosphoribosyl pyrophosphate synthetase) and the *glpF* (encoding a glycerol transporter) are selected as targets for improvement of production of (IGF-If). These two genes are involved in biosynthetic pathway of nucleotides and amino acids (Trp and His) and glycerol utilization, respectively (Choi *et al.*, 2003). Up-regulation of these

two genes resulted in increase of IGF-If from 1.8 g/liter to 4.3 g/liter. A similar approach is used by Wierckx *et al.* (2008) to understand the genetic basis for improved phenol production by a recombinant *Pseudomonas putida S12*.

Though there are some successful cases for identifying targets for strain improvement using gene expression data, development of systematic procedure for this purpose is still a vision (Bro and Nielsen, 2004; Lee *et al.*, 2005). In this thesis, a data-driven framework to identify the genetic targets for strain improvement is proposed. The proposed framework combines data-mining methods proposed in this thesis in a systematic way for identifying genetic targets. The framework is described in detail in Chapter 3.

3. OVERVIEW OF PROPOSED DATA-MINING FRAMEWORK FOR STRAIN IMPROVEMENT

As described in Chapter 1, the currently available methods for identifying genetic targets for strain improvement rely on known metabolic network of organisms. Since all the interactions among different compounds in the cell are not yet known, the currently used metabolic networks are incomplete and hence do not represent the true nature of the cells. The high throughput -omics data such as transcriptomics, proteomics and metabolomics overcomes this limitation by providing the genome-scale picture of the cells by virtue of their ability to measure several thousands of compounds simultaneously (Lee *et al.*, 2005). Especially, gene expression data contain vast amount of information by measuring expression levels of large number of genes simultaneously. There is a lot of potential to use these data for identifying genetic targets for strain improvement (Van der Werf, 2005). However, lack of suitable computational techniques to mine and extract useful information from these data hinders this objective. In this chapter, I propose a data-driven framework for mining gene expression data and integrating it with TF-gene interaction data to identify genetic targets for strain improvement.

The proposed data-driven framework for gene expression data mining to identify gene targets is shown in Figure 3.1. The first step in the framework is to compare the gene expression datasets from two or more experiments and identify genes differentially expressed between them. The datasets could be from wild-type vs high-producing strains or wild-type vs gene knock-out studies or wild-type vs cells producing recombinant proteins, etc. In all these cases, the changes exhibited at the phenotype originate

at the molecular level. By carefully assaying the expression of genes, it is possible to identify these molecular level changes. The differentially expressed genes serve the purpose of identifying the molecular level changes of cell functions between different conditions. For example, the production of recombinant protein or heterogenous genes in host cells creates metabolic burden on host cells leading to decreased growth (Choi *et al.*, 2006). The metabolic burden on the cells would be reflected as change in expression levels of several genes. The differentially expressed genes can thus be used for understanding the change in metabolism of host and can be used as targets for genetic modification in order to increase the host strain's performance.

Though differentially expressed genes provide useful information, typically there are a large number of such genes, which makes the analysis difficult. In the second step of the proposed framework, the differentially expressed genes are clustered into different groups based on similarity in expression. The grouping of genes into clusters reveals the organization of reprogramming occurring in the of cells to due to the recombinant protein production. A key step in clustering is to specify the number of clusters. The number of clusters to be used is identified by the cluster validation procedures. The comparison of clusters of genes between wild-type cells and cells producing recombinant protein reveals reprogramming of metabolism in the recombinant cells due to metabolic burden. Clustering generally reveals higher level information of metabolic reprogramming such as alteration of biosynthesis pathways, regulation of ribosomal proteins, amino acids and nucleotide synthesis. It also reveals the strategies used by the cells to cope up with stress and changes in transportation genes. Such higher level information about metabolic reprogramming is useful to decide steps for

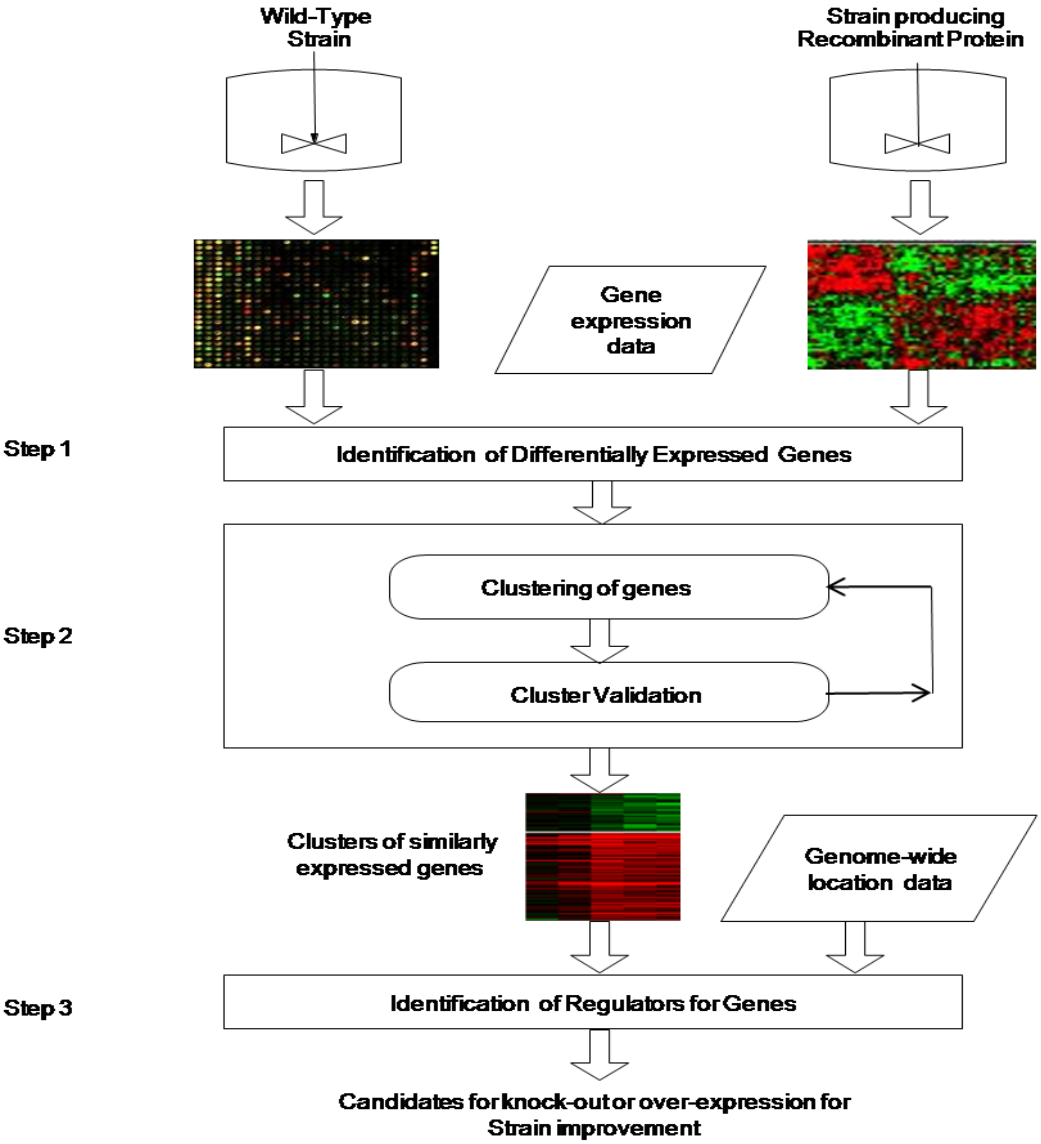

Fig. 3.1. The proposed data-driven methodology for identification of gene targets for strain improvement

strain improvement.

In the third step, other genomic datasets such as genome-wide location data is combined with gene expression data to reliably identify the Transcription Factors for the shortlisted genes. In case if genome-wide location data is not available, the known TF-gene interactions can be used in this step. Then, average correlation of cluster of

genes to different regulator genes (TFs) is used to assign TFs to cluster since similarly expressed genes often regulated by same TFs. A TF may regulate genes from multiple clusters. Also, TFs may regulate genes indirectly through other TFs. Hence, a high correlation of TF expression to gene cluster may not be sufficient to reliably assign TFs to clusters. To circumvent this problem, the number of genes in a cluster that are actually bound by a given TF is also considered while assigning TFs to a cluster. These key regulator genes are the potential candidates for further knock-out or over expression in order to improve the strain.

4. PCA BASED METHODOLOGY FOR IDENTIFYING DIFFERENTIALLY EXPRESSED GENES IN TIME-COURSE MICROARRAY DATA

4.1 Introduction

Microarray expression profiling is often carried out to identify genes whose expression change across biological conditions (Slonim, 2002). Several methods have been proposed in literature to identify differentially expressed genes in static experiments where snapshots of gene expression levels are measured in two different cell populations (Pan, 2002; Troyanskaya *et al.*, 2002). These methods are not directly applicable for time-course experiments where differential expression has to be calculated globally in the temporal space and not just between corresponding time points (Storey *et al.*, 2005).

Recently, several methods have been proposed to identify the differentially expressed genes in time-course data. Bar-Joseph *et al.* (2003a) proposed a method that represents expression profiles as continuous curves and then uses a global difference between the curves to identify differentially expressed genes. In their approach, clustering of genes is used as a preprocessing step; although simple, this makes the method computationally expensive for large datasets. Storey *et al.* (2005) proposed a method that measures the improvement in goodness-of-fit when two separate curves are used to fit the data from two conditions compared to single curve is used to fit the data from both conditions together. If the improvement in goodness-of-fit is significant then that par-

ticular gene is considered as differentially expressed. Their approach treats all genes as equal irrespective of their expressions levels in the experiments. This leads to the spurious identification of genes with low expression in both conditions as differentially expressed genes. Conesa *et al.* (2006) proposed a regression-based approach that models the expression profile of each gene with time as regressor and tests the hypothesis on the equality of regression coefficients. A similar method is proposed by Vinciotti *et al.* (2006) where the expression profiles are fitted using cubic polynomials and tested for similarity of coefficients. Modeling individual genes is generally not recommended due to noise in the microarray data (Bar-Joseph *et al.*, 2003c). Cheng *et al.* (2006) proposed an approach that represents the time-course data from both conditions as two different gene relationship networks where each node is a gene and each edge links the two similarly expressed genes. Differentially expressed genes are identified by comparing the neighborhood, genes that have very similar and very dissimilar expression profiles, of each gene i in both networks. Genes with dramatic change in neighborhood are deemed as differentially expressed. Since the actual expression of gene is not directly compared in both conditions, genes similarly expressed in both conditions can be declared as differentially expressed if their neighbors are changed. Reverter *et al.* (2006) proposed a method that identifies genes that are simultaneously differentially expressed and differentially connected. However, they quantify the difference in expression of a gene as the sum of differences in individual time-points which may not capture systematic variations. Methods based on ANOVA (Park *et al.*, 2003) and ANCOVA (Tabibiazar *et al.*, 2005) models have also been proposed specifically for replicated time-course data.

Each one of the available methods for identifying differentially expressed genes in time-course data have particular drawbacks. They also do not consider natural dependencies among different time-points. The noise in the data is also not explicitly considered in these methods. Here, a statistical method is proposed for identifying differentially expressed genes in time-course data. The proposed method uses PCA to consider the correlation among different time-points and reveal fundamental patterns in the data. The scores of genes on these fundamental patterns are used to identify the differentially expressed genes. Noise is discounted by considering only the most significant PCs (patterns) in the analysis.

Let time-course gene expression be measured at two different biological conditions, C_1 and C_2. The proposed method relies on PCA to model the expression data from C_1. Noise is removed from the model by using only the dominant components. When the expression data from C_2 is projected on this PCA model, differences in the gene expression program can be identified. Genes whose expressions do not change between the two conditions will have similar scores, while scores will be different for differentially expressed genes. A statistical test is used to find the significance of the difference in scores and reliably identify differentially expressed genes and their p-value.

There are several advantages of using PCA for finding differentially expressed genes: (1) The score of a gene on a PC is the correlation between the gene and the PC. Comparing the scores is equivalent to comparing the similarity of temporal expression profiles. So the proposed approach uses the systematic differences in expression to identify differentially expressed genes, (2) Since only the dominant PCs are used for

analysis, the effect of noise in the data is alleviated. This leads to meaningful comparison of expression profiles across conditions and identifies significant differentially expressed genes. (3) PCs are the fundamental patterns in the data. They can be interpreted and hence provides more information about the differences in expression of genes (Holter *et al.*, 2000; Raychaudhuri *et al.*, 2000; Alter *et al.*, 2000).

4.2 Methods

4.2.1 Modeling C_1 expression data using PCA

Let $X^{(1)}_{n\times t}$ be the expression data containing n genes measured at t time-points. The superscript refers to the biological condition at which the expression data is collected. Each element x_{ij} represents the expression level of i^{th} gene measured at the j^{th} time-point. PCA decomposes the expression matrix $X^{(1)}$ as the sum of outer product of two vectors $\mathbf{z_i}$ and $\mathbf{p_i}$ plus a residual matrix $\mathbf{E}$ (Jackson, 1991)

$$X^{(1)}_{n\times t} = \mathbf{z_1^{(1)}p_1^T} + \mathbf{z_2^{(1)}p_2^T} + ... + \mathbf{z_k^{(1)}p_k^T} + \mathbf{E} \tag{4.1}$$

where $\mathbf{z_i^{(1)}}$ vectors, known as scores, are of size $n \times 1$, the $\mathbf{p_i}$ vectors are called loadings and their size is $t \times 1$. Here $k \leq \min(n, t)$.

PCA relies on the eigenvalue decomposition of the covariance matrix of $X^{(1)}$, given by:

$$S = \frac{X^{(1)^T} X^{(1)}}{n-1} \tag{4.2}$$

provided $X^{(1)}$ is mean-centered. The $\mathbf{p_i}$ vectors are the eigenvectors of the covariance matrix of data and represent the Principal Components (directions) of variation in the data, *i.e*

$$S\mathbf{p_i} = \lambda_\mathbf{i}\mathbf{p_i} \tag{4.3}$$

where λ_i is the eigenvalue associated with the eigenvector $\mathbf{p_i}$. The eigenvalue λ_i is the variance ins direction represented by $\mathbf{p_i}$. The Principal Components $\mathbf{p_i}$ form an orthogonal set. Hence the score vector for each $\mathbf{p_i}$ is given by:

$$\mathbf{z_i^{(1)}} = X^{(1)}\mathbf{p_i} \tag{4.4}$$

The Principal Components (PCs) are similar to the eigengenes of Alter *et al.* (2000) that represent the fundamental patterns of the gene expression program that contribute to the expression of genes all over the genome. In this model (Equation 4.1), the expression profile of each gene is represented as a linear combination of the PCs with associated gene-specific scores. So the expression dataset can be reconstructed if all the pairs of score and loading vectors are retained. The $(\mathbf{z_i^{(1)}}, \mathbf{p_i})$ pairs are arranged in descending order of λ_i. So, the first few components associated with larger variance represent the systematic variation in data whereas components with lower variance essentially contain noise due to uncontrolled experimental and instrumental variations. The filtering of the insignificant components removes noise from the expression data and enables a meaningful comparison of the expression profiles.

The identification of significant components translates to selecting a value for k, the number of PCs to be retained. The simplest approach is to find the number of PCs that

can capture at least a predefined amount (say 95%) of the original variance in the data. Another technique, scree test, plots the eigenvalues in non-increasing order to find the 'knee' between dominant and insignificant PCs. The number of PCs can also be found by significance tests Bartlett (1950). Here, the cross-validation procedure proposed by Wise and Ricker (1991) is used for selecting number of PCs. In this procedure, the dataset is divided into a predefined number of equal sized segments. PCA model is developed on all but one of the segments. The developed PCA model is used to reconstruct the un-modeled data. The error in reconstruction, the Root-Mean Square Error of Cross-Validation (RMSECV), is plotted as function of number of PCs and the number of PCs, k, is selected with minimum RMSECV.

4.2.2 Projection of expression data on PCA model

Through the procedure described above, a PCA model of C_1 expression is generated where the expression profile of each gene over time, x_i, is represented as a combination of PCs. The expression data from condition C_2 can then be compared for statistically significant differences from this PCA model. Let the expression data from C_2 be denoted as $X^{(2)}_{n\times t}$ where the same genes are measured at the same time points in a different biological condition C_2. If there are differences in the time points between C_1 and C_2, it can be addressed by resampling either/both C_1 and C_2. Projection of $X^{(2)}$ on to the PCA model gives the corresponding scores vectors

$$\mathbf{z_i^{(2)}} = X^{(2)}\mathbf{p_i}, i \in [1, k] \tag{4.5}$$

Genes whose expression is not significantly altered in C_2 will have approximately the same scores, *i.e.* $\mathbf{z_i^{(1)}} \approx \mathbf{z_i^{(2)}}$, while differentially expressed genes will have significant differences in their $\mathbf{z_i}$s . A statistical test is used to find the significance of the difference in scores and thus identify differentially expressed genes.

4.2.3 Calculation of significance of differential expression

Let Z^{Δ} be the difference between Z^1 and Z^2 where the i^{th} row of Z^{Δ} is the difference in the scores of gene g_i

$$\mathbf{z_i}^{\mathbf{\Delta}} = \mathbf{z_i}^{(1)} - \mathbf{z_i}^{(2)} \tag{4.6}$$

We test the hypothesis that the differences in scores is due to noise. Therefore, the null and alternative hypotheses are:

H_0 = Difference in the scores of gene is due to noise

H_1 = Difference in scores of gene is not due to noise

This hypothesis is tested based on the following insight. When we depict each gene g_i on the scores plot, genes with small $\mathbf{z_i}^{\mathbf{\Delta}}$ will form a k-dimensional cloud around the origin while genes that are differentially expressed will be away from the origin. The distance of $\mathbf{z_i}^{\mathbf{\Delta}}$ from the origin measured using a suitable metric and considering the null distribution, reveals the significance of the difference in the scores, and thus that of differential expression of that gene.

The Mahalanobis distance is a common metric used with PCA and is given by:

$$MD_i^2 = (Z_i^{\Delta} - \overline{Z})\Sigma^{-1}(Z_i^{\Delta} - \overline{Z})^T \tag{4.7}$$

where $\overline{Z}$ is the centroid of Z^{Δ} and Σ is the covariance matrix of Z^{Δ}. We use the Mahalanobis distance to find the distance between each point to the centroid and use it as evidence for the differential expression. Mahalanobis distance is the most widely used distance metric with PCA analysis (Jackson, 1991). The larger the distance, the more evidence there is to conclude that a particular gene is differentially expressed and hence the null hypothesis can be rejected. When the difference in scores follows a multidimensional normal distribution, the Mahalanobis distance follows a χ^2 distribution with k degrees of freedom. The p-value that the differential expression occurred due to noise is then given by the cumulative distribution function:

$$P_i = 1 - \int_0^{MD^2} \frac{t^{(k-2)/2} e^{(-t)/(2)}}{2^{k/2}\Gamma(k/2)} \tag{4.8}$$

where $\Gamma(\cdot)$ is a Gamma function.

4.3 Results

We evaluate the proposed method using two case studies. The first case study involves genome-wide study of differences in the heat-shock response of wild-type mouse and strain lacking Heat-Shock Transcription Factor 1 (HSF1). The second case study concerns the Yeast cell-cycle response between the wild-type and a mutant lacking forehead proteins (Fkh1 and Fkh2). We compare the results from these studies with results from other recent approaches.

4.3.1 Case Study 1: Mouse time-course dataset

HSF1 is the primary regulator for many heat-shock proteins in mammalian cells. To characterize its role, Trinklein *et al.* (2004) measured the transcription levels and also assayed the binding of HSF1 on human promoters. From this study, Trinklein *et al.* (2004) hypothesized that the induction of several heat response genes is independent of HSF1. To test the hypothesis, Trinklein *et al.* (2004) measured the expression levels of 9468 mouse genes using cDNA microarrays. Expression levels of genes are measured at 0, 0.5, 1, 2, 3, 4, 6, and 8 h after the heat-shock in both wild-type and mouse lacking HSF1. Trinklein *et al.* (2004) analyzed the transcriptional response of different gene groups: (A) mouse genes homologues of human genes that are bound by HSF1 and induced, (B) homologues that were bound by HSF1 but not induced, (C) homologues that were induced but not bound by HSF1, (D) genes induced by heat in wild-type but not in mutant, (E) genes induced in mutant mouse, (F) genes induced similarly in both wild-type and mutant. Ideally, genes belonging to groups A, D and E should be identified as differentially expressed between wild-type and HSF1 mutant mouse and genes belonging to groups C and F as similarly expressed.

Modeling the wild-type mouse time-course data

We modeled the time-course expression data from the wild-type mouse using PCA. The number of PCs, k, to be retained in the model was found using cross-validation. The RMSECV takes the minimum value at $k = 2$ (Figure 4.1). In PCA, the extracted PCs are arranged in the descending order of data variance they capture. The first two PCs capture 42.12% and 24.75% of the total variance, respectively. The third PC cap-

tures only 9% of the variance and the remaining PCs smaller amounts. The expression profiles of PCs are shown in Figure 4.2. Though several PCs modeling systematic changes in expression data, the variance captured by PCs 3 to 8 is small compared to variance captured by first two PCs. So the first two PCs are used to model this dataset.

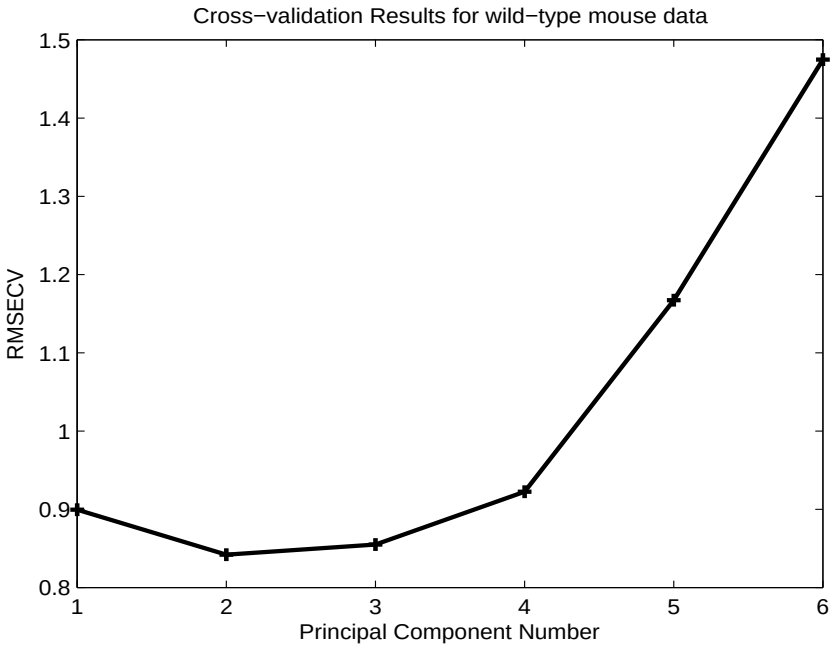

Fig. 4.1. Cross-validation results for the wild-type mouse time-course data. The RMSECV has the minimum value at number of PCs 2. So two PCs are used to model this dataset.

In order to validate the PCA model, we analyzed the expression profiles of these two PCs shown in Figure 4.3. In wild-type mouse, the heat-shock activates several heat inducible genes. The first PC has an upward trend while the second PC shows an upward trend initially after the heat-shock and a downward trend afterward. Genes whose scores are positive on first PC has upward trend after heat shock. Some of these genes include known heat inducible genes hsp60, hsp70, hsp86, etc. This indicates that

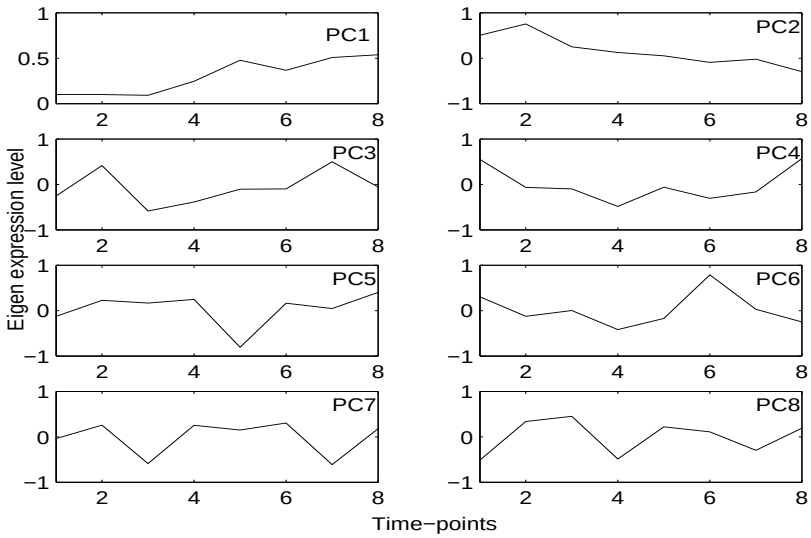

Fig. 4.2. Expression profiles of PCs extracted in mouse dataset. Though several PCs modeling systematic changes in expression data, the variance captured by PCs 3 to 8 is small compared to variance captured by first two PCs.

the first PC corresponds to activation of the genes due to heat-shock. The second PC represents the dynamic changes in the expression of genes over time.

Identifying differentially expressed genes

The time-course data from the mouse lacking HSF1 is projected on the developed PCA model and the scores of these genes on the two PCs extracted. The differences in their scores are used to calculate the p-values for the genes. The histogram of the p-values for all the genes is shown in Figure 4.4. There are 288 genes in the p-value range $0-0.01$. The frequency drops to 70 in the range $0.01-0.02$ (inset in Figure 4.4) and the p-values for the rest of the genes are distributed more or less uniformly. So we selected a p-value threshold of 0.01 for this dataset.

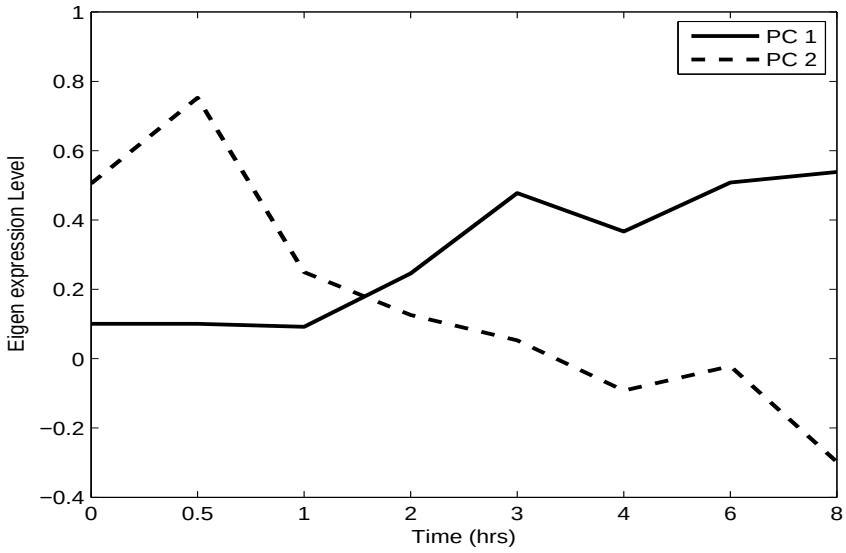

Fig. 4.3. Expression profiles of the 2 PCs used to model wild-type mouse dataset. First PC shows the pattern related to activation of genes. The second PC has the increased expression in the first time-points and then decreased. It corresponds to the dynamic changes in genes expression due to heat-shock.

The proposed method identifies 288 genes as differentially expressed at this p-value threshold. The differences in the scores on two PCs are shown in Figure 4.5. The differentially expressed genes (marked as '*') are far away from the majority of the genes. This confirms that the proposed hypothesis test identifies the genes with large difference in scores. Since the HSF1 gene is knocked-out in the experiment, we expect that the targets of HSF1 gene will be differentially expressed in the mutant mouse. On the other hand, genes related to metabolism and signaling processes are expected to be similarly expressed in the wild-type and mutant mice. The differentially genes identified by the proposed method include genes previously reported as the targets of the HSF1 such as hsp60, hsp70, hspa8 (McMillan *et al.*, 1998). In contrast, several metabolic and signal transduction genes including methylene tetrahydrofolate dehydrogenase, carbon

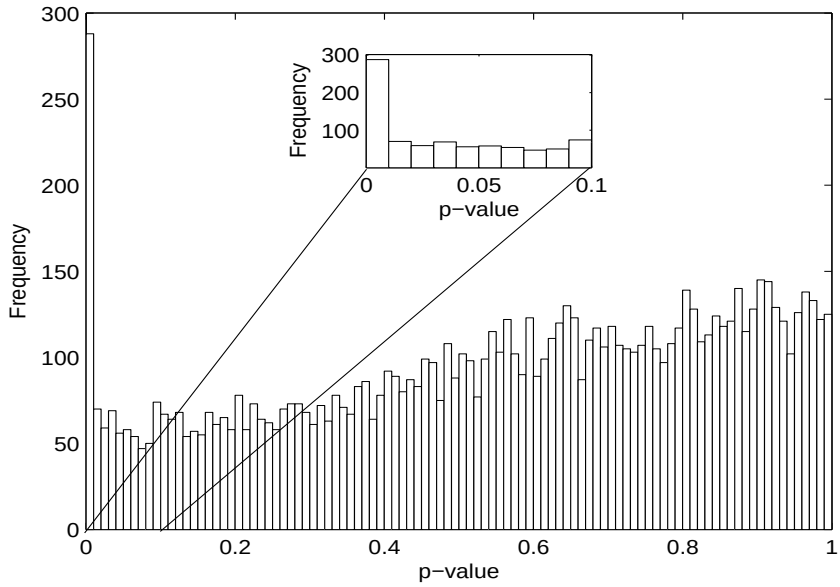

Fig. 4.4. The distribution of p-values of the genes in mouse dataset. There are 288 genes in the p-value range 0-0.01. After that the distribution if more or less uniform. The p-value threshold selected for this dataset is 0.01.

catabolite repressor, Protein kinase C alpha binding protein, and MAD homologue 7 are not identified as differentially expressed. The p-values for these genes are between 0.018-0.9989. This clearly shows that the proposed method is able to identify differentially expressed genes with biological implications.

Our method identifies four (out of 9), group A mouse genes homologues of human genes that are both bound by HSF1 and induced in wild-type mouse. These are Hsp105, Dnajb1, hsp84-1, and Cacybp and the corresponding p-values are 1.0×10^{-15}, 7.014×10^{-8}, 3.0614×10^{-4}, and 4.7355×10^{-4}. On the other hand, 13 (out of 15) group C mouse genes homologue to human genes that are induced in wild-type but not bound by HSF1 are not identified as differentially expressed genes. The p-values for these

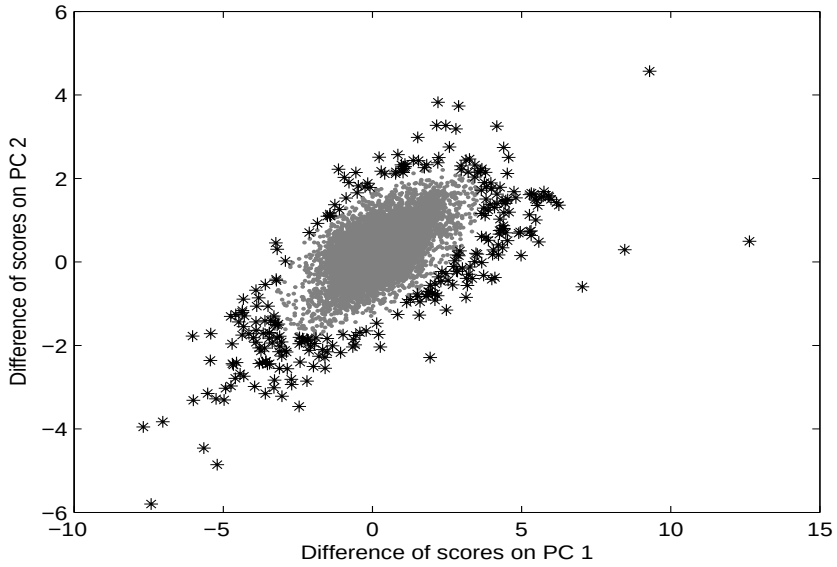

Fig. 4.5. Difference of scores of mouse genes on first two PCs. The differentially expressed genes identified by the proposed method are marked '*'.

genes are in the range of 0.035-0.927. These results support the hypothesis that HSF1 does not regulate all the heat induced genes.

Comparison of results with previous study

Trinklein *et al.* (2004) reported 167 genes differentially expressed in the experiment (groups D and E). Our approach identified 78 of the genes out of these 167 genes identified by Trinklein *et al.* (2004). Most of the remaining genes identified by Trinklein *et al.* (2004) have <2-fold change at all the time-points in both wild-type and the mutant mouse. Trinklein *et al.* (2004) used the heatmaps of the clusters to identify differentially expressed genes. In heatmaps, small positive and small negative values are showed in different colors and hence lead to the identification of genes with small changes as differentially expressed genes. The proposed approach also identified 210

novel genes as differentially expressed. We clustered these genes using hierarchical clustering (Figure 4.6).

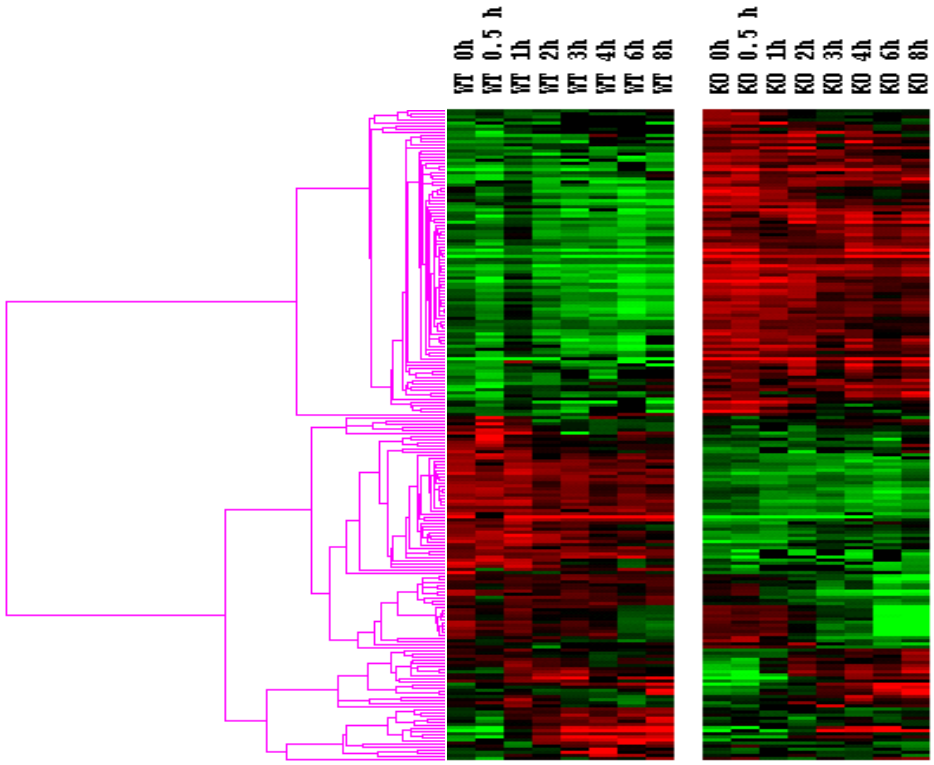

Fig. 4.6. Heatmap of the novel genes identified by the proposed method in mouse time-course dataset. Up-regulation of gene is indicated by red color and down-regulated genes are represented by green color. From this figure, it is clear that these novel genes are differently expressed between wild-type and mouse lacking HSF1 gene.

The figure shows that the novel genes are differentially expressed between the wild-type and mutant mouse. Trinklein *et al.* (2004) identified the genes that have completely up- or down-regulated between the wild-type and mutant mice. This can be seen in Figure 4.7 where the genes identified by Trinklein *et al.* (2004) are spanned only in the direction of first PC that represents activation of genes after heat-shock. Genes on the positive side of the plane are up-regulated in wild-type and down-regulated in mutant

mouse. Genes on the negative side of the plane are down-regulated in wild-type and up-regulated in mutant mouse. This indicates that Trinklein *et al.* (2004) identified only the genes that are completely up- or down-regulated. The proposed approach identifies all the genes with differential expression between the two mice.

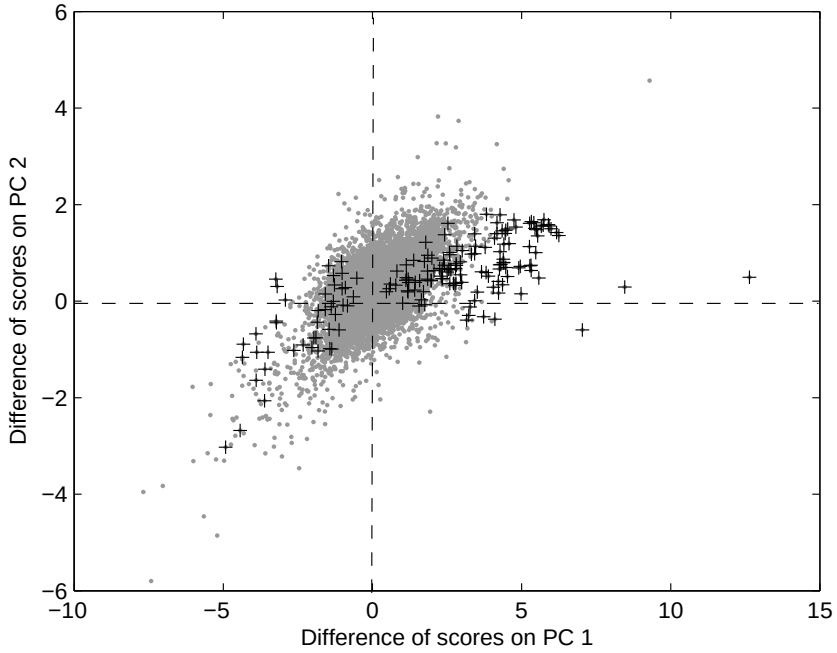

Fig. 4.7. Difference of scores of mouse genes on first two PCs. The differentially expressed genes identified by Trinklein *et al.* (2004) are marked '+'.

4.3.2 Case Study 2: Yeast cell-cycle dataset

For the second case study, we use the Yeast cell-cycle dataset where the expression levels of genes are measured over two cell-cycles in a wild type and Fkh1, Fkh2 double mutant strain. Spellman *et al.* (1998) monitored the expression levels of almost all genes during two cell-cycles. Eighteen samples were taken following the α factor release with a sample period of 7 mins. They identified 800 cell-cycle regulated genes

using periodic algorithms. Zhu *et al.* (2000) monitored the expression levels of Yeast genes in a mutant strain that lacks two forkhead transcription factors Fkh1 and Fkh2. They measured expression levels at 13 time-points, the first twelve at 15 min intervals from time 0 till 165 mins, and the last at 210 mins. Out of the 800 cell-cycle genes reported by Spellman *et al.* (1998) in the Wild-Type (WT) strain, expression data is available for 746 genes in the Knock-Out (KO) experiment. So we use the expression data for these 746 genes from both strains to evaluate the proposed method.

Since the number of samples and the time of samples are different in WT and KO experiments, we use dynamic time warping (Sakoe and Chiba, 1978) to align the expression profiles by warping their time scales. Particularly, we use asymmetric time warping algorithm to map the time axis of the KO genes signals to the WT ones. The expression profiles of both the WT and KO genes are fitted to cubic splines and resampled at each minute. These supersets are aligned using asymmetric DTW. After alignment, the resampled expression values for the KO are obtained at the time points corresponding to the original WT samples (0 to 119 mins with a period of 7 mins). The aligned datasets for both the WT and KO strains thus contain expression of 746 genes at 18 time points.

Modeling the wild-type time-course data

We modeled the expression time-course data from the wild-type Yeast strain using PCA. The RMSECV has local minima at k is 4, 8 and 11 (Figure 4.8). The first 4 PCs capture approximately 80% of the variance in the data. Considering the noise in microarray data, we use only the first 4 PCs. The expression profiles of the four

eigengenes (PCs) are shown in Figure 4.9. These PCs correspond to the different fundamental patterns in the WT cell-cycle data. Genes from different phases are found to be highly correlated with these patterns. For example, genes with high scores on the PC 1 such as Clb2, Clb1, Ace2 and Cdc5 are mainly from G2 and M phases. Similarly, genes from G1 and S phases have higher scores on PC 2 and the PC 3 maps to the M/G1 and G2 phases. PC 4 contributes to genes from different phases.

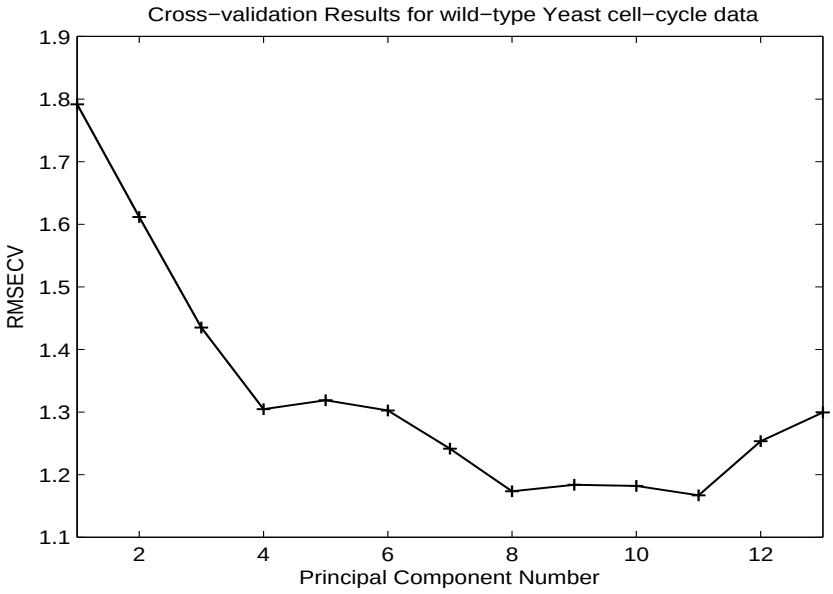

Fig. 4.8. Cross-validation results for wild-type yeast cell-cycle dataset. The RMSECV takes local minima at number of PCs 4, 8 and 11. The first 4 PCs captured almost 80% of variance in the data. The first 4 PCs are used to model this dataset.

For comparison, expression profiles of all PCs are shown in Figure 4.10. The first four PCs have systematic changes in expression and the expression profile of rest of PCs is random depicting noise. So modeling this dataset with 4 PCs good.

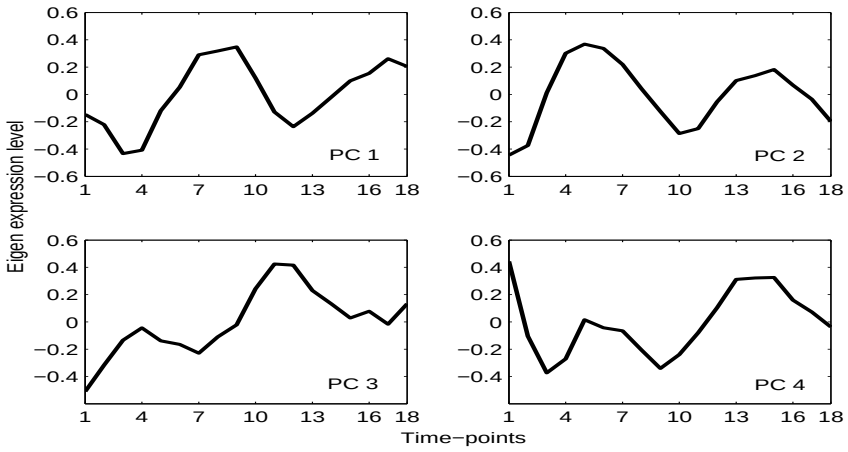

Fig. 4.9. Principal Components extracted from the wild-type Yeast cell-cycle dataset. The four PCs extracted from the wild-type Yeast cell-cycle dataset have distinct patterns and map to different phases of the cell-cycle.

Identifying differentially expressed genes

When the re-sampled KO (C_2) gene-expressions were projected to the PCA model, the proposed method identified 72 genes as differentially expressed at the p-value threshold of 0.05 since the total number of genes are small. We identified several genes expressed at high levels in WT strain but showing little or no expression in KO strain. For example, 40 genes had 2-fold change in at least at one time-point in the WT strain that lost their expression in the KO strain and showed less than 2-fold change in all the time-points. The proposed method also identified 4 genes that have less than 2-fold change in WT strain but having 2-fold change at one time-point (2 genes) and 2 time-points (2 genes) in the KO strain. We identified one gene that has less than 2-fold change in both WT and KO strain as differentially expressed. All the remaining genes showed high expression levels in both the WT and KO strains but differed in their ex-

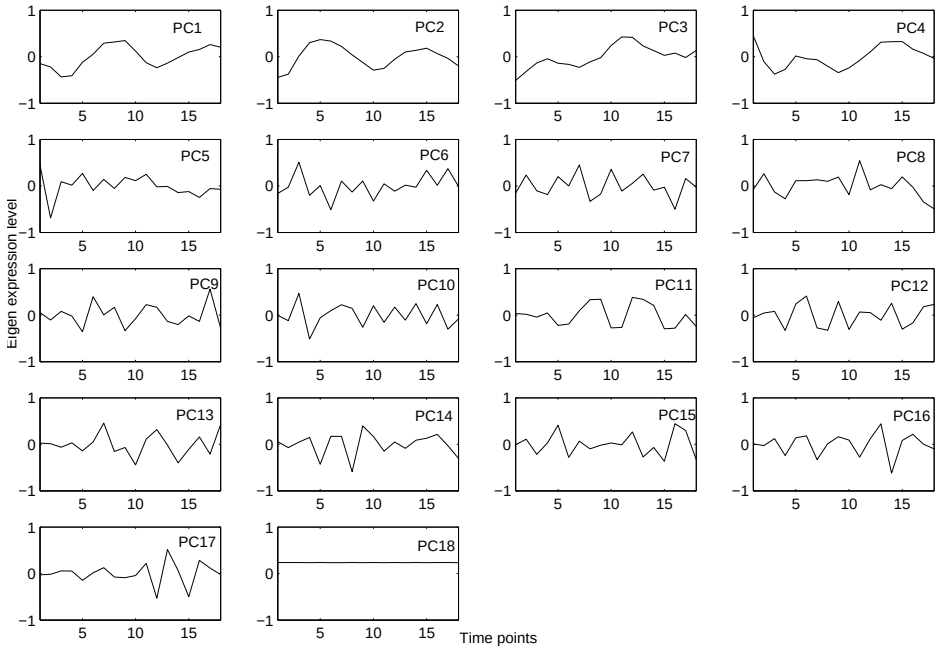

Fig. 4.10. Expression profiles of Principal Components (PCs) extracted in Yeast cell-cycle dataset. PCs 1-4 have systematic changes in expression over time where as the expression profile of rest of PCs is nearly random. This indicates that modeling this dataset with 4 PCs is good.

pression profiles.

Zhu *et al.* (2000) analyzed the heatmaps of clusters of co-expressed cell-cycle genes and reported that genes from CLB2 and SIC1 clusters are differentially expressed in the mutant strain. The proposed method identifies several genes from CLB2 and SIC1 clusters. We identified 11 genes (out of 31) from CLB2 cluster. The expression profiles of four of these genes in WT and KO are shown in Figure 4.11. These genes show a significant difference in their expression between the WT and KO strains - oscillatory behavior (with $>$ 2-fold change) in the WT strain and almost no expression in KO

strain. Some of the remaining genes in this cluster have flat expression profiles in the KO as well as the WT (Figure 4.12). The genes identified by the proposed method are the most significantly differentially expressed genes in CLB2 cluster. In the SIC1 cluster, we identified 16 (out of 26) genes. The expression profiles of some of these genes in WT and KO are shown in Figure 4.13. From this figure, it is clear that the genes identified are differentially expressed. The remaining 10 genes showed a little expression in both the WT and KO (Figure 4.14). The benefit of the proposed method is the quantitative comparison of the expression profiles which enables the identification of significantly differentially expressed genes and minimizes subjective errors.

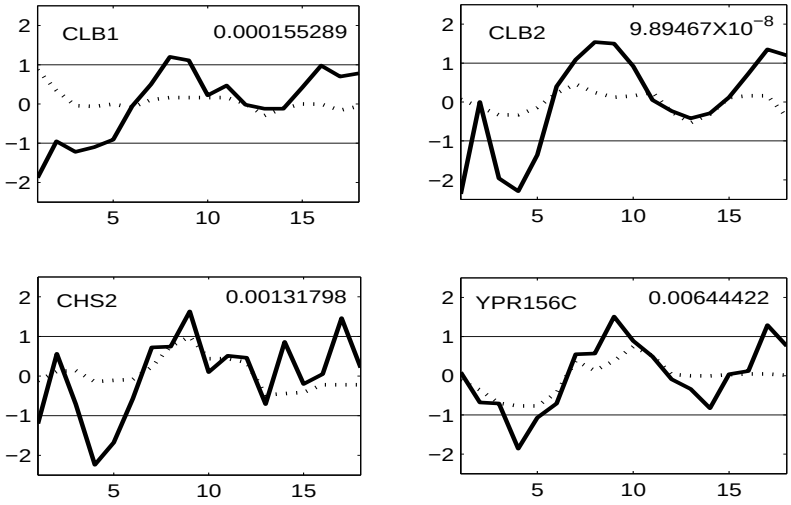

Fig. 4.11. Expression profiles of four genes identified by the proposed method in the CLB2 cluster. The solid line represents the expression of gene in the WT and the dotted line represents the expression of gene in the KO strain. Gene names and the p-values are shown for all genes. The WT genes show an oscillatory behavior while the expression in KO is significantly changed.

We validate the results at different levels. First, we compare the genes identified by the proposed method with the results from other approaches for identifying differen-

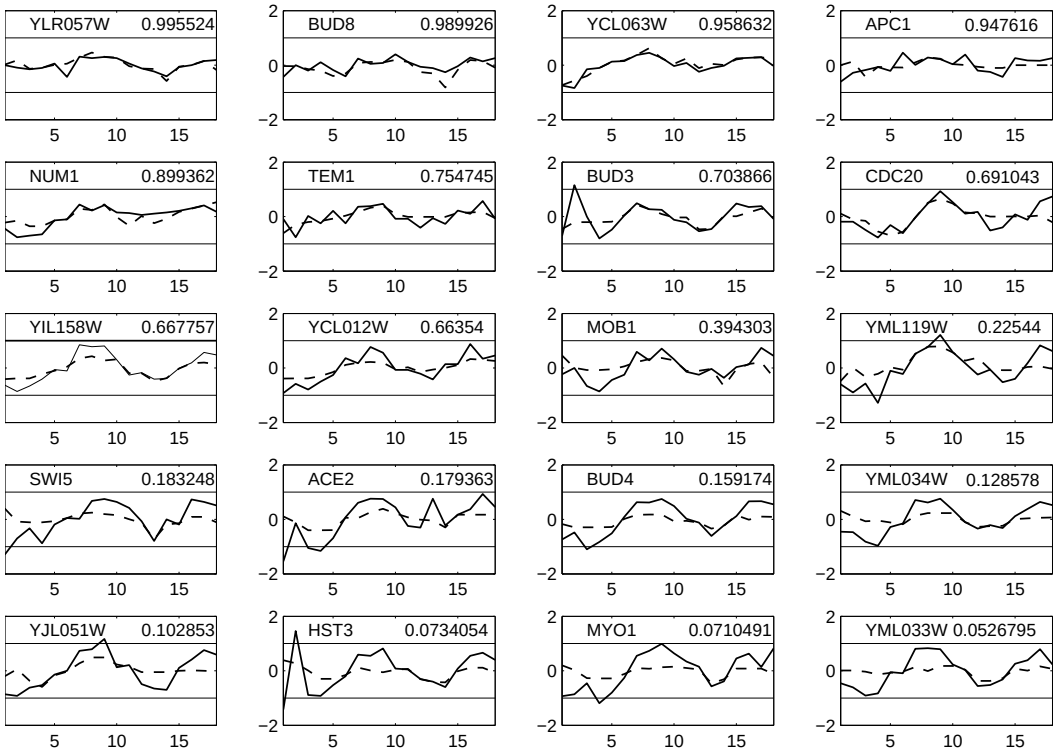

Fig. 4.12. Expression profiles of genes from CLB2 cluster that are not identified as differentially expressed by the proposed method. Solid line represents the expression profile in WT strain and the dash line represents the expression profile in KO strain. Horizontal lines correspond to 2-fold change. Most (15 of 20) have less than 2-fold change in both WT and KO strains. Increasing the p-value threshold from 0.05 to 0.10 will lead to identification of 3 more genes as differentially expressed.

tially expressed genes. The novel genes identified by our method are evaluated using the Genome-wide location data from Simon *et al.* (2001) who studied genome-wide transcription factor (TF)-DNA interactions for nine cell-cycle TFs including Fkh1, Fkh2, Ace2 and Swi5. Finally, differential expression of genes is also confirmed by directly comparing the actual expression profiles.

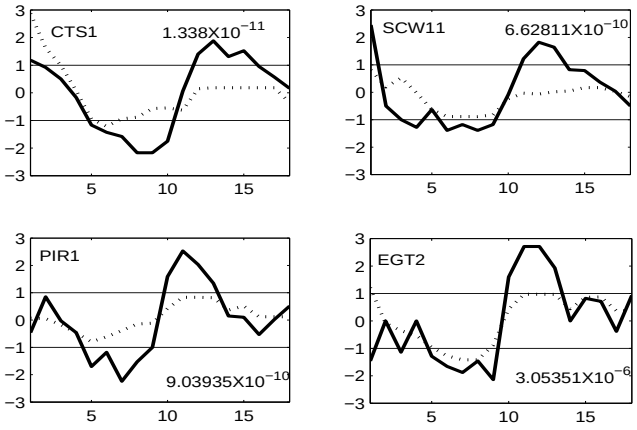

Fig. 4.13. Expression profiles of four genes identified by the proposed method in SIC1 cluster. The solid line represents the expression of gene in the WT and the dotted line represents the expression of gene in the KO strain. Gene names and the p-values are shown for all genes. There is a considerable change in the expression of SIC1 genes between WT and KO strain.

Comparison with results from other methods

We compare our results with the results from the different approaches proposed for identifying differentially expressed genes in time-course microarray datasets. Bar-Joseph *et al.* (2003a) used the same datasets and reported 56 genes as differentially expressed. There is a significant overlap between the genes identified by our method and those reported by Bar-Joseph *et al.* (2003a). Our method identifies 44 of these 56. Changing the p-value threshold to 0.1 includes 5 more genes. We found all the genes identified by Bar-Joseph *et al.* (2003a) in CLB2 cluster. Additionally, our list includes Cdc5 and YPR156C from that cluster. Cdc5 is a pole-like kinase, possibly a substrate of Cdc28, which is found to be bound by Ndd1. Even though Ndd1 is not directly affected in this experiment, its binding is mediated by Fkh2 in G2/M (Koranda *et al.*, 2000). The second gene YPR156C is involved in polyamine transport. There are no

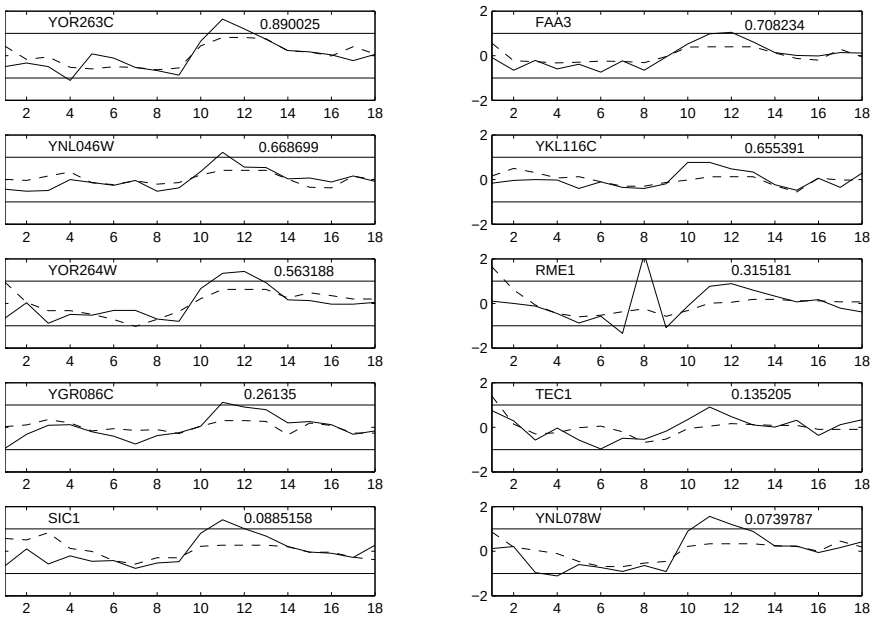

Fig. 4.14. Expression profiles of genes from SIC1 cluster that are not identified as differentially expressed by the proposed method. Solid line represents the expression profile in the WT strain and the dash line represents the expression profile in the KO strain. Horizontal lines correspond to the 2-fold change.

regulators found to be bound to this gene in TF-DNA interaction data. However, its expression is different between WT and KO. Similarly, most of the genes reported by Bar-Joseph *et al.* (2003a) from the SIC1 cluster have been identified by our method.

We used the EDGE software developed by Storey *et al.* (2005) to identify differentially expressed genes based on goodness-of-fit approach. Using natural cubic splines with basis of 4, their method identifies 73 genes as differentially expressed at the p-value threshold of 0.001. Only 30 (out of these 73) genes overlap with the genes identified by our method, and only 22 genes with those identified by Bar-Joseph *et al.* (2003a). Overall, 21 genes are identified by all the three methods, while 42 are novel genes identified only by Storey *et al.* (2005) approach. Most of these novel genes show

very little expression in both the WT and KO strain (Figure 4.15). Only 7 of the 42 novel genes are found to be bound by one or more of Fkh1, Fkh2, Ace2 and Swi5. The normalization procedure they use equally weighs highly expressed genes and genes with little expression. This is the probable reason for the misidentification of genes with little expression as being differentially expressed.

Recently, Cheng *et al.* (2006) used the cell-cycle dataset to evaluate their approach and identified 100 genes as differentially expressed, among which 41 genes are present in our dataset (we used 746 cell-cycle regulated genes). We identified 19 out of these 41 genes as differentially expressed. Additional 6 genes will be identified as differentially expressed if the p-value threshold is increased to 0.1. The expression profiles of the remaining 22 genes are showed in Figure 4.16. Several genes showed similar expression in both wild-type and the mutant strain. The approach proposed by Cheng *et al.* (2006) considers the change in neighborhood of a gene in two conditions. Since the actual expression profile of genes is not compared in different conditions, genes with similar expression profiles could also be detected as differentially expressed if their neighborhood genes are differentially expressed.

Validation of Novel genes

The proposed PCA based approach identified 28 novel genes that have previously not been identified. We find the TFs for the novel genes using Genome-wide location data from Simon *et al.* (2001) with a strict p-value threshold of 0.005 for TF-DNA binding (Table 4.1). The novel genes we identified are from all cell-cycle phases. It is known that cell-cycle is carried out by serial regulation of transcription factors (Simon

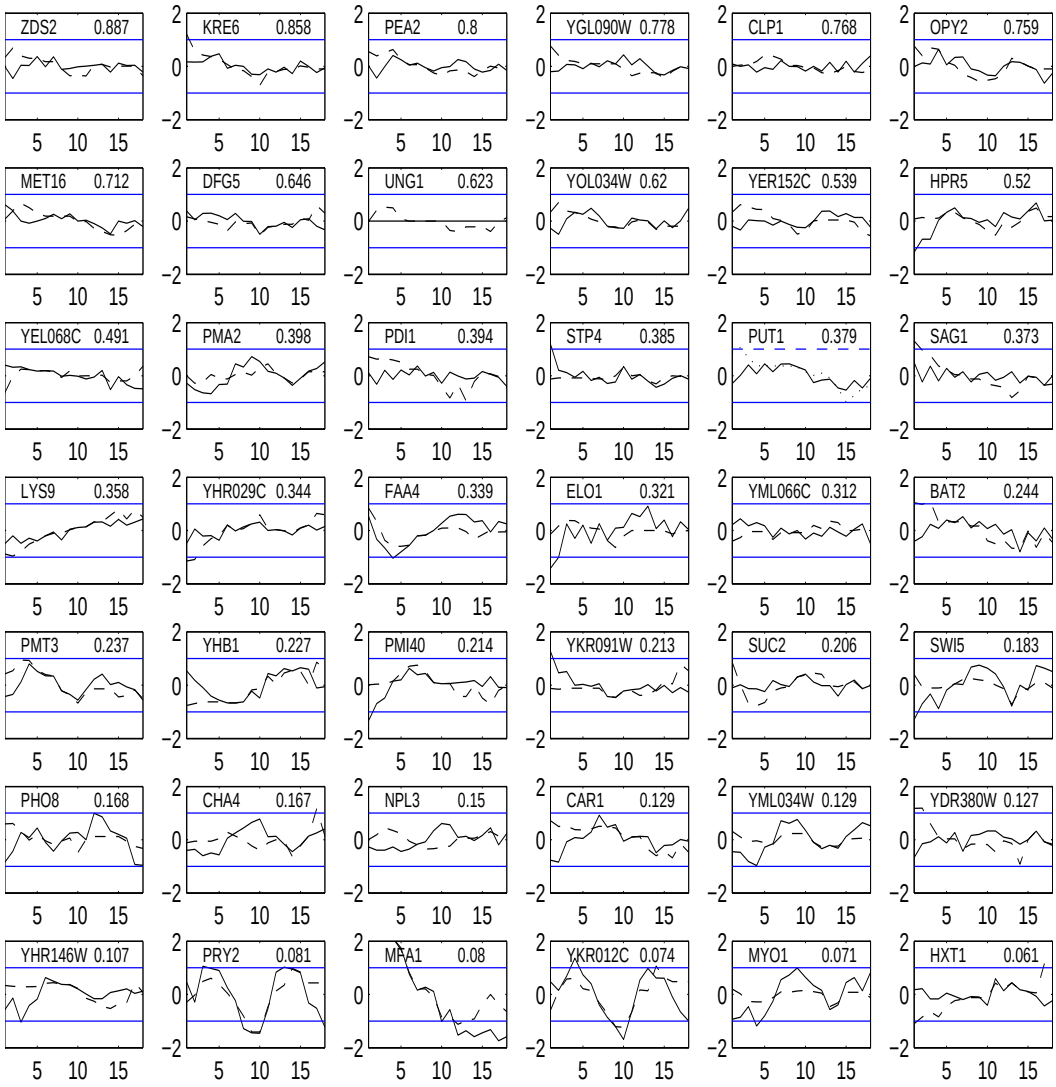

Fig. 4.15. Expression profiles of novel genes identified by EDGE method proposed by Storey *et al.* (2005). Solid line represents the expression profile in WT strain and the dash line represents the expression profile in KO strain. Horizontal lines correspond to the 2-fold change. Most of the genes have $<$ 2-fold change both in WT and KO strains and also has similar expression profiles.

et al., 2001). So it is expected that a change in the cell-cycle will affect the different phases. 13 genes (out of 28) are found to be bound by one or more of Fkh1, Fkh2, Ace2, and Swi5. Fhk2 is the predominant binding partner for Mcm1 and it also mediates the binding of Ndd1 (Koranda *et al.*, 2000). So genes regulated by Mcm1 or Ndd1 would

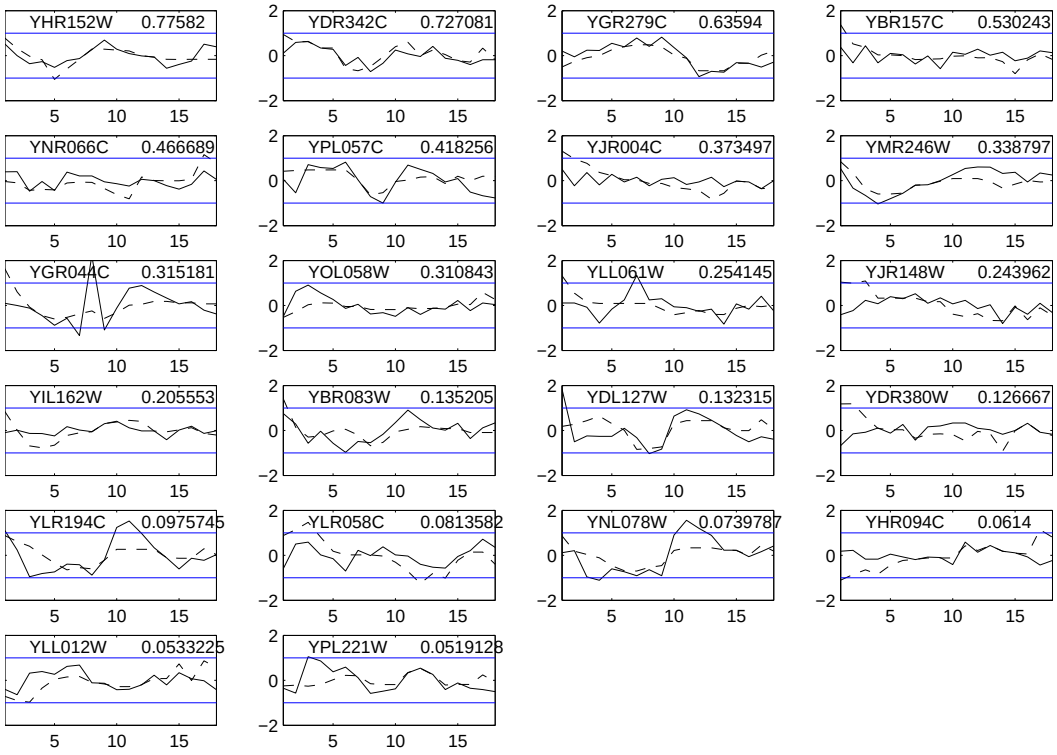

Fig. 4.16. Expression profiles of genes from identified as differentially expressed by Cheng *et al.* (2006) but not by the proposed method. Most of these genes have very little expression in both the WT and KO Yeast strains. Moreover, their expression profiles are similar in both strains. Increasing the p-value threshold from 0.05 to 0.10 will lead to identification of 6 more genes as differentially expressed by our method.

possibly change their expression in the mutant strain. The remaining genes are found to be bound by one or more of Swi4, Swi6, and Mbp1. Both Swi6 and Mbp1 have very little expression in WT and they were not identified as cell-cycle regulated genes by Spellman *et al.* (1998). So, the data we used includes only Swi4. The p-value for Swi4 is 0.06 which is very close to the threshold we used. It also shows a difference in expression between WT and KO. This differential expression of Swi4 is probably the reason for the differential expression of genes bound by it.

Table 4.1
Novel differentially expressed genes identified by the proposed method. Genes are grouped based on the phase of the cell-cycle where they show peak expression.

Gene	Phase	p-value	TFs
PCL9	M/G1	0.0495	Swi5
CHS1	M/G1	0.0098	Swi5
YDL117W	M/G1	0.0110	
YBR296C	M/G1	0.0104	
SST2	M/G1	0.0224	
AGA1	M/G1	0.0048	Mcm1, Mbp1,Swi4, Swi6
TSL1	G1	0.0274	Fkh1, Fkh2, Ndd1, Ace2, Swi5,Mbp1, Swi4, Swi6
CLB6	G1	0.0027	Fkh2, Mbp1, Swi4, Swi6
SVS1	G1	0.0001	Fkh1, Fkh2, Swi4, Swi6
POL30	G1	0.0268	
MCD4	G1	0.0257	
YOX1	G1	0.0085	Fkh2, Mbp1, Swi4, Swi6
CLN2	G1	0.0088	Swi6
YMR305C	G1	0.0187	Mcm1, Mbp1, Swi4, Swi6
HHT1	S	0.0403	Fkh2
HHO1	S	0.0162	Swi4, Swi6
YIL129C	G2	0.0021	Swi5
YMR215W	G2	0.0025	Fkh1,Fkh2, Mbp1, Swi6
CIK1	G2	0.0126	Fkh1, Fkh2
CDC5	M	0.0033	Ndd1
YPR156C	M	0.0064	
YPR157W	M	0.0219	
NCE2	M	0.0203	Fkh2, Ndd1, Swi4
FET3	M	0.0042	
YOR383C	M	0.0371	
YDL039C	M	0.0089	
CLN3	M	0.0108	Mcm1, Ace2, Swi5, Swi4, Swi6
MFA2	M	0.0216	Fkh1,Ndd1, Mcm1, Swi5

Understanding cell-cycle using novel genes

To understand how the cell-cycle is affected by the deletion of the two forkhead proteins Fkh1 and Fkh2, we constructed a heatmap of the cell-cycle regulated genes

using the Treeview software (Eisen *et al.*, 1998) (Figure 4.17).

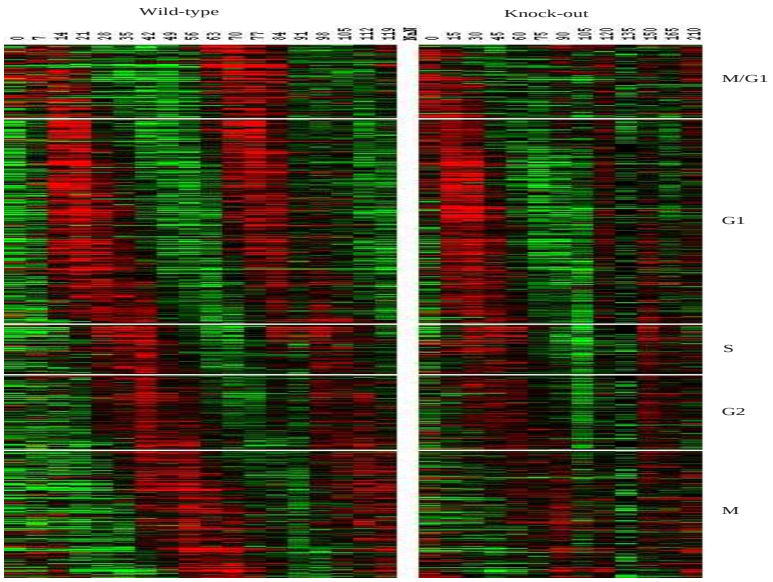

Fig. 4.17. Heatmap of cell-cycle expression data from WT and KO strains. Most of the genes from M/G1 and M phases differentially expressed in KO strain compared to WT strain. Genes from G1 phase retained their expression during first cell-cycle but differentially expressed in second cell-cycle. Most of the genes from G2 and S phase showed little or no change from their WT expression.

As expected, genes having peak expression in M (CLB2 genes) and M/G1 (SIC1 genes) phases of cell-cycle have lost their expression in the KO strain. Several G1 genes also showed a significant difference in their expression. One interesting aspect we observed in the heatmap is that in the KO strain most of the genes from G1 phase retained their expression in the first cell-cycle but not in the second cycle. However, the phenotype indicates that cells entered into second cell-cycle: mother and daughter cells budding synchronously (Zhu *et al.*, 2000). The novel genes we identified as differentially expressed partially explain this phenomenon.

To understand the cell-cycle regulation in Yeast, consider Figure 4.18, a simplified form of Simon *et al.* (2001) cell-cycle model. Two transcription factor complexes SBF (complex of Swi4 and Swi6) and MBF (complex of Mbp1 and Swi6) are major regulators of G1 phase genes. SBF requires Cln3-Cdc28 to change to active state by post-transcriptional action (Koch *et al.*, 1996).

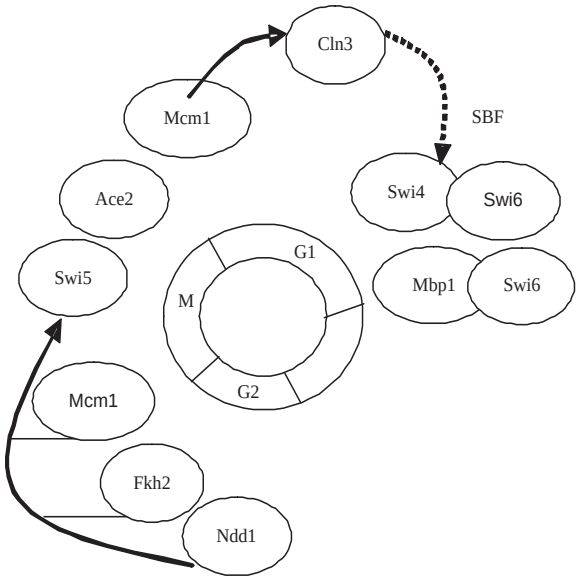

Fig. 4.18. Simple model of cell-cycle-regulation of Yeast. Transcription factors (TF) that regulate genes from different phases of cell-cycle are represented as ovals and placed near to the corresponding phases. Solid lines represent the regulatory interaction and dotted line represents the post transcriptional actions.

In contrast to the other approaches which identify only Cln1, we identified all three CLN genes (Cln1, Cln2 and Cln3) as differentially expressed. The expression profile of these three genes is shown in Figure 4.19. In the WT strain, all three show oscillatory behavior. Cln1 loses its oscillatory behavior in the KO strain and its expression is very low. Cln2 retains its oscillatory behavior but at a lower magnitude. Cln3 is not expressed in the KO strain. Cln3 is found to be bound by Mcm1, Ace2, Swi5, Swi4

and Swi6 (Table 4.1). So we hypothesize that for the KO strain, expression of Cln3 is affected, because of which SBF is in an inactive state. Consequently the expression of G1 phase genes during the second cell-cycle is altered. It has been reported that the other two CLN genes (Cln1 and Cln2) are regulated by SBF (Nasmyth and Dirick, 1991). The significant decrease in their expressions in the KO strain also lends evidence to the hypothesis that Cln3 affected SBF which in turn affected several G1 phase genes in the second cell-cycle (Figure 4.17). Further evidence is that CLB6, which is bound by SBF (Table 4.1), is also identified as differentially expressed.

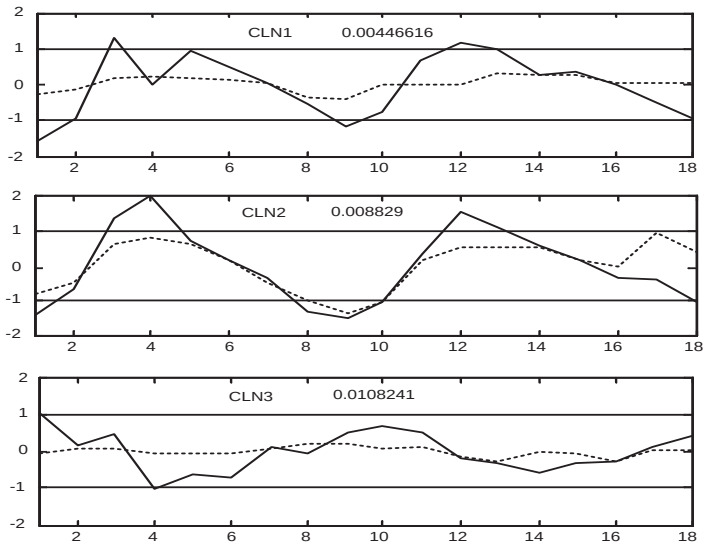

Fig. 4.19. Expression profile of three CLN genes in WT and KO strain. Cln1 lost its oscillatory behavior and almost flat in KO strain. Cln2 retains its oscillation but the magnitude of oscillation id diminished. Cln3 is not expressed in KO strain. Only Cln1 is reported previously as differentially expressed. We identified the remaining two CLN genes.

4.4 Discussion and Conclusions

In both the case studies, the Wild-Type (WT) dataset was modeled using PCA and the Knock-out data was projected on the model. When the KO data is used for model development and WT data projected on the model to identify differentially expression genes, the results are almost the same. For the Yeast cell-cycle Case study, 5 PCs are needed to model the Knock-out data (Figure 4.20). With this model, 89 genes were detected as differentially expressed at a p-value threshold of 0.05. There is a significant overlap between the two sets. Out of 72 genes from the WT model, 69 were also identified by the KO model. The median rank of these 72 genes is 37.5 which is very close to median rank of 36.5 if all these 72 were top in the list. This indicates that almost same genes are identified as differentially expressed in both scenarios and the proposed method is robust.

The proposed method uses Mahalanobis distance as the distance metric to find differentially expressed genes. Mahalanobis distance is the most widely used distance metric with PCA analysis. It weighs different directions (PCs) differently and the weights are inversely proportional to the variance in those directions. So, differences in expression in directions with large variance (higher noise) are given less credit when identifying differentially expressed genes. The advantage of having weightage for different directions is to separate the natural disturbances from real difference in expression of genes. However, giving weights to directions could lead to failing to detect some genes as differentially expressed if their co-expressed genes are highly variant. This is because the highly variant genes result higher variance in the PCs. This probably happens for the genes whose products work as Transcription Factors since their expression

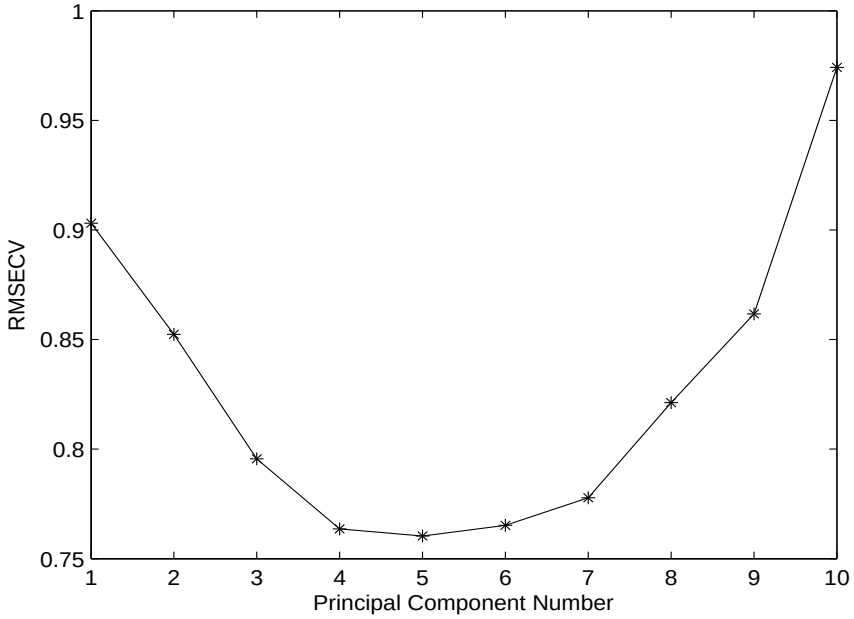

Fig. 4.20. Cross-validation results for Knock-out Yeast cell-cycle dataset. The RMSECV takes minimum value at 5 PCs. The first 5 Principal components (PCs) captured almost 87% of the variance in the data and are used to model this dataset.

may not be high. For example, the p-value for gene ACE2 in Yeast cell-cycle case study is 0.1794. ACE2 is a Transcription factor that activates expression of early-G1 genes. The expression levels of ACE2 is lower compared to its other co-expressed genes such as ALK1, CLB1, and IQG1 etc. which are identified as differentially expressed genes. This problem is with all computational methods as they depend on quantitative analysis. ACE2 is also not reported as differentially expressed by Bar-Joseph *et al.* (2003a). It is better to treat known Transcription factors separately or use suitable normalizing techniques in processing step.

Another important issue is the estimation of covariance matrix for the Mahalanobis distance calculation. Estimations of Covariance is prone to outliers in the data. Dif-

ferentially expressed genes are outliers in the PCA scores data. Hence, the covariance matrix is affected by these outliers. Methods are available for estimation of robust covariance matrix which is not effected by outliers in the data (Rousseeuw and Leroy, 1987). These methods use re-sampling approach and identify minimum volume ellipsoid that capture predefined (say 75%) of the data points in multidimensional space. The covariance matrix corresponding to minimum volume ellipsoid is un-effected by outliers as outliers are excluded from analysis. The eigen-values of robust covariance matrix are generally smaller than eigen-values of covariance matrix estimated from whole sample data. Hence, the proposed significance test for identifying differentially expressed genes becomes more sensitive and identifies more genes as differentially expressed. This could increase the quality of results. The robust covariance matrix is used in Chapter 9.

The proposed method currently does not include replicates information. Since the gene expression data is known to be noisy, it is always recommended to use replicates. Replicates allow comparison of variation in gene expression within each group and between groups and improve the reliability of identifying differentially expressed genes. The idea of using within and between group variation should be included in PCA analysis. The Multiway Principal Component Analysis (MPCA) which is routinely used to analyze data from multiple batches could be used (with modifications) to explicitly include replicates.

The proposed method uses a hypothesis test to find the significance of the differential expression of a gene between two biological conditions. This test assumes that

difference of scores between WT and KO follows a multivariate normal distribution. The scores are the weighted linear combination of original expressions (Equation 4.4). As per the central limit theorem, linear combinations of different variables would follow normal distribution even if the individual variables are non-normal. If scores are normally distributed, so would their difference. We tested the normality of the difference of scores on each PC using quantile-quantile plots for mouse dataset (Figure 4.21) and Yeast cell-cycle dataset (Figure 4.22). The coefficient of determination, between the observed values and the expected values ranges from 0.92 to 0.97. We also tested the multivariate normality of scores using beta probability plot of Small (1978). In multivariate normality test, the proportionality between the ordered squared Mahalanobis distances is compared with beta distribution. A high correlation indicates the data is multivariate normal. The coefficient of determination, using all genes, is 0.65 for mouse dataset which further increases to 0.95 after removal of only 1% of genes (Figure 4.23). Similarly, for Yeast cell-cycle dataset the coefficient of determination is 0.81 when all genes are used and 0.96 after removal of 5% outlier genes (Figure 4.24). Hence, the assumption of the normality is reasonable.

Finally, the proposed method is useful especially for large datasets since it relies on PCA which is computationally efficient even for large number of genes. In large datasets, most of the genes are generally unchanged between different biological conditions. Consequently, the differential expression may not be reflected in all dominant PCs as the PCs are not driven by differential expression between different conditions. Even in such situations, the proposed method is sensitive to use the changes in scores on first a few dominant PCs and correctly identifies differentially expressed genes. To

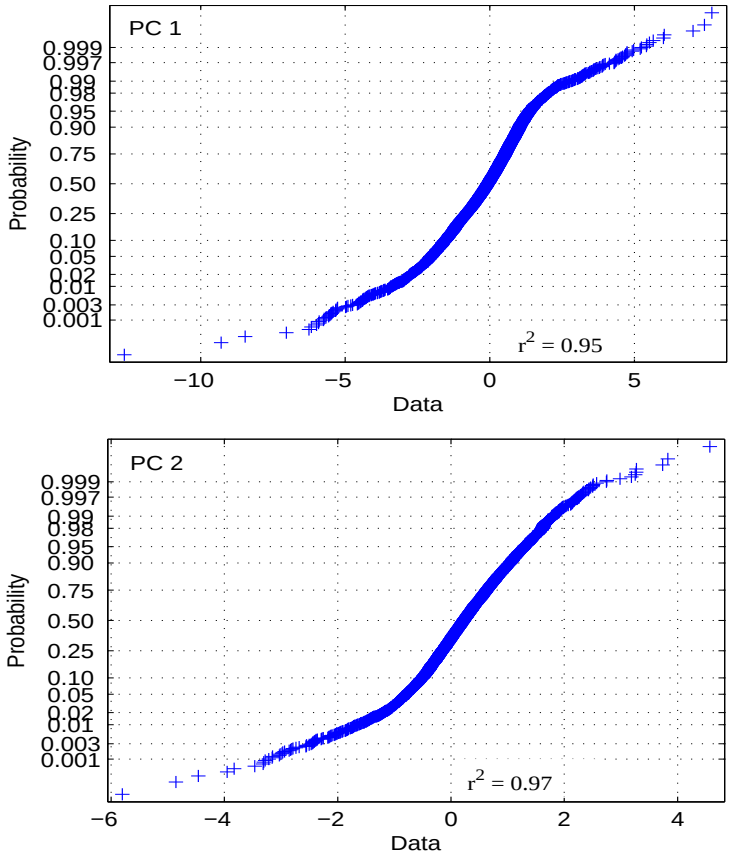

Fig. 4.21. Normal distribution plots for the difference of scores on individual PCs for mouse dataset. The coefficient of determination, r^2, between the observed values and the expected values ranges from 0.95 to 0.97.

illustrate this, we used the complete dataset containing all cell-cycle- and non-cell-cycle-regulated genes. The datasets contain measurements for 5696 genes at 18 time points. Considering the large number of genes, a more stringent p-value threshold of 0.001 is used instead of 0.05 that was used for the cell-cycle genes. We identified 151 genes as differentially expressed which contained 68 (out of 72) genes identified in the cell-cycle data alone.

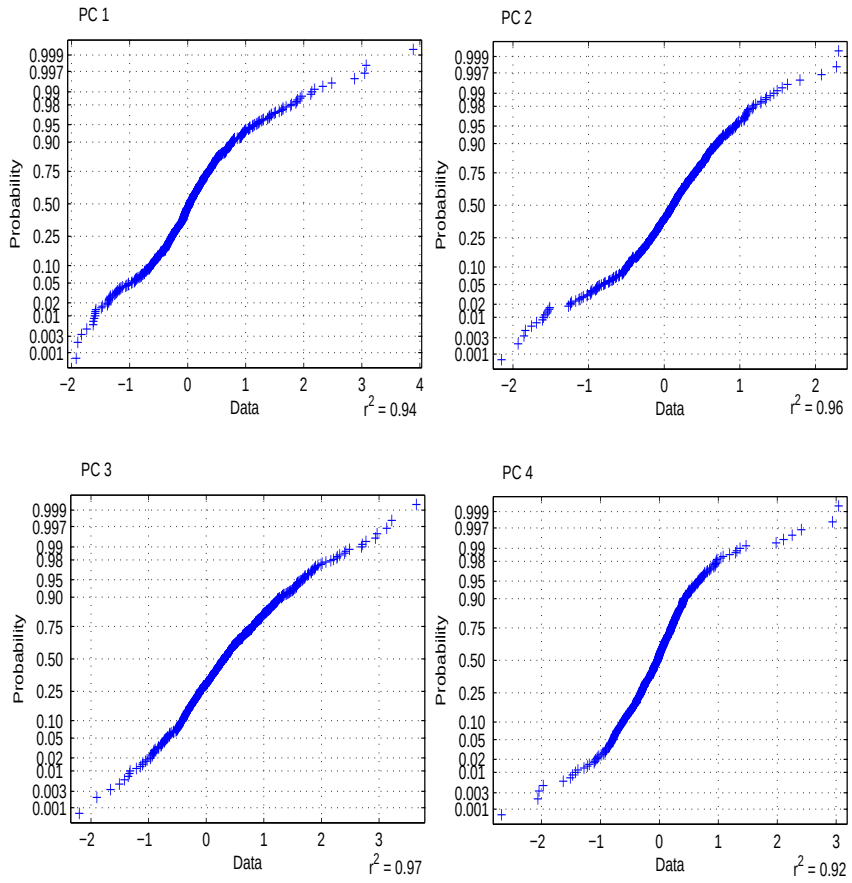

Fig. 4.22. Normal distribution plots for the difference of scores on individual PCs for Yeast cell-cycle dataset. The coefficient of determination, r^2, between the observed values and the expected values ranges from 0.92 to 0.97 indicating normal distributions for all directions.

Here, a method was proposed for identifying differentially expressed genes in time-course data. The proposed method was evaluated using two gene expression datasets and the results are compared with previously published results. The proposed method models the expression data from one condition using PCA and projects the expression data from different condition on the developed PCA model. The scores of genes are used to identify differentially expressed genes. Since scores represent the linear relation between the expression profile of genes and the PC, comparison of scores measures

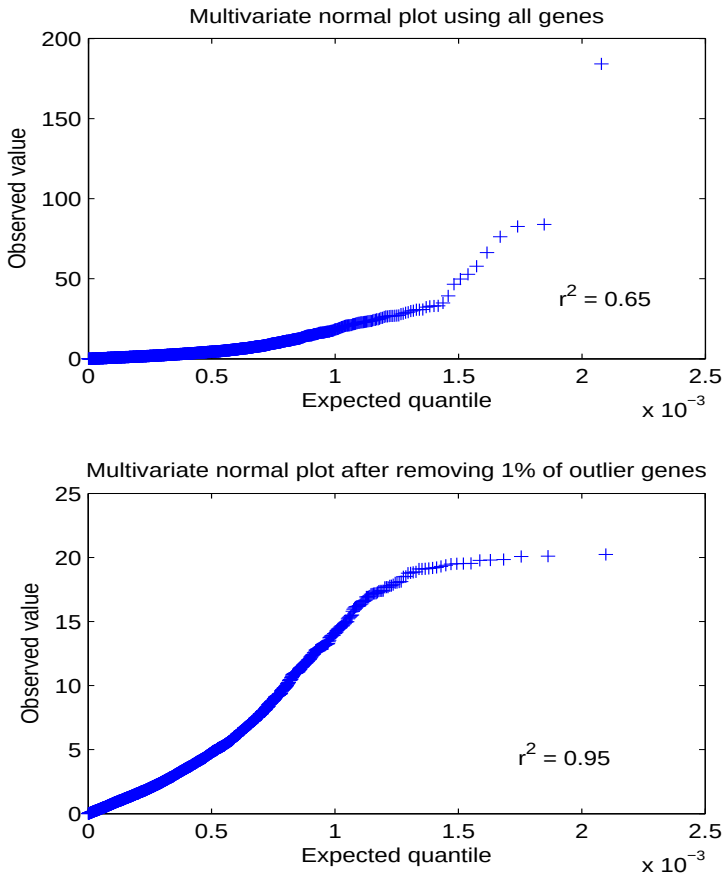

Fig. 4.23. Multivariate normal distribution plot for the difference of scores of mouse dataset. The coefficient of determination, r^2, is 0.65 when all genes are used and its value increases to 0.95 after removing only 1% of outlier genes.

the systematic variation in the gene expressions. In contrast to previously published methods that treat all the genes equally irrespective of actual expression levels (Storey *et al.*, 2005), direct comparison of expression profiles (Bar-Joseph *et al.*, 2003a) or not use of expression levels (Cheng *et al.*, 2006), our approach uses PCA where different PCs contribute differently to the gene expression profiles and provide comparison at multiple levels represented by different PCs. This is important because, for some genes a small change in expression is sufficient for change of biological function whereas

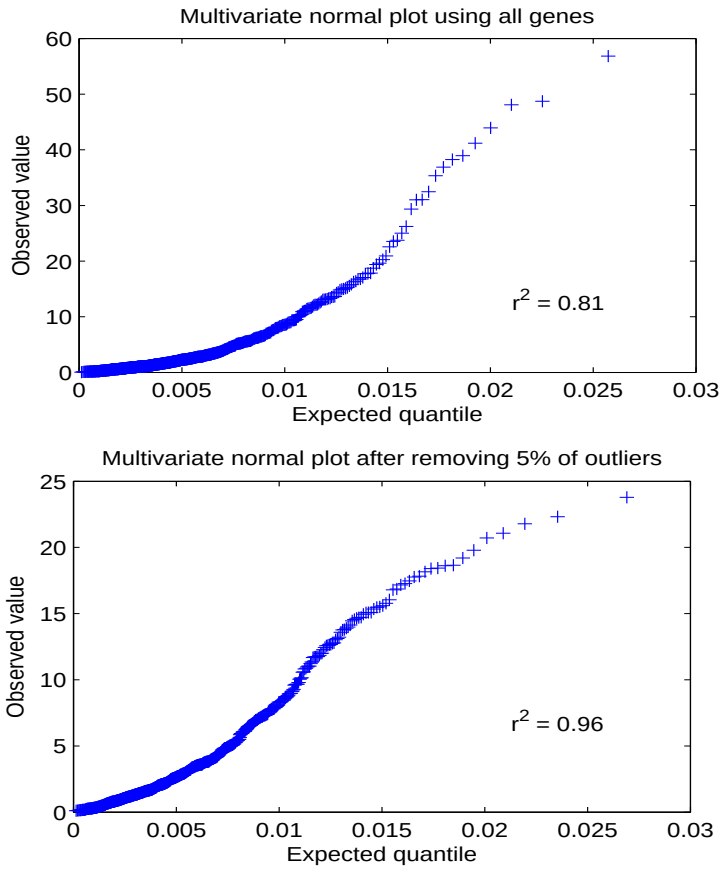

Fig. 4.24. Multivariate normal distribution plot for the difference of scores of Yeast cell-cycle dataset. The coefficient of determination, r^2, is 0.81 when all genes are used and its value increases to 0.96 after removing only 5% of outlier genes. The plots indicates that the multivariate normality assumption for the difference of scores is reasonable.

some genes require large expression change. Comparing genes at multiple levels considers these differences and identifies biologically meaningful genes that explain the biological phenomena. For example, CLN3 has similar scores on PC 1, 2 and 4 in both wild-type and mutant Yeast strain. However, it has large difference in score on PC 2 which made it to be identified as differentially expressed gene. None of the previously mentioned approaches identified this gene. It confirms that the proposed method identifies differentially expressed genes which have biologically meaningful information.

5. DETECTING ELLIPSOIDAL CLUSTERS IN GENE EXPRESSION DATA

5.1 Introduction

Clustering is the most widely used data-mining tool for gene expression data analysis. The objective of clustering gene expression data is to organize large number of genes into a few groups, called clusters, such that genes within a cluster are more similar in expression compared to genes belonging to other clusters. Some of the advantages of clustering gene expression data such as functional annotation, identifying TFs etc are described in Section 2.2.

Several clustering approaches have been used for clustering genes including hierarchical (Eisen *et al.*, 1998), k-means (Tavazoie *et al.*, 1999), Self-Organizing Maps (SOM) (Tamayo *et al.*, 1999), graph-theoretic (Sharan *et al.*, 2003), model-based (Yeung *et al.*, 2001) and fuzzy clustering algorithms (Dembele and Kastner, 2003). Gibbons and Roth (2002) compared different clustering algorithms using real gene expression data with functional enrichment of clusters as objective. The study shows that the performance of hierarchical clustering is poor and more or less equal to random clustering. It also shows that partitional clustering methods such as k-means and SOM perform better than hierarchical clustering.

Although the partitional clustering methods are successful in some cases, they have the following drawbacks. Any partitional clustering algorithm has two critical compo-

nents: (1) the distance metric used for measuring the similarity of expression profiles, and (2) an algorithm for assigning each gene to a cluster. The is a challenge associated with each component:

1. The generally used Euclidean distance metric identifies spherical clusters whereas the objective of clustering is to identify the natural structure in the data.

2. The optimization algorithm for assigning genes to clusters generally lead only to a local minima whereas reaching global minimum is preferred.

Let $X_{n\times p}$ denotes the time-course gene expression data with n genes measured at p time-points. Each element x_{ij} is the expression level of the i^{th} gene at the j^{th} time-point. The objective of partitional based clustering algorithms is to classify these genes $X = \{x_1, x_2, x_3, ..., x_n\}$ into k disjoint clusters represented as $C = \{C_1, C_2, C_3, ..., C_k\}$ such that the total genes to cluster centroid distance is minimum:

$$J = \sum_{i=1}^{n}\sum_{j=1}^{k} \mu_{ij} D^2_{ijA_j} \tag{5.1}$$

where $\mu_{ij} = 1$ if x_i belongs to cluster C_j and 0 otherwise. $D^2_{ijA_j}$ is the distance metric that measures the distance between the given gene x_i to the centroid of cluster given by:

$$D^2_{ijA_j} = ||x_i - v_i||^2_{A_j} = (x_i - v_j)A_j(x_i - v_j)^T \tag{5.2}$$

where v_j are the cluster centroids given by:

$$v_j = \frac{1}{n_j}\sum_{i=1}^{n} \mu_{ij} x_i \tag{5.3}$$

where n_j is the number of genes in j^{th} cluster. The matrix A_j is called as norm matrix which is a positive definite symmetric matrix. The norm matrix determines the size and shape of cluster. Since the cluster shapes and sizes are unknown a priori, A_j are typically taken to be identity matrix. Hence, $D^2_{ijA_j}$ becomes Squared Euclidean distance. The Squared Euclidean distance identifies hyper-spheroid clusters in the data (Krishnapuram and Kim, 1999).

Methods such as GK (Gustafson and Kessel, 1979), GG (Gath and Geva, 1989), and HEC (Mao and Jain, 1996) have been proposed to overcome the drawback associated with Euclidean distance. The feature of these methods is the adaptation of distance metric to the shape of cluster by estimating A_i from the data. To have a non-trivial solution for Equation 5.1, additional constraint is necessary for A_j. The constraint that is generally used is

$$det(A_j) = \rho_j, \quad \rho_j > 0, \quad 1 \leq j \leq k \tag{5.4}$$

where ρ_j is the volume for each cluster. This allows clusters to have different shapes while the cluster volume is fixed. Solving Equation 5.1 results A_j as the inverse fuzzy covariance matrix of each cluster (Gustafson and Kessel, 1979). For partitional clustering approaches, A_j is replaced with inverse covariance matrices (Mao and Jain, 1996). Hence, the distance metric is given by:

$$D^2_{ijA_j} = ||x_i - v_i||^2_{A_j} = (x_i - v_j)\Sigma_j^{-1}(x_i - v_j)^T \tag{5.5}$$

where Σ_j is the covariance matrix for cluster j.

Then $D^2_{ijA_j}$ becomes Squared Mahalanobis distance which generally identifies ellipsoidal clusters. The GK clustering has been employed with gene expression data and showed that it outperforms k-means and SOM indicating the importance of adaptive distance metric (Kim *et al.*, 2005). However, problems arise when estimating the covariance matrix when the number of genes in the cluster are smaller than number of time-points or if the time-points are linearly correlated (Babuska *et al.*, 2002). In such cases, the covariance matrix becomes singular or close to singular and cannot be inverted for calculation of adaptive distance metric. In adaptive distance calculation, the distance in the directions of major axes are reduced and the distance in minor axes are magnified. Such normalization leads to elongation of clusters in the direction of larger variance and grab objects from other clusters (Krishnapuram and Kim, 1999). The singularity of covariance matrix add to this problem since singularity of covariance matrix leads to nearly zero variance in some directions (Babuska *et al.*, 2002).

The problems with GK clustering with singularity of covariance matrix is illustrated using artificial data shown in Figure 5.1. The data contains 500 objects over three dimensions and they are arranged into three clusters. The determinant of covariance matrices for clusters 1, 2, and 3 are 37.19, 0.626, and 4.242×10^{-7}, respectively. The values for z direction of cluster 3 are generated using the formula $z = x + y + f$ where x and y represent the first and second dimensions and f is uniform distribution in the range [0 0.001]. So, the covariance matrix for cluster 3 becomes near singular. The resultant partition from GK clustering method is shown in Figure 5.2. As can be

seen from Figure 5.2, Cluster 3 is elongated and incorrectly takes objects from Clusters 1 and 2. More importantly, the elongation is in the direction of third dimension (inset in Figure 5.2) which shows the effect of singularity of covariance matrix. Kim *et al.* (2005) used large datasets (approximately 6000 genes) in their study and results are compared for number of clusters k in the range [2 10]. Hence, the problems with singularity of covariance matrix in GK clustering for gene expression data analysis are not revealed in their study.

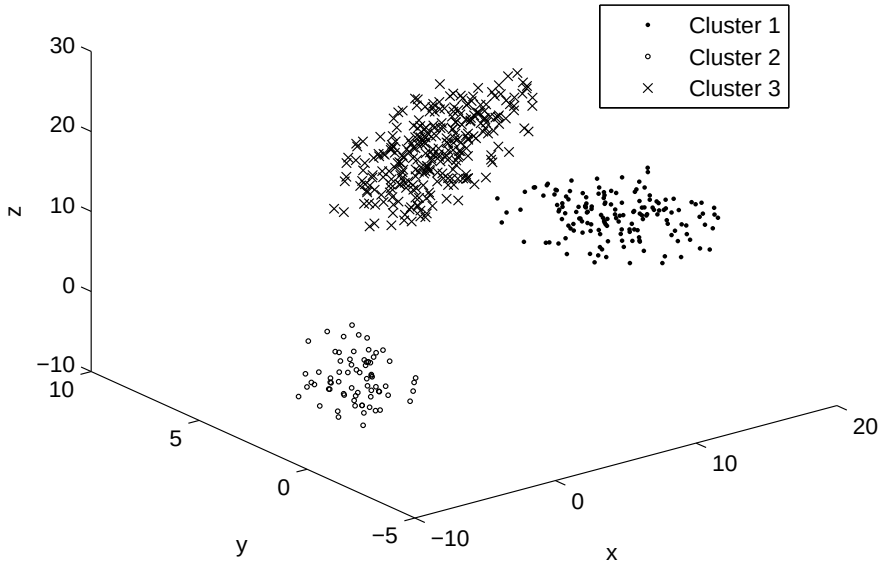

Fig. 5.1. Artificial dataset containing 500 objects arranged into three clusters

From the above illustration, it is clear that methods based on adaptive distance fail for cluster data if covariance matrix becomes singular. Singularity of covariance matrix is common in gene expression data as different time-points are often correlated to each other. For example, the determinants of covariance matrices of clus-

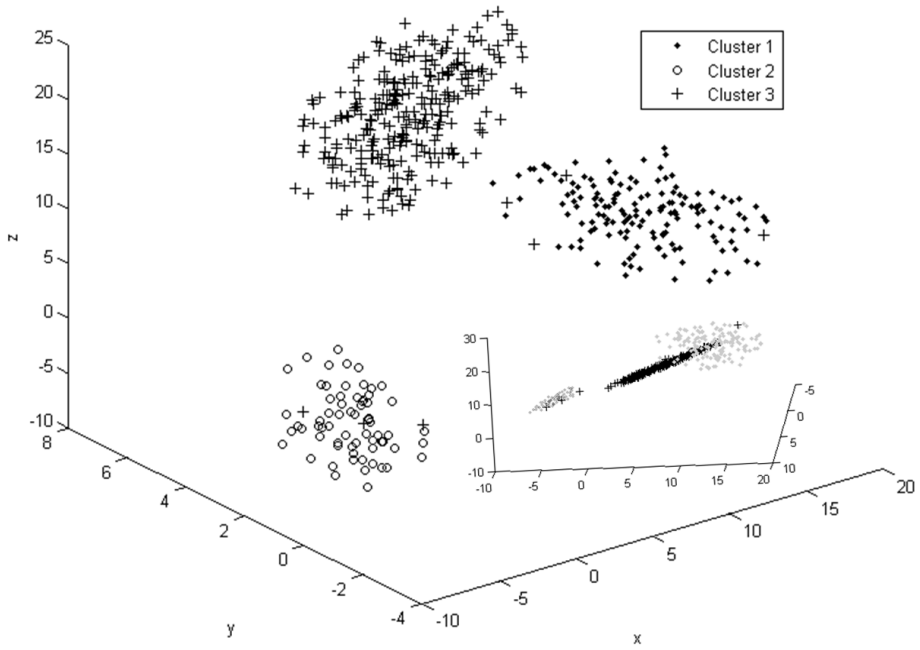

Fig. 5.2. Results from GK clustering for artificial data. Cluster 3 is extended and incorrectly takes objects from other clusters

ters reported by Spellman *et al.* (1998) in yeast cell-cycle data are in the range of $[4.25 \times 10^{-99} \quad 2.07 \times 10^{-229}]$ indicating that the covariance matrices are singular. This necessitates the development of clustering methods that are able to identify ellipsoidal clusters in gene expression data.

Here, a clustering method is proposed. The proposed method takes the natural structure, *i.e.* 'shape', of the cluster into consideration while calculating the distance and able to identify clusters even the covariance matrix becomes singular. In order to address the issues with minimization of objective function, the proposed method uses Genetic Algorithms to optimize the objective function towards a global minimum.

5.2 Methods

In this section, a Principal Components Analysis based clustering method is proposed to identify ellipsoidal clusters in gene expression data. The proposed method employs PCA which splits the original space spanned by the gene expression data into two subspaces, namely a *PCA subspace* and a *residual subspace*. The *PCA subspace* is formed by the dominant PCs that capture most of the variance in data and *residual subspace* is formed by non-dominant PCs that capture the remaining variance. A PCA distance metric that measures the distance in both subspaces independently and then combines them is used. The distance measured in *PCA subspace* is the squared Mahalanobis distance that captures the geometrical shape of cluster whereas the distance measured in *residual subspace* is the squared Euclidean distance which represents the thickness of cluster. Since the *PCA subspace* is formed by dominant PCs, the covariance matrix is non-singular and hence, the problem with singularity of covariance matrix is eliminated. Reduced distances are calculated from these two measurements which are comparable across clusters. The objective function for clustering is formed using this distance metric. A Genetic Algorithm based optimization procedure is used to minimize the objective function towards global minimum and identify clusters in gene expression data.

5.2.1 PCA distance metric

The PCA distance metric employs PCA on each cluster for calculating the distance from each gene to its cluster centroid. PCA is a multivariate statistical technique that finds the Principal Components (directions) of variability in the data, and transforms

the related variables into a set of uncorrelated ones (Jackson, 1991). Mathematically, PCA is a linear transformation of original data in such a way that the covariance matrix becomes diagonal. Then Equation 5.2 becomes

$$D^2_{ijA_j} = x_i^{'} {\Sigma_j^{'}}^{-1} {x_i^{'}}^T \tag{5.6}$$

where $\Sigma_j^{'}$ is the covariance matrix of j^{th} cluster which is diagonal and off-diagonal elements are zero. PCA is employed with column mean centered cluster so the cluster centroid, v_j, moves to the origin. The superscript $'$ indicates the values after PCA transformation.

The diagonal elements of $\Sigma_j^{'}$ are the eigenvalues $\{\lambda_1, \lambda_2, ...\lambda_p\}$ of the covariance matrix. Equation 5.6 can also be written as

$$D^2_{ijA_j} = x_i^{'} \begin{pmatrix} 1/\lambda_1 & 0 & \dots & 0 \\ 0 & 1/\lambda_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1/\lambda_p \end{pmatrix} {x_i^{'}}^T \tag{5.7}$$

The PCs are arranged in the descending order of the variance they capture, *i.e.* $\{\lambda_1 \geq \lambda_2 \geq \lambda_3 ... \geq \lambda_p\}$. The first few PCs capture most of the variance and represent the dominant patterns in the data. The last PCs capture very little variance are essentially represent noise. By selecting the dominant PCs suitably the whole space can be divided into two subspaces namely *PCA subspace* and *residual subspace*. The PCA subspace is spanned by the dominant PCs whereas the *residual subspace* is spanned by the remaining PCs. Hence, the distance of a gene to its cluster centroid can be calcu-

lated individually in both subspaces and then combined to get the total distance. Since *PCA subspace* is spanned by dominant PCs, a distance measure that captures the geometrical shape of cluster using covariance matrix can be used. The distance measure used in the *PCA subspace* is the Hotelling's T^2 which is squared Mahalanobis distance. The Hotelling's T^2 for i^{th} gene in j^{th} cluster is given by:

$$T_{ij}^2 = [x'_{i1}, x'_{i2}, ..., x'_{il}] \begin{pmatrix} 1/\lambda_1 & 0 & \dots & 0 \\ 0 & 1/\lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \dots \\ 0 & 0 & \dots & 1/\lambda_{l_j} \end{pmatrix} [x'_{i1}, x'_{i2}, ..., x'_{il}]^T \tag{5.8}$$

where l_j is the number of dominant PCs for j^{th} cluster.

The distance measures in the *residual subspace* is the Q statistic which is the squared Euclidean distance that measures the perpendicular distance from the *PCA subspace* to the gene . The Q statistic for i^{th} gene in j^{th} cluster is given by:

$$Q_{ij} = (x'_{i(l_j+1)}, x'_{i(l_j+2)}, ..., x'_{ip})(x'_{i(l_j+1)}, x'_{i(l_j+2)}, ..., x'_{ip})^T \tag{5.9}$$

The distances measured by both T^2 and Q are shown in Figure 5.3. As shown in Figure 5.3, Hotelling T^2 measures the distance between the gene and the centroid of the cluster in the *PCA subspace* and Q statistic measures the distance in *residual space*. Hotelling T^2 considers the variation in different directions and weights the directions in order of decreasing order of the variation in those directions. This will make the identification of the ellipsoidal clusters. In the *residual subspace*, the proposed approach

measures the perpendicular distance from the *PCA subspace i.e.* measures the thickness of the cluster.

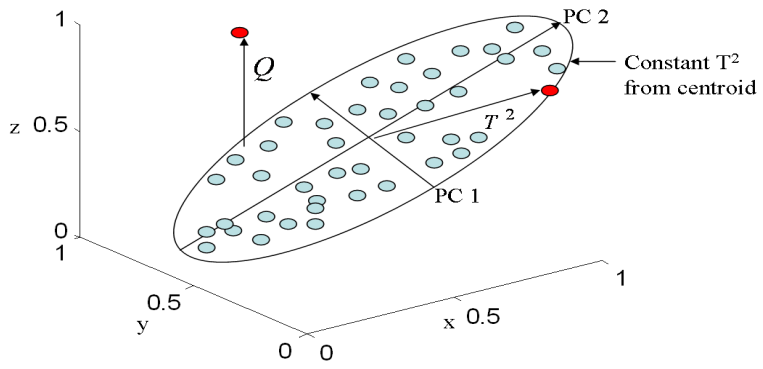

Fig. 5.3. Graphical visualization of proposed distance metric

Residual distances

The Hotelling's T^2 and Q statistic are comparable across clusters if all clusters used same number of PCs. However, different clusters are needed to modeled different clusters as the correlation between time-points are different. To account for this, the T^2 and Q metrics should be made independent of number of PCs used for modeling. The common procedure for making T^2 and Q independent of number of PCs is by normalizing by confidence limits for T^2 and Q. The residual distances are given by:

$$T^2_{ijr} = T^2_{ij}/T_{\alpha j} \tag{5.10}$$

$$Q_{ijr} = Q_{ij}/Q_{\alpha j} \tag{5.11}$$

where $T_{\alpha j}$, $Q_{\alpha j}$ are the values for confidence limits for T^2 and Q statistics corresponding to j^{th} cluster. α is the confidence level which is generally taken as 0.95 (95% confidence).

The confidence limits for T^2, $T_{\alpha j}$, can be calculated by means of the F distribution

$$T_{\alpha j} = \frac{l_j(n_j - 1)}{n_j - l} F_{l_j, n_j - l_j, \alpha} \tag{5.12}$$

where, l_j is the number of PCs used for j^{th} cluster and n_j is the number of genes in j^{th} cluster. Confidence limit for Q statistic, $Q_{\alpha j}$, can be calculated provided all the eigenvalues $\{\lambda_1, \lambda_2, ..., \lambda_p\}$ of the covariance matrix are available (Jackson, 1991)

$$Q_{\alpha j} = \Theta_1 \left[\frac{c_\alpha \sqrt{2\Theta_2 h_0^2}}{\Theta_1} + 1 + \frac{\Theta_2 h_0 (h_0 - 1)}{\Theta_1^2} \right]^{\frac{1}{h_0}} \tag{5.13}$$

where c_α is the standard normal deviate corresponding to upper $(1 - \alpha)$ percentile and

$$\Theta_a = \sum_{b=l_j+1}^{p} \lambda_b^a \quad for\ a = 1, 2, 3 \tag{5.14}$$

$$h_0 = 1 - \frac{2\Theta_1 \Theta_3}{3\Theta_2^2} \tag{5.15}$$

The Hotelling's T_r^2 and the Q_r statistic are generally combined as given below to calculate the total distance from an object to its centroid (Wold, 1976)

$$D_{ijA_j}^2 = \sqrt{(T_{ijr}^2)^2 + (Q_{ijr})^2} \tag{5.16}$$

Clustering based on PCA distance metric

The PCA distance metric given in Equation 5.16 can be used with clustering objective function (Equation 5.1) to identify ellipsoidal clusters in gene expression data. However, we need additional constraints to avoid non-trivial solution for clustering. Without constraints, the minimization procedure allows clusters to grow arbitrarily large resulting an arbitrary partition. This is avoided by constraining the cluster volumes (Gustafson and Kessel, 1979). Since the proposed distance metric splits the space into two subspaces, we need to have constraints on cluster volume on both spaces. Cluster volumes in *PCA subspace* and *residual subspace* are the product of eigenvalues for the PCs spanned in those subspaces. Hence, the cluster volume for *PCA subspace* and *residual subspace* are $\prod_{a=1}^{l_j} \lambda_a$ and $\prod_{a=l_j+1}^{p} \lambda_a$, respectively. In the proposed method, the cluster volumes are constrained as given below

$$\prod_{a=1}^{l_j} \lambda_a = \rho_{pca} \quad \rho_{PCA} > 0 \tag{5.17}$$

$$\prod_{a=l_j+1}^{p} \lambda_a = \rho_{res} \quad \rho_{RES} > 0 \tag{5.18}$$

where ρ_{pca} and ρ_{res} are the volume of cluster in PCA and residual spaces, respectively.

Adding the constraint on cluster volume and selecting ρ_{pca} and ρ_{res} as 1 in the absence of actual cluster volumes, results in the final equation for the proposed distance metric as

$$D^2_{ijA_j} = \sqrt{((\prod_{a=1}^{l_j} \lambda_a))^{(1/l_j)} T^2_{ijr})^2 + ((\prod_{a=l_j+1}^{p} \lambda_a))^{(p-l_j)} Q_{ijr})^2} \tag{5.19}$$

The objective function for clustering is thus given by:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} \mu_{ij} D^2_{ijA_j} \tag{5.20}$$

The above equation is minimized to identify clusters in gene expression data. Here, a Genetic Algorithm based optimization procedure is used to minimize the objective function to identify clusters. The cross-validation approach proposed by Wise and Ricker (1991) is used for finding the number of PCs in each cluster. The cross-validation approach is described in Section 4.2.1. The GA approach for clustering is described in following section.

5.2.2 Minimization of objective function using GA

The objective of clustering is to assign the genes into predefined number of clusters, k, that minimizes the objective function shown in Equation 5.20. Here, Evolutionary Algorithms are used to minimize the objective function to avoid trapping in the local optima. Evolutionary Algorithms are stochastic optimization algorithms which are based on natural process of evolution: natural selection, mutation and crossover. In this thesis, the Genetic Algorithms (GA) proposed by Holland (1975) which comprise the majority part of Evolutionary algorithms is used. Krishna and Murty (1999) used GA for clustering and have shown that GA converges to the best optimum.

GA works on population of solutions, $P = \{s_1, s_2, ..., s_M\}$, to solve the optimization problem. A solution, s_i, consists of a string of symbols or binary values and it is associated with fitness value generated from objective function value. The number of solutions that GA works on is called as population size, M. During evolution process, GA produces new population from current population by applying genetic operators such as mutation, crossover and natural selection. The crossover and mutation help to explore the search space where as the natural selection operators selects the next population from the current population with probability proportional to fitness value. After a given number of generations, N , the solution with the best fitness is selected as the final solution for the optimization problem.

In the current work, the population is initialized randomly such that each solution s_i is of length of number of genes, n, and selected from uniform distribution over the set $\{1, 2, ..., k\}$. This means that each solution is a clustering result where each gene is assigned to one of the k clusters. In the subsequent generations, the solutions are updated using crossover and mutation operations and new population is selected using natural selection.

Crossover

In general, the crossover operator works on two solutions, called parents, from old population and exchanges portions of it to generate two new solutions called children. However, this operator is not directed by objective function and hence inefficient. So, the crossover operator is replaced with the new operator proposed by Krishna and Murty (1999). The new operator works on individual solution and reassigns each gene to new

cluster based on minimum centroid distance (Equation 5.19). The procedure includes calculations of centroids for clusters from the given solution s_i and then calculation of the distance from each gene to the clusters centroids. Then each gene is reassigned to the clusters for which the centroid distance is minimum. This directed approach always minimizes the objective function (Krishna and Murty, 1999).

Mutation

Mutation operator is also works on individual solution. For each gene in a given solution, s_i, a random number is generated from uniform distribution within the range $[0\,1]$. If this random number is smaller than predefined mutation probability, p_m, then, that gene is reassigned to a new cluster chosen randomly from the set $\{1, 2, ..., k\}$ with probability given as:

$$p_r = \frac{D_{ij,max} - D_{ij}}{\sum_{j=1}^{k} D_{ij,max} - D_{ij}} \tag{5.21}$$

The crossover and mutation operators described above work of individual solutions and update them. Hence, the updated population is taken as population for the next generation. The natural selection operator, hence , not necessary.

In the next section, the proposed clustering method is evaluated using artificial and real gene expression datasets and results are compared with other clustering approaches meant for identifying clusters of different geometric shapes.

5.3 Results

Here, the performance of the proposed clustering technique is illustrated using artificial data shown in Figure 5.1 and two gene expression time-course datasets.

5.3.1 Case Study 1: Artificial dataset

The first dataset is the artificial dataset shown in Figure 5.1. The partition resulted form clustering approaches based on covariance matrix such as GK clustering is not accurate for such datasets (Figure 5.2) due to the singularity of covariance matrix of cluster 3. Here, the proposed method is tested using this dataset to show efficacy of proposed method to identify clusters even when the covariance matrix becomes singular.

The proposed clustering approach uses GA to partition data while minimizing the global distance metric given in Equation 5.20. The population size M and mutation probability, p_m, are selected as 100 and 0.01, respectively. The number of generations, N, for GA is set to 100. The resulted partition is shown in Figure 5.4. All the three clusters are clearly identified without overlap. Since the structure of this is known to us, the number of PCs is selected such that the selected PCs capture at least 93% of variation. The proposed clustering method models cluster 1 using 3 PCs whereas cluster 2 and 3 are modeled using 2 PCs. The first 2 PCs capture 97% of variance in cluster 2. Since cluster 3 actually contains only two independent dimensions, 2 PCs are sufficient for modeling this clusters. The performance of GA for minimizing the objective function is shown in Figure 5.5. The minimum value for objective function

over all solutions in the population is shown as a function of number of generations. The objective function is minimized within a few generations and unchanged thereafter. This indicates that the GA technique is able to find the best possible minimum in this case though it is not guaranteed to be globally minimum.

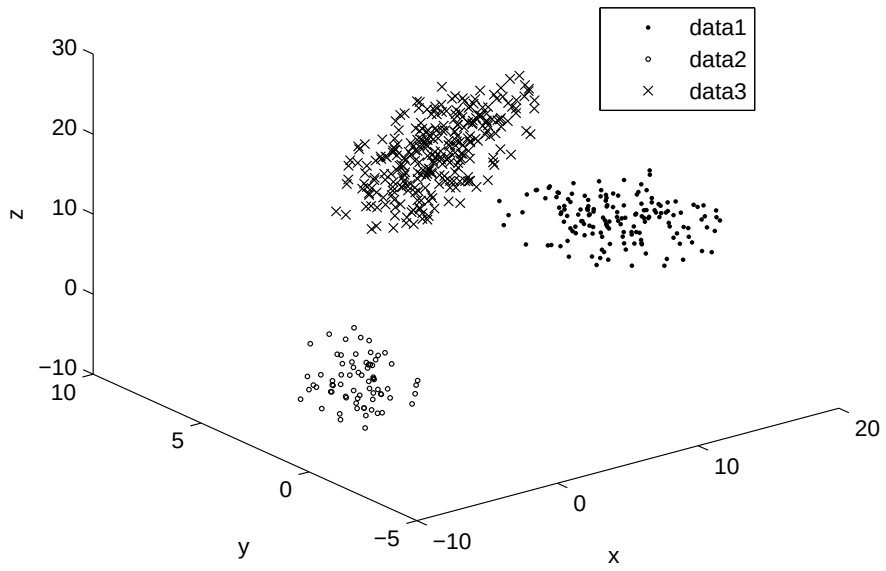

Fig. 5.4. Resulted partition for artificial data from the proposed clustering approach.

5.3.2 Case Study 2: Human macrophage dataset

The first gene expression dataset is the response of human macrophages to pathogens conducted by Nau *et al.* (2002). Macrophages are large versatile immune cells. They play important role in host defence by recognizing, swallowing, and killing microorganisms. Understanding the response of macrophages to bacteria provides insights of tactics used by bacteria to circumvent these responses and hence helps disease prevention. In this study DNA microarrays were used to capture the response of human

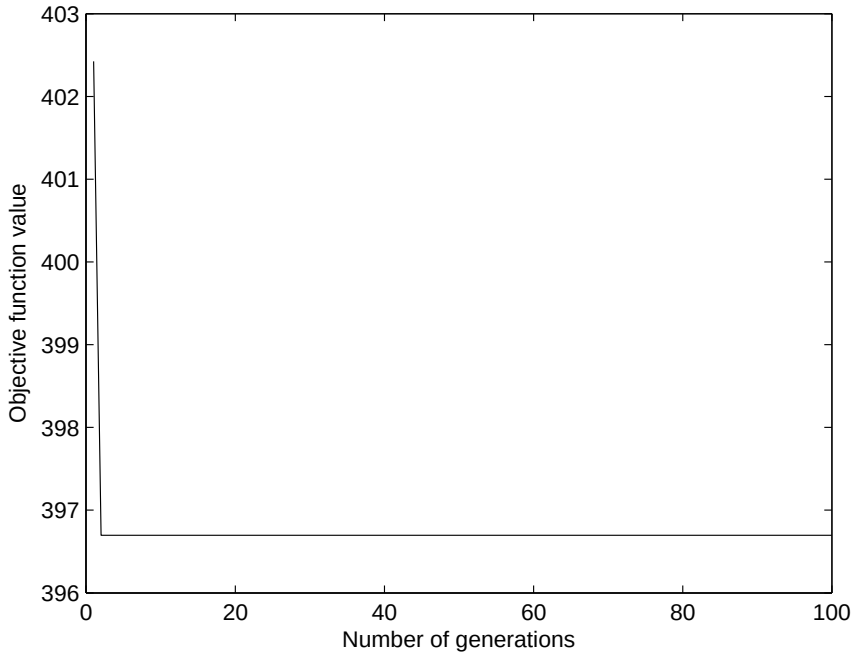

Fig. 5.5. Performance of GA in minimizing the objective function.

macrophages to a variety bacteria. Gene expression levels are measured for 6800 genes at five different time-points over a period of 24 hrs. Out of the several genes that have significant expression during the experiment, 198 genes are similarly responded to all the eight pathogens used in this test. Nau *et al.* (2002) clustered these 198 genes into two distinct clusters. The first cluster contains genes up-regulated during the experiment and the second cluster contains genes that are down-regulated. We used the proposed algorithm on this data to identify these two clusters.

The proposed algorithm is used with a population size of 100 and mutation probability of 0.01. The number of generations are kept to 300. The objective function is shown in Figure 5.6 as a function of generations. The objective function is unchanged after 12 iterations. No change in objective function was observed even after 1000 iter-

ations. The heatmap of the two clusters identified in this dataset are shown in Figure 5.7. Heatmap is frequently used in gene expression data visualization. In heatmap, red color is used to indicate the up-regulation of genes and green for down-regulation. As shown in Figure 5.7, two patches of red and green are formed by these clusters. It clearly shows that the proposed method groups the genes based on their similarity.

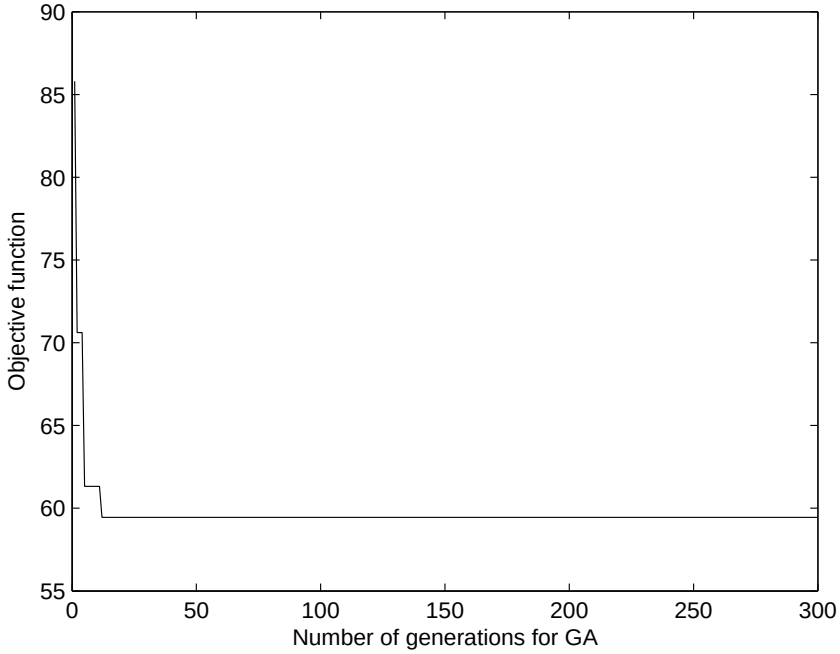

Fig. 5.6. Performance of GA in minimizing the objective function for Human macrophage dataset.

We compare the results using k-means, GK, GG clustering algorithms with the reported partition for this dataset. Results from the proposed method are also compared with reported partition to show the efficacy of proposed method. For this, the reported partition is first shown in two dimensional PC scores plot (Figure 5.8). These two PCs capture 93.09% of overall variance. From Figure 5.8, it is clear that the both clusters

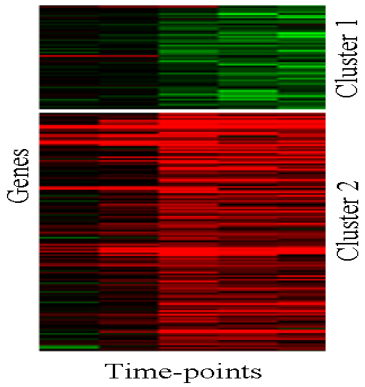

Fig. 5.7. Heatmap of two clusters identified by proposed method in Human macrophage dataset.

in this dataset are clearly separated from each other. The shape of Cluster 1 is elliptical where as the shape of Cluster 2 is spherical with some outliers. The results for k-means clustering is shown in Figure 5.9. The identified clusters are in spherical shape. This is because of the Euclidean distance used for clustering that forces clusters to have spherical shapes. Due to this, Cluster 1 is extended and incorrectly takes genes from Cluster 2.

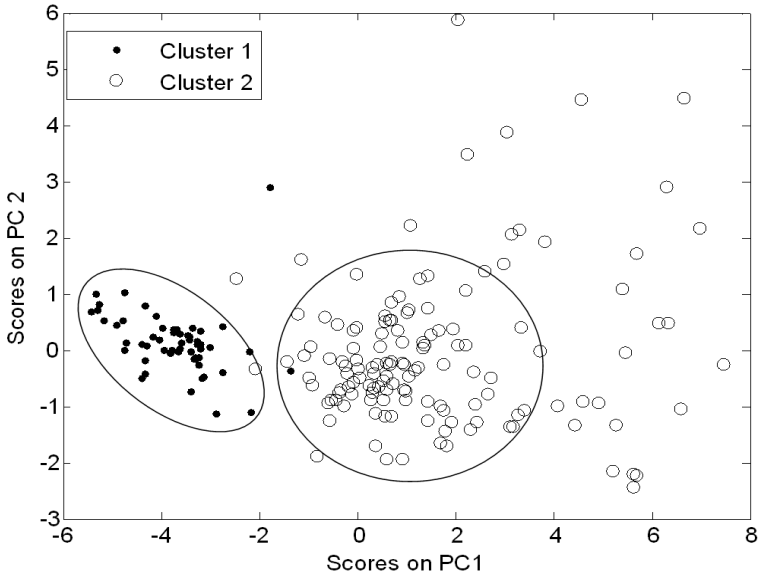

Fig. 5.8. Scores plot of reported partition for Human macrophage dataset

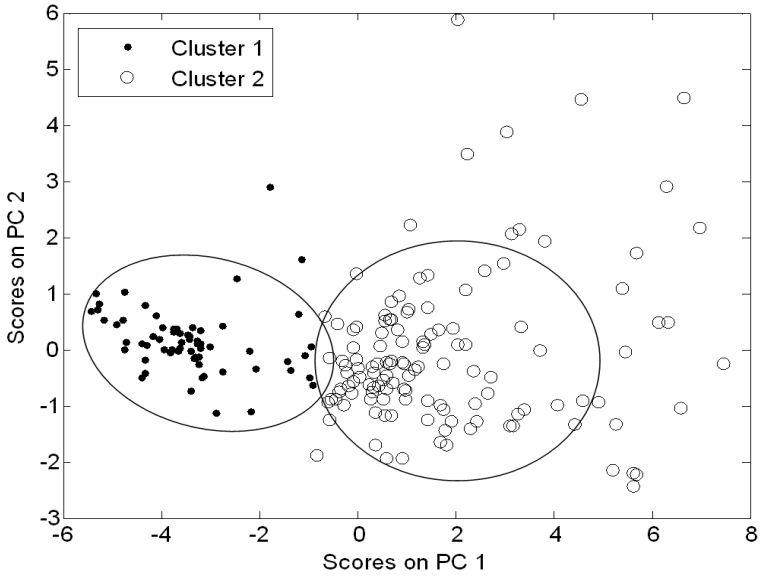

Fig. 5.9. Scores plot of clustering result for Human macrophage dataset using k-means clustering. Cluster 1 is extended and incorrectly takes genes from Cluster 2

The resultant partition for GK and GG clustering methods are shown in Figure 5.10 and Figure 5.11, respectively. The GK and GG clustering algorithms used in this thesis are from the Fuzzy Clustering and Data Analysis Toolbox developed by Janos Abonyi, Balazs Balasko, and Balazs Feil at the Department of Process Engineering at the University of Veszprem, Hungary. As shown in Figure 5.10, GK clustering is able to model Cluster 1 as elliptical cluster whereas Cluster 2 is modeled as spherical. But, Cluster 1 is allowed to extend and incorrectly take genes from cluster 2. The result from GG clustering is also similar to result from GK clustering. This is due to the near singularity of the covariance matrix of cluster 1 (the determinant of covariance matrix for cluster 1 reported by Nau *et al.* (2002) is 0.0014).

Figure 5.12 shows the clustered partition resulted from the proposed PCA clustering method. The clusters identified by proposed method are clearly separated without any

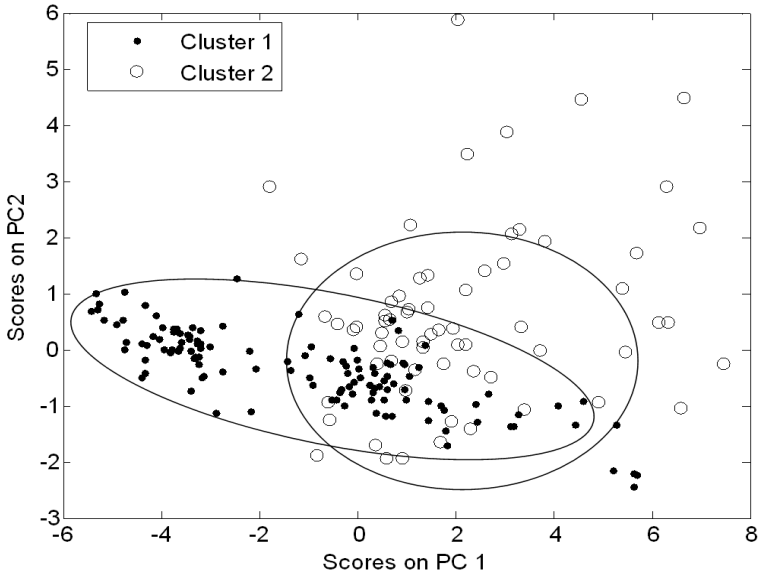

Fig. 5.10. Scores plot of clustering result for Human macrophage dataset from GK clustering approach. Cluster 1 is extended and incorrectly takes genes from cluster 2.

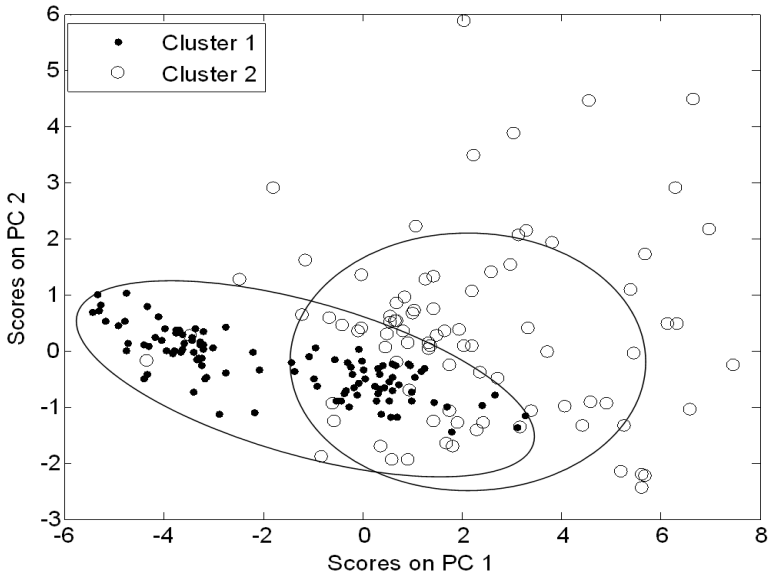

Fig. 5.11. Scores plot of clustering result for Human macrophage dataset from GG clustering approach. Cluster 1 is extended and incorrectly takes genes from cluster 2

overlap. The proposed algorithm used 1 PC to model cluster 1 and 2 PCs for cluster 2. The number of PCs used for cluster 1 captured only 39% of variance in that cluster.

A close look at the eigenvalues for this cluster shows that all eigenvalues are small. The first and largest eigenvalue is 0.811 and the second and third eigenvalues are 0.58 and 0.36, respectively. This might be the reason for selecting only 1 PC for modeling this cluster. The total variance captured by 2 PCs used for modeling cluster 2 capture 89.27% of the total variance in this cluster.

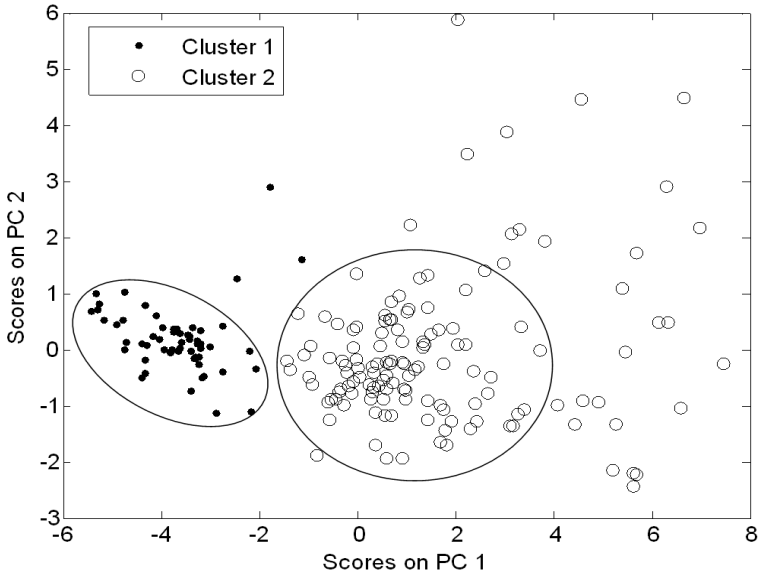

Fig. 5.12. Scores plot of clustering results from proposed clustering method for Human macrophage dataset. Both the identified clusters are clearly separated

Since the 'true' (reported) partition is available for this dataset, we verify the similarity between the partitions using clustering algorithms and the reported partition using Jaccard Coefficient (JC). JC is a measure of the similarity between two partitions. Let C^1 be the partition from the clustering algorithm and C^2 be the reported solution. The JC measures the extent to which C^1 matches with C^2

$$JC = \frac{n_{11}}{n_{11} + n_{10} + n_{01}} \tag{5.22}$$

where n_{11} is the number of pairs of objects that are in the same cluster in both C^1 and C^2, n_{10} is the number of pairs of objects that are in the same cluster in C^1 but not in C^2, and n_{01} is the number of pairs of objects that are in the same cluster in C^2 but not in C^1. JC takes a value between 0 (complete mismatch) and 1 (perfect match). The better the agreement between identified and the 'true' solution, the higher the value of JC. The JC for the partition resulted from the proposed method is 0.9332. The JC for k-means is 0.8562. This indicates that the ellipsoidal clusters results in partitions which are close to experts partition. The JC for partition from GK and GG clustering methods are 0.4143 and 0.4451,respectively. This clearly indicates that results from proposed clustering method are better than the other methods for clustering.

5.3.3 Case Study 3: Yeast diauxic dataset

The second gene expression dataset is from the Yeast *Saccharomyces cerevisiae* diauxic shift study from Brauer *et al.* (2005). In this study, the physiological response of Yeast was studied in glucose limiting condition in batch and steady-state cultures followed by global patterns of gene expression. During experiment, expression profile of 2284 genes were measured over 12 time-points with 15 min interval starting from 7.15 hrs to 10 hrs. In the initial phase, *Saccharomyces cerevisiae* preferably metabolize glucose by using high-flux, fermentative Embden-Meyerhof pathway and produce even when oxygen is abundant. When the glucose is exhausted, cells undergo a diauxic shift in which cells switch to fully respiratory metabolism and catabolize carbon compounds through TCA cycle (Brauer *et al.*, 2005). During this shift, several genes initially expressed as maximal are down-regulated and some other genes are activated to export and metabolize the new substrate. Here, this large gene expression dataset is used to

show the efficacy of proposed clustering approach to group functionally related genes.

Instead of clustering with fixed number of clusters, results are generated for different number of clusters from $k = 2$ to $k = 10$. For all the cases, the population size is fixed as 200 and the number of generations is selected as 500 considering the large number of genes. The value of objective function reaches minimum after some iterations for all situations and it is relatively unchanged with further increasing number of iterations (Figure 5.13). This indicates that the proposed GA algorithm is performing correctly. However, it is clear that the number of iterations should be increased with increasing number of clusters.

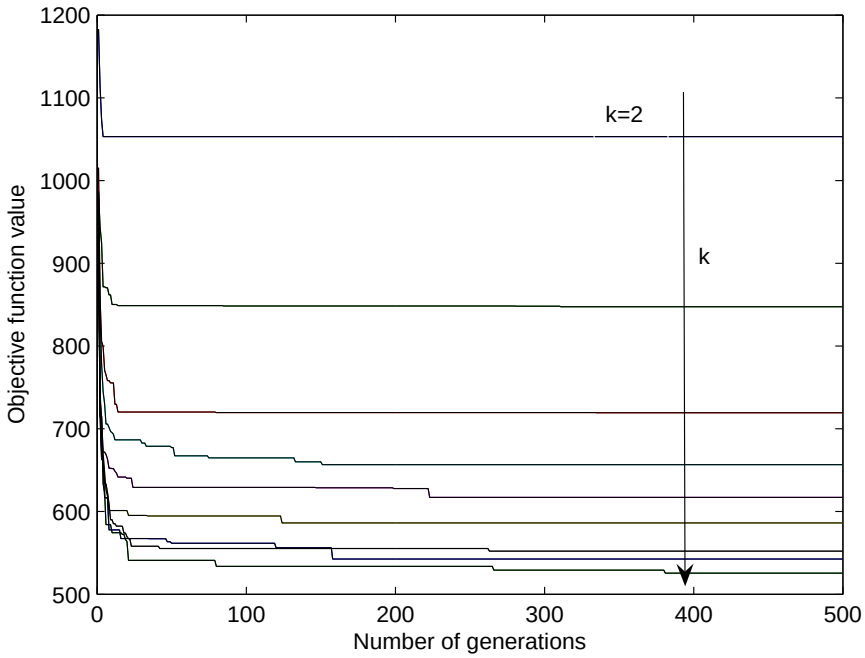

Fig. 5.13. Performance of GA in minimizing the objective function for Yeast diauxic shift data.

Since we don't have the 'true' solution in this case, we use *z-scores* proposed by Gibbons and Roth (2002) for evaluation of results. The *z-scores* of a partition indicates the enrichment of clusters with functionally related genes compared to the random partition. The higher the score, the better the partition. It uses, for Yeast, the Saccharomyces Genome Database (SGD) annotation of yeast genes with the gene ontology developed by Gene Ontology Consortium (Ashburner *et al.* (2000); Issel-Tarver *et al.* (2002)). Figure 5.14 shows the *z-scores* for this dataset as a function of number of clusters using proposed clustering method (solid line) and GK (dash line) and GG (dash-dot line). The large positive values of the *z-scores* indicate that the clusters are significantly enriched with functionally related genes than the random partitions. The *z-scores* for proposed clustering method span from 10.3 to 57.7. The z-scores range for GK clustering is 9 to 22, very low compared to other clustering methods. The z-score for GG clustering is 15.5 to 38.1. This indicates that the proposed method identifies clusters that are biologically significant.

5.4 Discussion and Conclusions

A clustering method is proposed to identify ellipsoidal clusters in gene expression data. The proposed method employs PCA on each cluster to and splits the space spanned by gene expression data into two subspaces. The distance of a gene to it's cluster centroid is calculated separately in both the spaces and then combined to get the total distance. The distance measured in PCA space captures the geometrical shape of clusters whereas the distance in residual space represents the thickness of cluster. The objective function for clustering is formed using this total distance and GA is used to optimize it. Two case studies with real gene expression data is used to validate the pro-

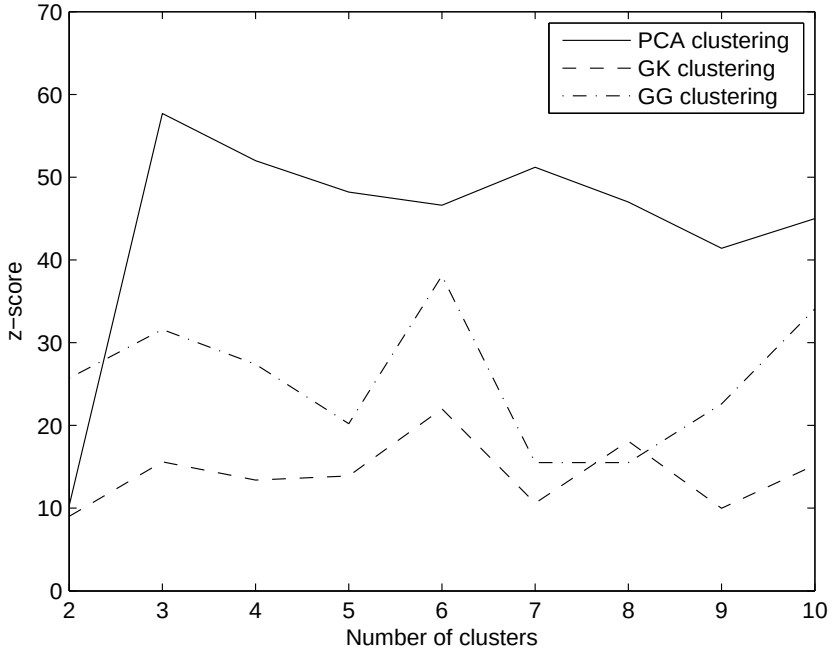

Fig. 5.14. Comparison of z-scores of proposed clustering method (solid line) with GK (dash line) and GG (dash-dot line) clustering methods for Yeast diauxic dataset.

posed method. In Case Study 1, the proposed method to identifies clusters which are very similar to the clusters reported by an expert. In Case Study 2, the proposed method identifies biologically significant clusters. The proposed method showed better performance when compared with other clustering methods.

Given that the actual 'shape' of clusters in gene expression data is unknown, it is essential to have methods with a general distance metric which is capable of identifying clusters of different shapes. Identification of ellipsoidal clusters is one step towards that goal. Methods that identify ellipsoidal clusters also identify spherical clusters as spherical cluster is a specific case of ellipsoidal cluster with equal eigenvalues for covariance matrix. However, singularity of covariance matrix hinders identification of ellipsoidal

clusters in gene expression data. The proposed method eliminates the problem using PCA and successfully identifies biologically significant clusters.

The proposed method uses a adaptive distance metric which is based on the idea of Soft Independent Method of Class Analogy (SIMCA) approach for pattern classification proposed by Wold (1976). The SIMCA approach develops a PCA model for each class in the training data with number of PCs appropriate for that class. Once the classier is built, new objects are classified to different class using the values T_r^2 and Q_r. The original SIMCA approach is for classification purpose where training data is available for model development. Here, the idea is extend for clustering with additional constraints in the cluster volume in both *PCA subspace* and *residual subspace*.

The constraint on cluster volume is necessary to get a non-trivial solution for clustering. Without this constraint, clusters can grow larger and the results in non-homogenous clusters. These constraints are implemented by multiplying T_{ijr}^2 and Q_{ijr} with $(\prod_{a=1}^{l_j} \lambda_a)^{(1/l_j)}$ and $(\prod_{a=l_j+1}^{p} \lambda_a)^{(p-l_j)}$, respectively. These two multipliers can also be seen as wightage factors for T_{ijr}^2 and Q_{ijr} for getting the distance matric (Equation 19). From this view, the weightage for Q_{ijr} is always smaller than the weightage for T_{ijr}^2 as the λ_i are arranged in descending order. This makes sense as the distance in *PCA subspace* is more important than the distance in *residual subspace* since *residual subspace* is spanned by non-dominant PCs.

Genetic Algorithms is used for minimizing the global distance objective function formed with proposed distance metric. From our results, it seems that GA is able to

identify the best possible minima. However, stochastic optimization techniques such as GA are computationally expensive. Along with this, the proposed distance metric requires to employ PCA on each cluster in each iteration. So the algorithm takes long time for large datasets with large number of clusters. For case study 1, the proposed method took approximately 20 mins for 100 iterations in $MATLAB^{TM}$ environment on Pentium 4 2.8 GHz Personal Computer with 1 GB of RAM. The time taken for the second case study is large (hours) as there are more genes. Development of deterministic optimization techniques may reduce the processing time.

6. EVOLUTIONARY APPROACH FOR FINDING NUMBER OF CLUSTERS IN MICROARRAY DATA

6.1 Introduction

Despite the widespread use of clustering algorithms in gene expression data analysis (Eisen *et al.*, 1998; Tavazoie *et al.*, 1999; Tamayo *et al.*, 1999; Yeung *et al.*, 2001; Dembele and Kastner, 2003; Sharan *et al.*, 2003), selection of clustering parameters continues to be a challenge. In many cases, the optimal specification of number of clusters, k, is difficult especially if there is inadequate biological understanding of the system (Jiang *et al.*, 2004). A suboptimal specification of number of clusters can generally result in misleading results — either all classes may not be identified or spurious classes may be generated (Bezdek and Pal, 1998). While the correct number of clusters can be identified by visual inspection in some cases, in most gene expression datasets, the data dimensions are too high for effective visualization. Hence, methods that find the optimal number of clusters are essential. Finding number of clusters is called as Cluster Validation (Halkidi *et al.*, 2001).

Several methods have been proposed for finding the number of clusters in data. The popular methods evaluate the partition using a metric and optimize it as a function of number of clusters. Comprehensive reviews of these methods are available elsewhere (Milligan and Cooper, 1985; Halkidi *et al.*, 2001). Here we briefly describe some recent methods recommended for gene expression data analysis. Tibshirani *et al.* (2001) proposed the gap statistic that measures the difference between within-cluster dispersion

and its expected value under the null hypothesis. The k that maximizes the difference is selected. Since the gap statistic uses within-cluster sum of squares around the cluster means to evaluate the within-cluster dispersion, this method is suitable for compact, well separated clusters. Dudoit and Fridlyand (2002) proposed a prediction based resampling method for finding the number of clusters. For each value of k, the original data is randomly divided into training and testing sets. The training data is used to build a predictor for predicting the class labels of the test set. The predicted class labels are compared to that obtained by clustering of test data using a similarity metric. This value is compared to that expected under an appropriate null distribution. The k for which the evidence of significance is the largest is selected. Ben-Hur *et al.* (2002) proposed a similar re-sampling approach where two random subsets (possibly overlapping) are selected from the data. The two random subsets are subsequently clustered independently and the similarity between the resulting partitions is measured for the common objects between two subsets. The distribution of this similarity (obtained from multiple runs) is visualized for each k and the optimal number of clusters is selected where transition from high to low similarity occurs. The approach of Dudoit and Fridlyand (2002) as well as Ben-Hur *et al.* (2002) assume that the sample subset can represent the inherent structure in the original data which may not be true for small clusters. Furthermore, the user has to manually locate the transition in Ben-Hur *et al.* (2002) approach.

Recently, Bolshakova and Azuaje (2003) employed Silhouette (Rousseeuw, 1987), Generalized Dunn's index (Bezdek and Pal, 1998), and Davies-Bouldin index (Davies and Bouldin, 1979) on gene expression data. These methods use the intra- and inter-clusters distances to identify the best partition (detailed description is given in Section

2.3). In general, cluster validation is easier when the underlying clusters are well separated. But, most cluster validation methods lead to suboptimal results when inter- and intra-cluster distances vary largely. To illustrate this, consider the artificial dataset in Figure 6.1 consisting of 600 objects in three clusters (A, B, and C). Clusters B and C are closer to each other and far from Cluster A. Figure 6.2 shows the results of Silhouette, normalized Dunn's and normalized Davies-Bouldin indices for this dataset. For ease of visualization, all indices have been min-max re-scaled to [0 1]. For a given index value $I_k(k = 1, 2, 3, ...k_{max})$, the re-scaled index value is obtained as

$$\hat{I_k} = \frac{I_k - min(I_k)}{max(I_k) - min(I_k)} \tag{6.1}$$

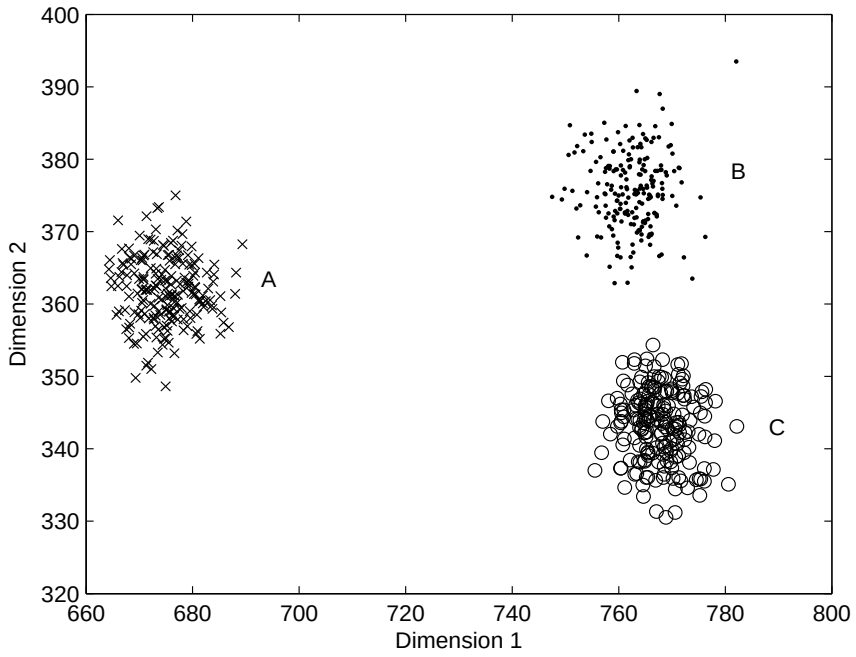

Fig. 6.1. Two dimensional artificial dataset with 3 inherent clusters (A, B, and C). Clusters B and C are closer to each other and far from Cluster A.

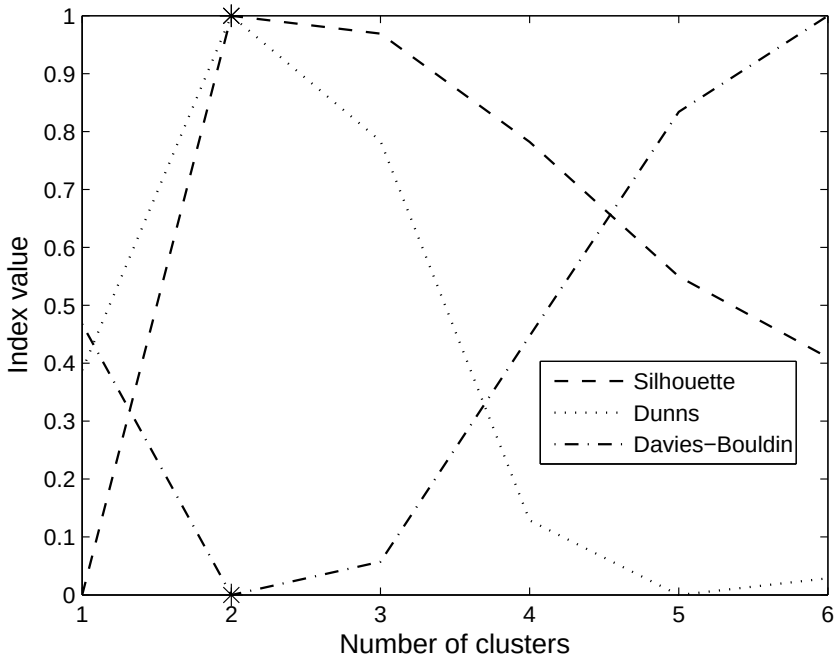

Fig. 6.2. Cluster validation results for the artificial dataset in Figure 6.1. All three indices, Silhouette (dash line), Dunn's (dot line), and Davies-Bouldin (dash-dot line) incorrectly predict 2 clusters although the underlying data can be seen to have 3 clusters (* indicates the optimal number of clusters predicted by specific index)

Silhouette, Generalized Dunn's index, and Davies-Bouldin indices incorrectly identified only 2 clusters in this dataset. A partition with two clusters $\{A\}$ and $\{B \cup C\}$ is more favorable according to intra- and inter-cluster distance based methods. Since the gene expression data contain clusters of varying inter- and intra-distances which are often intersecting and embedding in other clusters (Jiang *et al.*, 2003), the cluster validation methods based on intra- and inter-cluster distances are not suitable for gene expression data (Jonnalagadda and Srinivasan, 2004). This finding motivates development of new methods that do not rely on intra- and inter-cluster distances.

In this chapter, we propose a new method to find optimal number of clusters in the data. Our approach is based on an evolutionary view of the clustering process (Figure 6.3). We start by considering the whole dataset as a single cluster and notate it as Generation 1 (G_1). In each subsequent generation, the number of clusters, k, is incremented by one and the data re-clustered. A generation with k clusters is notated as G_k. The net change in the information content due to the addition of a cluster is measured using Net InFormation Transfer Index (NIFTI). NIFTI includes two components —*direction* of information change and *magnitude* of information change—in its calculation. The *direction* of change indicates whether information is gained or lost during evolution. The *magnitude* indicates the extent of change. During evolution, objects from i^{th} cluster, C_k^i, in the current generation, G_k, will be distributed across several clusters in the next generation, G_{k+1}. The clusters in G_{k+1} that receive objects from C_k^i are called as offspring of parent cluster C_k^i. NIFTI considers this rearrangement of cluster members when a new cluster is added for calculating the information change. The net information change is the sum of the information change for all parent clusters. Information increases if offspring clusters are separable. We use a simple but effective procedure with statistical basis to check the separability of offspring clusters. The *magnitude* of information change is calculated using information theory. This evolutionary procedure is carried out for a predefined number of generations (G_{max}). The Total Information Content, TIC, of a partition is defined as the cumulative information gained till that generation. A partition with the highest TIC is selected as the best partition. While testing for separability of clusters, NIFTI does not give weightage for largely separated clusters or penalize marginally separated clusters, thus eliminates the problems associated with varying inter-cluster distances.

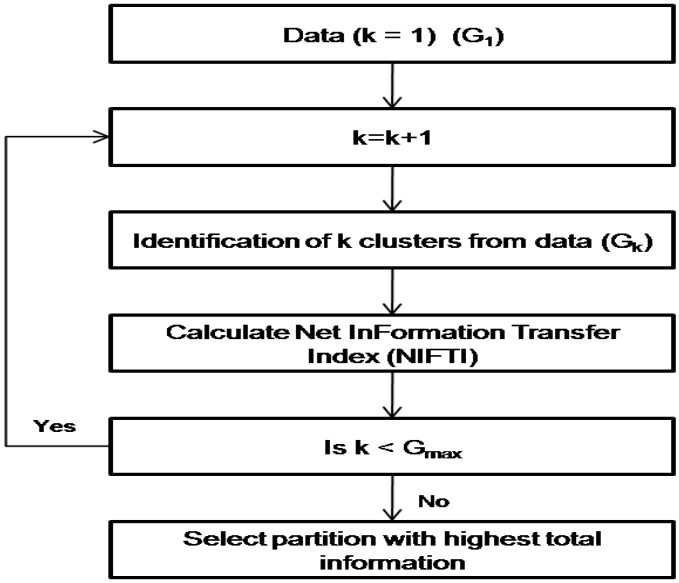

Fig. 6.3. Proposed cluster validation procedure. The procedure starts with unclustered data (G_1). In each subsequent generation, an additional cluster is added and the data reclustered. The Net InFormation Transfer calculated based on the evolution of objects during the generation. This procedure is carried out for a predefined number of generations (G_{max}). Finally the partition with highest total information is selected as the optimal partition.

6.2 Methods

Let $Z_{N\times m}$ be the dataset to be clustered containing N objects on which m features are measured. In gene expression data analysis, N is number of genes and m is number of assays. We use a clustering algorithm to generate a series of partitions from G_1 through G_{max} with an increment of one cluster in each generation. The migration of the objects during evolution from parent clusters in G_k to their offspring in G_{k+1} forms the basis for evaluating the quality of partition in G_{k+1}. Consider the migration of objects among clusters during evolution from G_k to G_{k+1} shown in Figure 6.4. Three scenarios are possible during evolution:

1. All objects in C_k^i may continue to be clustered together as a single cluster in G_{k+1}. We call this phenomenon as *cluster conservation*. Example: The cluster C_k^1 is conserved as C_{k+1}^1 with all objects intact.

2. Most members of C_k^i may stay together as a single cluster in G_{k+1}, but a few escape to other clusters. This phenomenon is termed as *cluster leakage*. Example: Out of 400 objects in cluster C_k^2 most stay together in C_{k+1}^2, and 15 leak to C_{k+1}^3

3. Members of C_k^i migrate to a small number ≥ 2 of clusters in G_{k+1} such that each recipient cluster receives a significant fraction of objects. This is called as *cluster disassociation*. Example: Cluster C_k^3 disassociates to C_{k+1}^3 and C_{k+1}^4

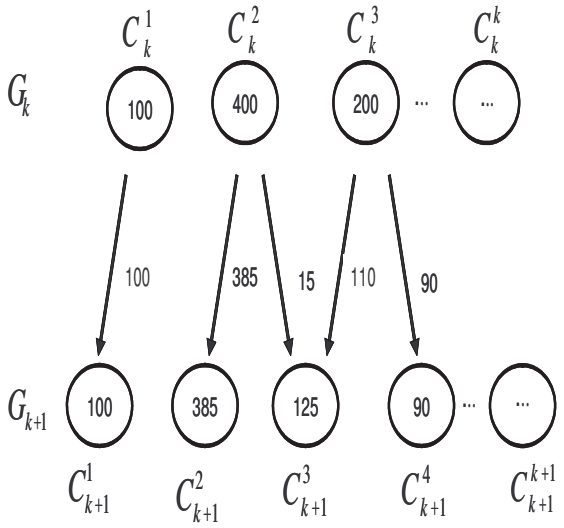

Fig. 6.4. Behavior of cluster members during evolution. A few clusters in G_k continue as single clusters in G_{k+1} while others disassociate or undergo leakage.

During evolution from G_k to G_{k+1}, some clusters are conserved, some disassociated, and others undergo leakage. The quality of the partition is measured in terms information transferred from G_k to G_{k+1} using the Net InFormation Transfer Index

(NIFTI). The TIC of partition is calculated for each generation as the sum of cumulative information transferred till that generation. The partition with the largest TIC is selected as the optimal one. The TIC for a partition at $(k+1)^{th}$ generation is given by:

$$TIC_{k+1} = TIC_k + NIFTI_{G_k \to G_{k+1}} \tag{6.2}$$

where $TIC_1 = 0$.

The optimal number of clusters is given by:

$$k_{optimal} = \underset{1 \le k \le k_{\max}}{\arg\max} TIC_k \tag{6.3}$$

6.2.1 Net InFormation Transfer Index (NIFTI)

The Net InFormation Transfer Index during evolution from G_k to G_{k+1} is defined as the sum of the information changes of all parent clusters weighted by the fraction of total objects they contain.

$$NIFTI_{G_k \to G_{k+1}} = \sum_{i}^{k} \frac{N_k^i}{N} \times g_k^i \tag{6.4}$$

where N_k^i is the number of objects in i^{th} parent cluster and g_k^i is its change in information as it evolves from G_k to G_{k+1}. Equation 6.4 is similar to the one used by Li *et al.* (2004) for calculating the information content of a partition.

The change in information of a parent cluster C_k^i is given by:

$$g_k^i = D_k^i \times M_k^i \tag{6.5}$$

D_k^i is the direction (gain or loss) and M_k^i the magnitude of information change arising from i^{th} parent cluster.

The objective of clustering is to identify clusters where objects within a cluster are more similar to each other compared to objects within other clusters. Geometrically, this means that clusters should be distant and separable from each other in the m dimensional feature space. Here, we propose a statistical test to check whether offspring clusters are separable or not. If the offspring of parent cluster are separable from other sibling, information is deemed to have been gained during transfer and D_k^i takes +1. In contrast, if offspring are not separable, information is deemed to be lost during transfer and D_k^i is –1. In contrast to other methods, the NIFTI is not weighted as per the inter- and intra-cluster distances.

The magnitude of information change, M_k^i, is calculated using Shannon entropy given by:

$$M_k^i = \sum_{j=1}^{r} -p_k^{ij} \ln p_k^{ij} \tag{6.6}$$

where r is the number of offspring and p^{ij} $(j = 1, 2, ..., r)$ is the fraction of objects that j^{th} offspring inherits.

As described before, during evolution from G_k to G_{k+1}, some clusters are conserved, some disassociated, and others undergo leakage. Consequently M_k^i is 0 for conservation, small for leakage, and large for cluster disassociation. Offspring clusters are tested using a separability test and NIFTI increases if they are separable and decreases otherwise. We propose a simple but effective test for separability of clusters. The cluster separability test is described below.

6.2.2 Test for separability of offspring

Though a parent cluster can result in many offspring, in practice it is observed that most members of a parent cluster migrate to a few proximal offspring. This is not a surprise since only one additional cluster is added at each step. Therefore, the incremental reorganization that takes place during evolution is minimal. We term those offspring which inherit large fractions of objects from a parent as the dominant offspring. The information transferred for a parent cluster can be approximated by considering only the dominant offspring. The information change arising from the other offspring (non-dominated) is very small and can be neglected. Hence, r in Equation 6.6 is set to 2 for all parent clusters.

Let X and Y be the two dominant offspring of a parent cluster given by:

$$X = \arg\max_j p^{ij} \tag{6.7}$$

$$Y = \arg\max_{j \neq X} p^{ij} \tag{6.8}$$

where p^{ij} is the fraction of objects migrated from i^{th} parent cluster, C_k^i to the j^{th} offspring cluster, C_{k+2}^j.

We use inter- and intra-cluster distances to identify whether X and Y are separable or not. X and Y are said to be separable if the distance between their centroid, δ_{XY}, is larger than the sum of their radii (Δ_X and Δ_Y). A variety of methods can be used to measure the cluster radius Bezdek and Pal (1998). Here, the mean distance between the cluster centroid to all members of that cluster is used for this purpose.

Radius of cluster X:

$$\Delta_X = \frac{1}{|X|} \sum_{x \in X} d(x, \bar{v}_X) \tag{6.9}$$

where $|X|$ is the number of objects in X, x represents the object in cluster X, d is the distance metric used for clustering, and $\bar{v}_X$ the centroid of the cluster. Similarly, the radius of cluster Y is given by:

$$\Delta_Y = \frac{1}{|Y|} \sum_{x \in Y} d(x, \bar{v}_Y) \tag{6.10}$$

The centroid distance between X and Y is the distance between their centroids given as:

$$\delta_{XY} = d(\bar{v}_X, \bar{v}_Y) \tag{6.11}$$

Hence, the separability of offspring of C_k^i notated as D_k^i is given by:

$$D_k^i = \begin{cases} +1 \quad if \quad \delta_{XY} \geq (\Delta_X + \Delta_Y) \\ -1 \quad if \quad \delta_{XY} < (\Delta_X + \Delta_Y) \end{cases} \tag{6.12}$$

Geometrically, the proposed procedure for finding the separability of clusters is equal to modeling each offspring clusters as a hyper-spheres with radii (Δ_X and Δ_Y)and check whether the hyper-spheres overlap. Statistically, this procedure is a hypothesis test with the following null and alternative hypotheses:

H_0 = Offspring clusters are part of single cluster

H_1 = Offspring clusters are different clusters

The equations for hypothesis testing are derived considering the situation where a single cluster is artificially broken into two clusters. Let us consider a single cluster C containing n objects. Assume that the data is drawn from Gaussian distribution with mean μ and covariance matrix Σ. Without loss of generality, we can assume that the mean is at origin and covariance matrix has only diagonal elements and off-diagonal elements are all zero (if the original covariance matrix contains non-zero off diagonal elements it can be converted to diagonal matrix by principal axis rotation). Suppose, now that we partition C into two clusters (offspring), we can reject the null hypothesis using the distribution functions of both centroid distance and radii of offspring clusters. There are two cases:

1. Same variance in all dimensions *i.e.*

$$\Sigma = \begin{vmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \dots \\ 0 & 0 & \dots & \sigma_m^2 \end{vmatrix}$$

and $\sigma_1^2 = \sigma_2^2 = ... = \sigma_m^2$.

2. The σ_is of Σ are different.

We derive the equations for proposed test of separability of offspring for case 1 and show how it can be extended to case 2.

Case 1: Geometrically, this means that the cluster of objects form a spheroid in m-dimensional space. Application of any clustering algorithm to partition this cluster into two offspring results in optimal (based on the objective function used for clustering) partition. If we know the analytical solution for that optimal partitioning, we could determine the distribution functions for centroid distance and radii of clusters. Lacking the analytical solution for the optimal partitioning, we cannot derive the actual sampling distributions. However, approximate estimates can be obtained by considering the suboptimal partition provided by a hyperplane through the centroid of parent cluster (Duda and Hart, 1973). This hyperplane approximation is schematically described in Figure 6.5 for two dimensional data. The data contains 1000 samples drawn from 2 dimensional Gaussian distribution with mean at origin and covariance matrix [1 0;0 1]. k-means clustering algorithm is used to generate the two partitions.

Because of the hyperplane, the centroids for individual offspring clusters will be same as centroid of original parent cluster except in one dimension (the dimension $\perp$ to hyperplane). Let the dimension $\perp$ to hyperplane be denoted as f. Then f follows half-normal distribution with mean $\sqrt{2/\pi}\,\sigma$ (Figure 6.5). So, the centroid distance between the two offspring is $2\sqrt{2/\pi}\,\sigma$. Considering the sample size, n, the squared centroid

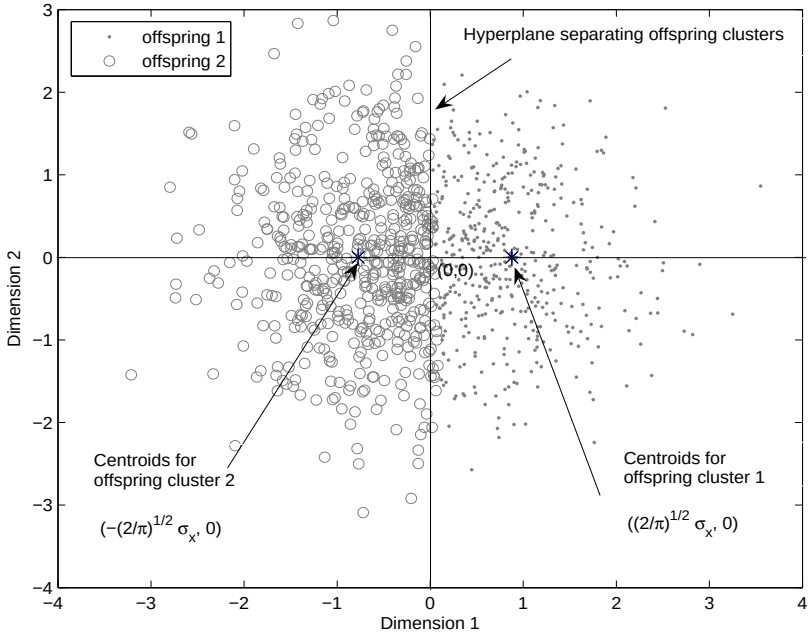

Fig. 6.5. Artificial partitioning of natural cluster

distance between the two offspring cluster follows Gaussian distribution with mean as $((n-1)/n)(8/\pi)\sigma^2$ and variance $2((n-1)/n^2)(64/\pi^2)\sigma^4$. The squared radius of cluster Δ^2 also follows a Gaussian distribution with mean $((m-2)/\pi)\sigma^2$ and variance $4((m-8)/\pi^2)\sigma^4$ (Duda and Hart, 1973).

Now consider the Equation 6.12 for testing the separability of offspring clusters.

$$\delta_{XY} \geq (\Delta_X + \Delta_Y) \tag{6.13}$$

Squaring both sides

$$\delta_{XY}^2 \geq (\Delta_X + \Delta_Y)^2 \tag{6.14}$$

Since the clusters are separated by a hyperplane passing through the origin, the two offspring clusters approximately contain same number of samples and hence ($\Delta_X \approx \Delta_Y = \Delta$).

Hence the test of separability of offspring clusters reduces to

$$\delta^2 \geq 4 \times \Delta^2 \tag{6.15}$$

where the subscripts X and Y have been removed for convenience. Hence, the offspring clusters are deemed to be separable if

$$h \geq 0 \tag{6.16}$$

where $h = \delta^2 - 4 \times \Delta^2$

Using the distributions for δ^2 and Δ^2 derived above the distribution for above equations can be obtained. This distribution refers to the null distribution for the proposed hypothesis test as this derivation is through artificial portioning of a single cluster. Hence, the null hypothesis can be rejected considering the distribution of above equation. Since, both δ^2 and Δ^2 follows Gaussian distribution, h follows a Gaussian distribution with mean as $4(\frac{n-1}{n})(4/\pi - m)\sigma^2$ and variance $\frac{2}{n}[\frac{64}{\pi^2} + 8(m - 8/\pi^2)]\sigma^4$.

The false discovery rate for rejecting the single cluster hypothesis can be calculated using the distribution of h. The false discovery rate is the probability of $h > 0$. The false discovery rate indicates the probability that a offspring of a single parent cluster are incorrectly deemed as two separable clusters. Table 6.1 shows the false discovery

rate for different sample sizes. The values given in parenthesis are the false discovery rates obtained by computational study with 1000 datasets with mean at origin and $\sigma^2 = 2$. The false discovery rates are very low even for small samples sizes. It clearly shows that the proposed cluster separability test is able to correctly identify the artificial break of natural clusters. When a natural clusters is artificially broken, NIFTI decreases. So, selecting a partition with highest NIFTI gives number of natural clusters in the data.

Case 2: Geometrically this means that the cluster form a ellipsoid in m-dimensional space. An Analytical solution is difficult for this case. However, it is possible to show that $\delta^2 - 4\Delta^2 \geq 0$ for many situations. Assuming that the hyperplane separating the two offspring cluster is $\perp$ to the dimension of largest variance, the δ^2 is given by: $8/\pi\sigma^2_{max}$. Similarly, Δ^2 is given $\sum_{i=1,i\neq j}^{m} \sigma_i^2 + (1 - 2/\pi)\sigma^2_{max}$ where j corresponds to the dimension of largest variance. Hence, the separability test $\delta^2 - 4\Delta^2$ is given by: $4\sigma^2_{max}[4/\pi - 1] - \sum_{i=1,i\neq j}^{m} \sigma_i^2$. This means the artificial partition of single cluster is detected by proposed separability criteria whenever the sum of variances in all directions (except the variance of largest direction) has value at least $0.275 \times \sigma^2_{max}$. Since this criteria is satisfied in most of the cases, the proposed test for separability works well even in this case. To check the performance of proposed separability test, we generated 1000 random datasets with 1000 samples each in 3-dimensional space with the largest variance as $\sigma^2_{max} = 3$ and other variances equal to 0.75. In all the datasets the proposed method correctly identified the partition of a single cluster.

Table 6.1
False discovery rate of cluster separability test

	False Discovery Rate			
Sample size(n)	m=2	m=3	m=4	m=5
25	0.0068 (0.199)	$8.53 \times 10^{-7}(0.021)$	$2.99 \times 10^{-11}(0.001)$	$6.66 \times 10^{-16}(0.002)$
50	$1.84 \times 10^{-4}(0.008)$	$2.44 \times 10^{-12}(0.002)$	0 (0)	0(0)
100	$1.81 \times 10^{-7}(0)$	0(0)	0(0)	0(0)

6.3 Results

Four publicly available microarray datasets are used to illustrate the performance of the proposed approach. The first two datasets are time-course datasets and other two datasets contain data from different samples.

Two different clustering techniques, namely k-means and model-based, are used for generating partition with different number of clusters. The distance metrics used for clustering are the same as those used by the data publishers *i.e* Pearson coefficient for first, third, and fourth case studies and standard correlation coefficient for the second dataset. In all the case studies, the maximum number of generations, G_{max} is selected as $G_{max} \leq \sqrt{N}$ (Pal and Bezdek, 1995).

6.3.1 Case Study 1 : Yeast cell-cycle data

The Yeast cell-cycle dataset was generated by Cho *et al.* (1998). Oligonucleotide microarrays were used to monitor the expression levels of all known and predicted Yeast genes during two cell-cycles. Expression levels were measured at 17 time points with a time period of 10 min. The aim of this experiment was to identify the cell-cycle controlled genes in Yeast. Cho *et al.* (1998) visually observed the highly variant genes for consistent periodicity during the cell-cycle and identified 384 genes. These 384 genes were classified into five classes—early G1, late G1, S, G2, and M phases—based on their peak expression.

The proposed method, NIFTI, correctly identifies five clusters in this dataset using k-means method (Figure 6.6). For comparison, the results for Silhouette, Dunn's, and Davies-Bouldin indices are shown in Figure 6.6. All three indices predict 4 clusters in this data. The reason is as follows. At $k = 4$, genes from S and G2 phases are combined into one cluster while those from Early G1, Late G1, and M phases are clustered correctly. These four clusters are well-separated. When the number of clusters is increased to 5, while S and G2 clusters are identified correctly, the inter-cluster distance is small. The three methods therefore identify the partition with four clusters as optimal. In contrast to these distance based methods, the proposed method gives no weightage for larger inter-cluster distances and correctly identifies 5 clusters.

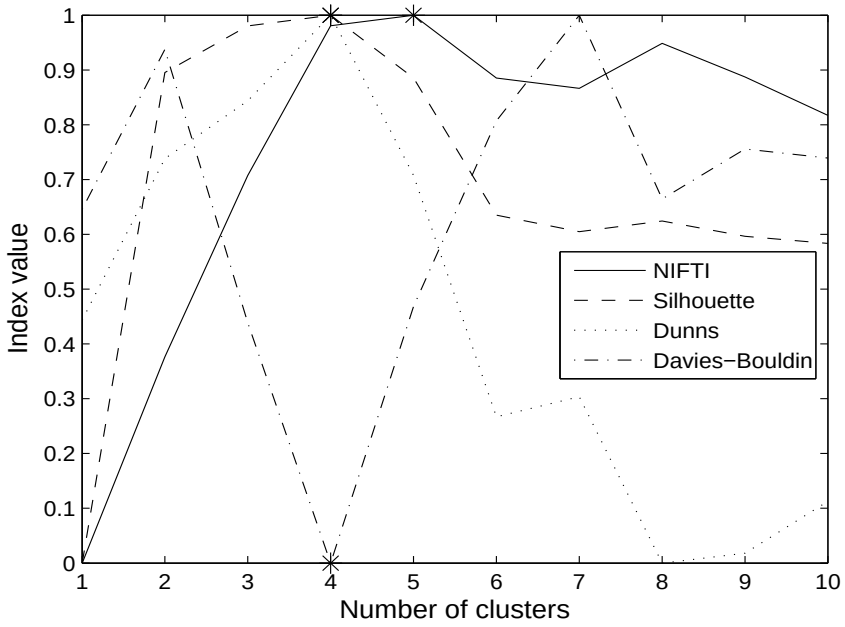

Fig. 6.6. Results for Yeast cell-cycle dataset using k-means clustering. NIFTI (solid line) correctly finds 5 clusters in this dataset. Silhouette (dash line), Dunn's (dot line), and Davies-Bouldin (dash-dot line) indices predict only 4 clusters.

The five clusters identified by k-means clustering correspond to the five phases of cell-cycle—early G1, late G1, S, G2, and M phases. For example, cluster 1 contains the cell-cycle regulated genes including PCL9, SIC1 and DNA replication genes CDC6 and CDC46 that are classified into early G1 by Cho *et al.* (1998). The mean expression profile of this cluster shows single peak during the early stage of G1 (Figure 6.7). Similarly, other clusters are also enriched with genes that are classified into one of the reported clusters and their mean expression profiles peak during the corresponding stages (Figure 6.7).

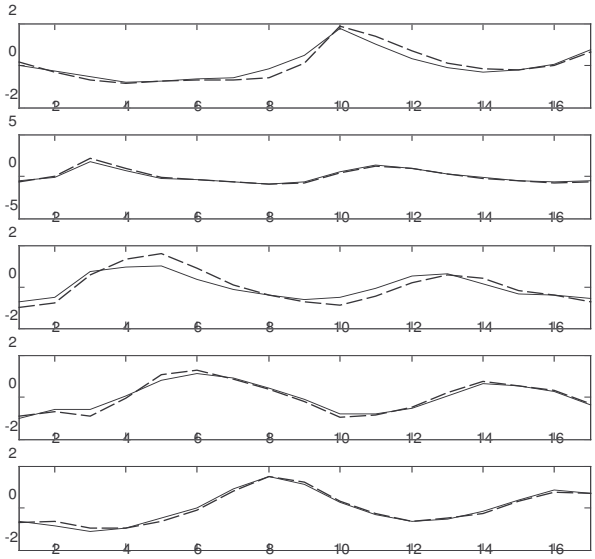

Fig. 6.7. Mean expression levels of Yeast cell-cycle clusters. Solid line represents the mean expression profile of clusters reported by Cho *et al.* (1998) and dash line corresponds to the optimal clusters from NIFTI. A strong similarity between the two can be observed.

However, some of the genes especially S phase genes are found to be 'mis-classified' by k-means clustering algorithm. To understand the discrepancy, we used Principal Component Analysis and plotted the scores with the first two dominant Principal Com-

ponents (Figure 6.8). From Figure 6.8, it is clear that some of the genes from reported classes, especially S phase genes, are distributed to other classes. The k-means algorithm put those genes in appropriate classes which explains the mismatch between the reported and k-means partitions.

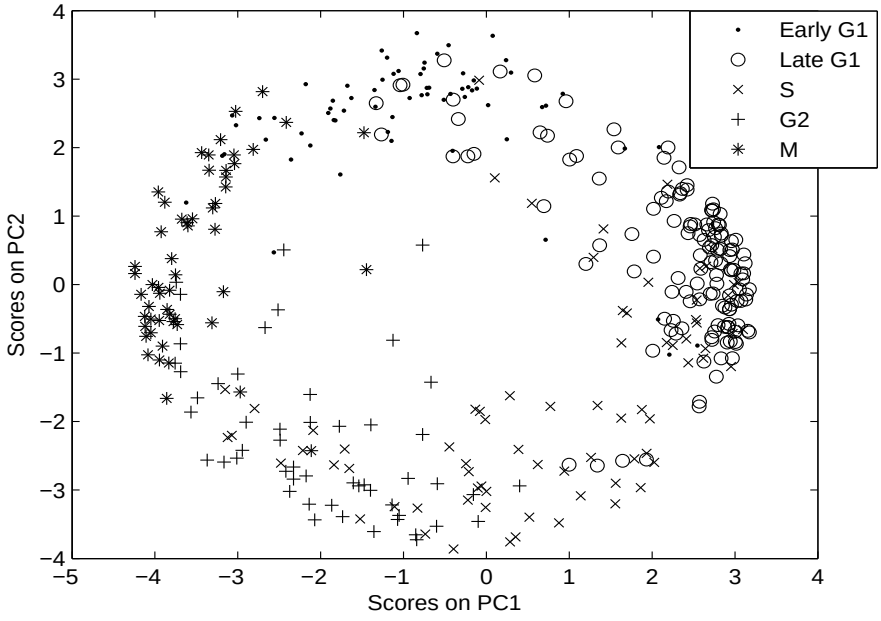

Fig. 6.8. Scores plot of Yeast cell-cycle dataset. The first two PCs capture 65% variance.

Results for this dataset using model-based clustering are shown in Figure 6.9. NIFTI correctly identifies 5 clusters using model-based clustering as well. Since the 'true' (reported) partition is available for this dataset, we compare the clustering results using k-means and model-based clustering with reported partition using Jaccard Coefficient (JC) (described in Section 5.3.2). JC takes a value between 0 (complete mismatch) and 1 (perfect match). The better the agreement between identified and the 'true' solution, the higher the value of JC. Figure 6.10 shows the JC for Yeast cell-cycle five phase criterion data as a function of number of clusters using k-means and model-based al-

gorithm. The JC takes a maximum value of 0.445 at $k = 5$ indicating that in the given range of k the extracted partition best matches with the reported one. This clearly shows that the 5 clusters identified using proposed method are correct.

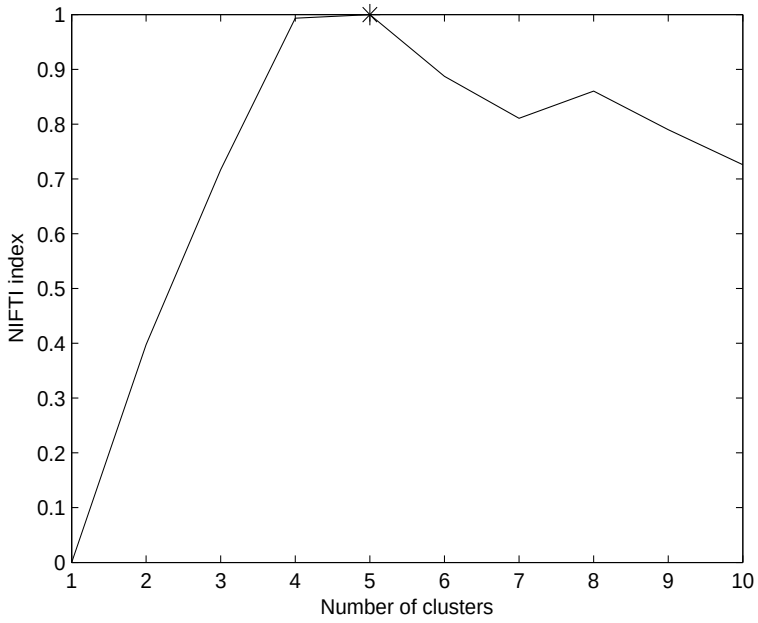

Fig. 6.9. Results for Yeast cell-cycle dataset using model-based clustering. NIFTI correctly finds 5 clusters in this dataset.

6.3.2 Case Study 2 : Serum data

The Serum gene expression dataset is reported by Iyer *et al.* (1999). In this study, the response of human fibroblasts to serum was measured using microarrays containing around 8000 probes. Filtering techniques were employed to short list 517 most variant genes.

NIFTI identifies 6 clusters in this dataset using k-means clustering (Figure 6.11). This result is supported by an other independent study using a graph-theoretical clus-

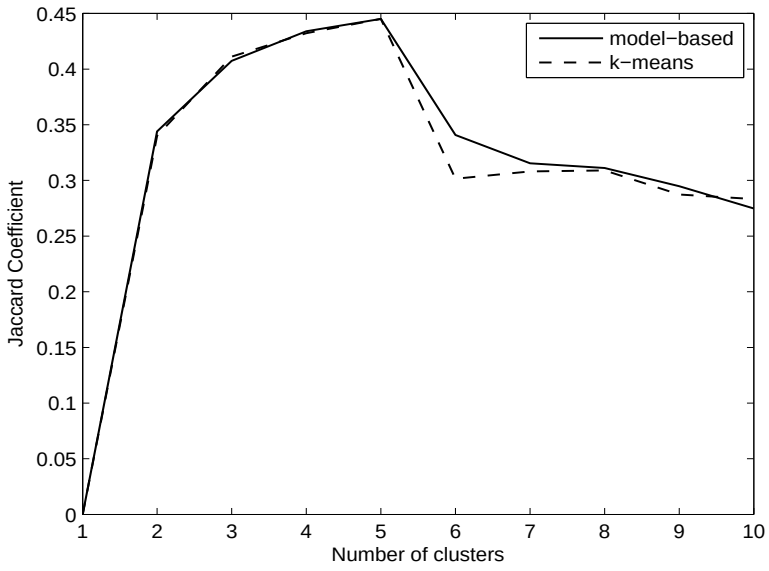

Fig. 6.10. Jaccard Coefficient for Yeast cell-cycle dataset. The JC has a maximum at $k = 5$ indicating that there are 5 clusters.

tering algorithm (Sharan *et al.*, 2003). The Silhouette, Dunn's and Davies-Bouldin indices identify only 2 clusters in the dataset (Figure 6.11). This dataset is more complex than the previous one. It contains two large clusters—one with up-regulated genes and another with down-regulated genes. All the other clusters are embedded in these large clusters. The ratio of difference between the intra- and inter-clusters distances is highest at $k = 2$. So any distance based method will generally identify only two clusters in this dataset. Multiple peaks observed for NIFTI index for this dataset while model-based clustering is used for generation of different clustering partitions (Figure 6.12). Though highest peak is at $k = 9$, the Jaccard Coefficient has the highest value at $k = 6$ (Figure 6.13) indicating 6 clusters in this dataset.

In the next two case studies, the datasets contain gene expression data from different cancer samples. In these datasets, samples are clustered based on their similarity in expression patterns. Model-based clustering is not suitable for these datasets as it uses

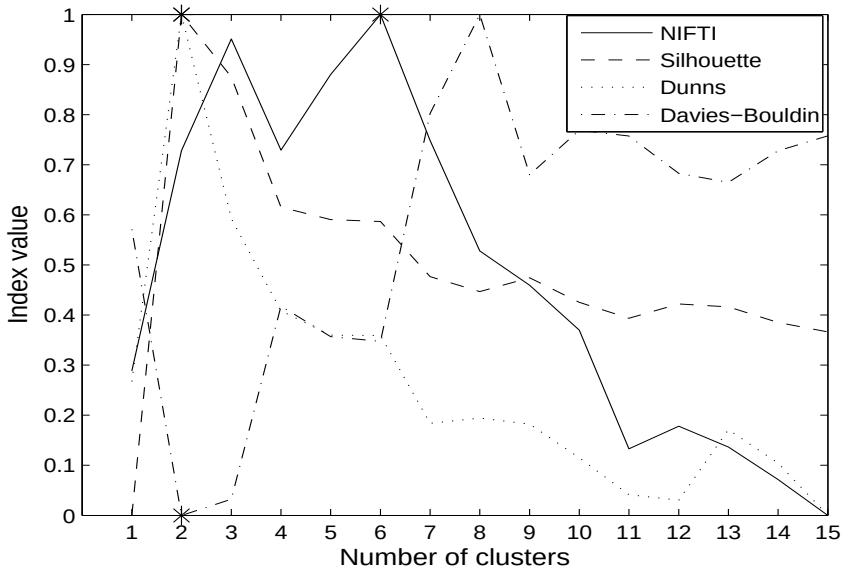

Fig. 6.11. Results for Serum dataset using k-means clustering. NIFTI (solid line) predicts 6 clusters. Silhouette (dash line), Dunn's (dot line), and Davies-Bouldin (dash-dot line) estimate only 2 clusters.

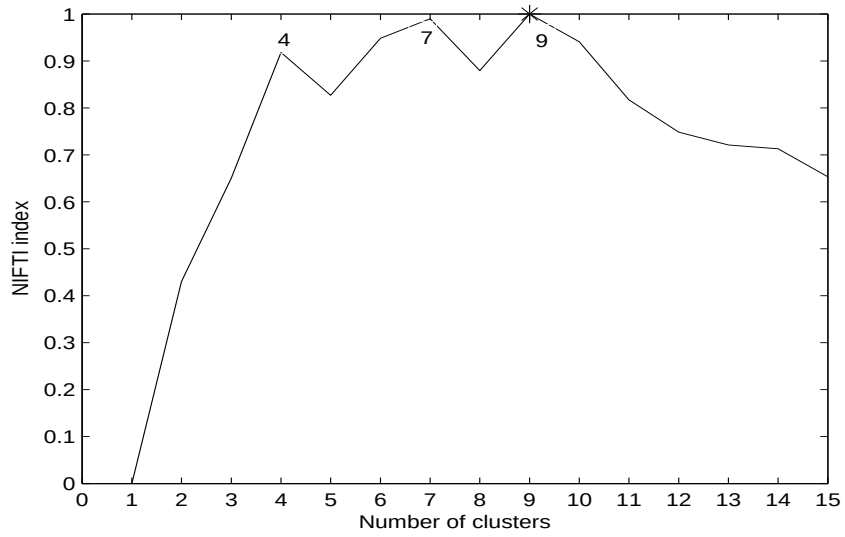

Fig. 6.12. Results for Serum dataset using model-based clustering. NIFTI index has multiple peaks with a maximum peak ak $k = 9$. However, the Jaccard coefficient between the partition from model-based clustering and expert partition has maximum at $k = 6$ (Figure 6.13).

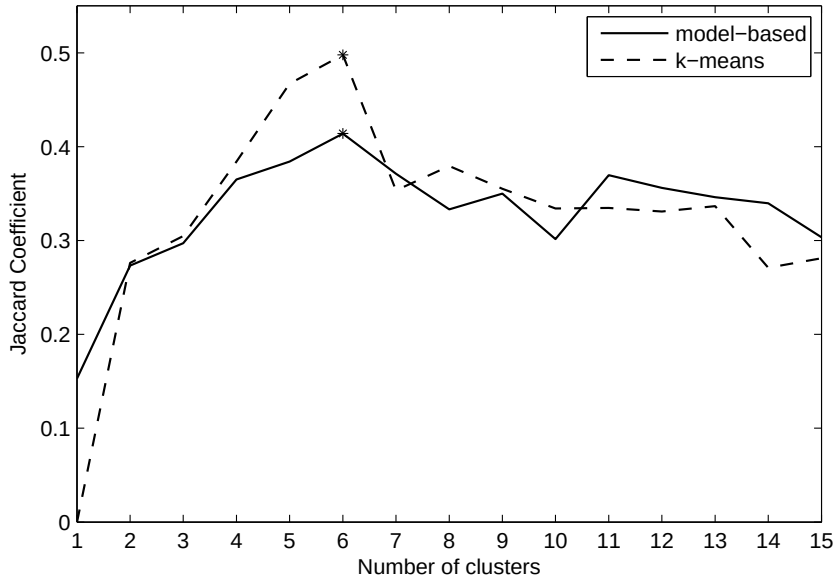

Fig. 6.13. Jaccard Coefficient for Serum dataset. The Jaccard Coefficient for Serum dataset has maximum at number of clusters $k = 6$ indicating that identifying 6 clusters is correct.

covariance matrix in its computation. The estimation of covariance matrix is inaccurate for sample clustering as the number of samples in each cluster are very small. So, results are given for only k-means clustering.

6.3.3 Case Study 3 : Lymphoma data

The lymphoma dataset was reported by Alizadeh *et al.* (2000). In this experiment, cDNA microarrays were used to characterize gene expression patterns in adult lymphoid malignancies. After filtering, the final dataset contain 4026 genes whose expression levels were measured using 96 arrays. The dataset comprises samples from three prevalent adult lymphoid malignancies - Diffuse Large B-cell Lymphoma (DLBCL), Follicular Lymphoma (FL), and Chronic Lymphocytic lymphoma (CLL). For comparison, the normal lymphocyte subpopulation under a variety of conditions is also

included. The objective of the study was to identify if the presence of malignancy and its type can be identified from gene expression patterns. Alizadeh *et al.* (2000) used hierarchical clustering for clustering the samples and identified two distinct subtypes of DLBCL-Germinal Center B-like DLBCL and Activated B-like DLBCL.

NIFTI finds 4 clusters in this dataset using k-means clustering algorithm with Pearson correlation as the distance metric (Figure 6.14). Not surprisingly, these four clusters correspond to the four distinct branches of the dendrogram reported in Alizadeh *et al.* (2000). Two of these clusters contain the samples from two subtypes of DLBCL namely germinal center B-like DLBCL and activated B-like DLBCL. The third cluster contains all FL and CLL samples along with the resting blood samples. Most of the cell-cycle control genes, checkpoint genes and DNA synthesis genes that are defined as 'proliferation signature' by Alizadeh *et al.* (2000) are under expressed in these samples. This makes these samples distinct from DLBCL samples in which the proliferation signature genes are up-regulated. The fourth cluster comprises the remaining normal lymphocyte subpopulation under different activation conditions. However, the transformed cell line samples which are grouped with other normal sub-populations by Alizadeh *et al.* (2000) are clustered with DLBCL samples by k-means. The over-expression of proliferation signature genes in these samples might be the reason that they appear 'closer' to DLBCL samples to k-means. Nevertheless, k-means clustering correctly clustered two out of the three DLBCL samples that were incorrectly clustered by the hierarchical clustering.

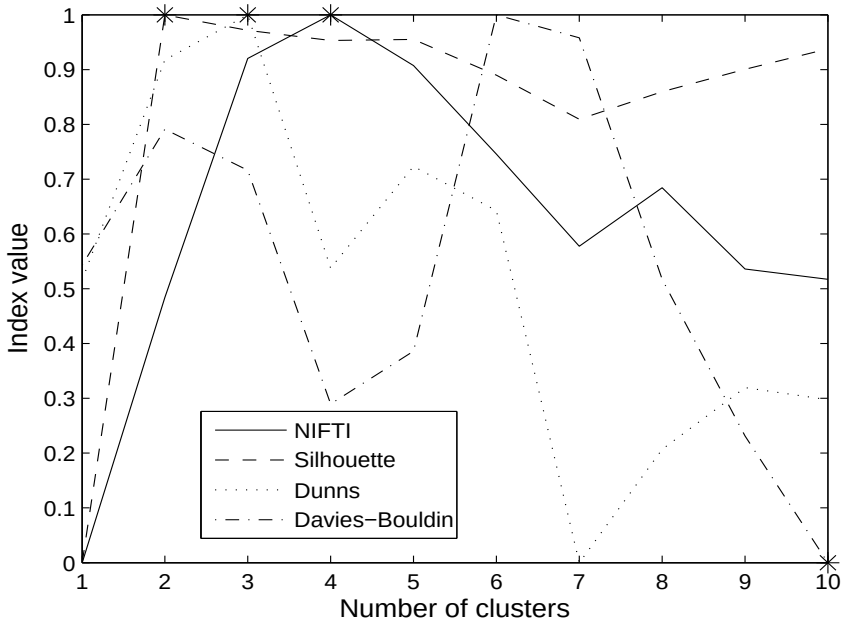

Fig. 6.14. Results for Lymphoma dataset. NIFTI (solid line) finds 4 clusters in this dataset. Silhouette (dash line) identifies 2 clusters. Dunn's (dot line) predicts 3 clusters. Davies-Bouldin (dash-dot line) predicts 4 clusters.

The Silhouette, Dunn's and Davies-Bouldin indices for this dataset are also shown in Figure 6.14. The Silhouette index estimates only 2 clusters and Dunn's index predicts 3. The lowest value of Davies-Bouldin index occurred at $k = 10$ in the range of k values tested (it continued to decrease further with increase of k). However, Davies-Bouldin index has a local minima at $k = 4$ indicating four clusters in this dataset. At $k = 2$, all DLBCL samples are grouped into one cluster and all other samples (FL, CLL, and normal) are lumped into other. At $k = 3$, the latter is split and normal samples are identified as the third cluster. This indicates that at $k = 2$ and $k = 3$ subclasses of DLBCL cannot be identified. Only at $k = 4$, the two subclasses of DLBCL are identified. This clearly shows the usefulness of proposed method to identify correct number of clusters that aids discovering novel sub-types of diseases.

6.3.4 Case Study 4 : Pancreas data

The Pancreas dataset used in this study was reported by Iacobuzio-Donahue *et al.* (2003). In this study, cDNA microarrays were used to analyze gene expression patterns in 14 pancreatic cell lines, 17 resected infiltrating pancreatic cancer tissues (two sub types), and 5 normal pancreases. The final filtered dataset consists of 1493 genes and 36 samples.

As shown in Figure 6.15, Silhouette, Dunn's, and Davies-Bouldin indices estimate 2 clusters for this dataset. A partition with two clusters lumps together the normal and pancreatic cancer tissues into a single cluster. The second cluster contains all the pancreatic cancer cell lines. NIFTI estimates four clusters in this data. A partition with four clusters describes this data well: all cancer cell line samples are accurately placed in one cluster, all normal samples are grouped together, and two different cancer tissues are well separated into two clusters. Only one sample was found to be mis-clustered. This partition with four clusters also exactly matches the dendrogram reported in Iacobuzio-Donahue *et al.* (2003).

6.4 Discussion and Conclusions

The use of clustering techniques in gene expression data analysis is increasing rapidly. To obtain the best results from these clustering techniques, optimal specification of the number of clusters is essential. Hence, methods that automatically identify the number of clusters in high-dimensional gene expression data have been proposed. Methods for finding the number of clusters in a dataset can be classified as global or lo-

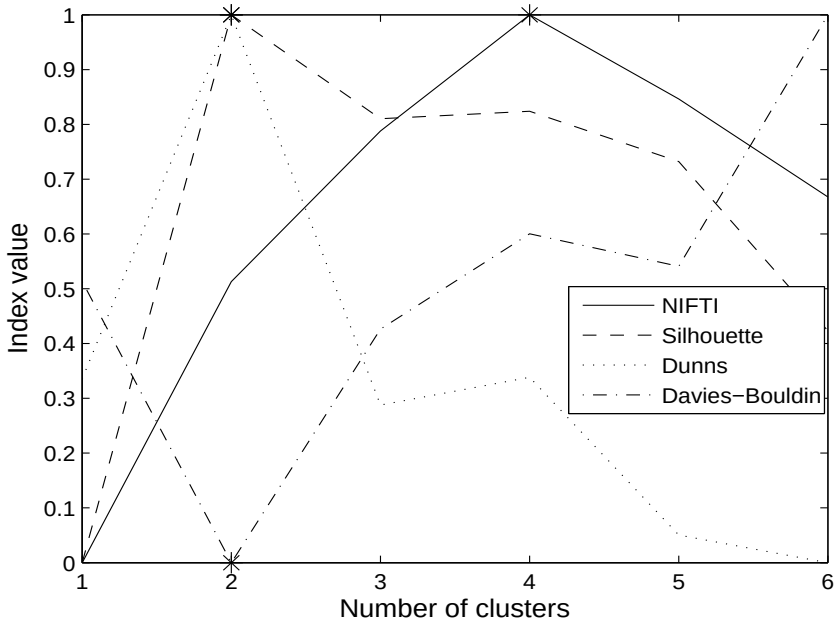

Fig. 6.15. Results for Pancreas dataset. NIFTI (solid line) finds 4 clusters in this dataset. Silhouette (dash line), Dunn's (dot line), and Davies-Bouldin (dash-dot line) indices predict only 2 clusters.

cal methods (Gordon, 1999). Global methods evaluate clustering results by calculating some measure over the entire dataset whereas local methods consider pairs of clusters and test whether they should be amalgamated. The disadvantage of the global methods is that there is no definition for the measure for $k = 1$, *i.e.*, the global methods do not provide any clue whether the data should be clustered or not. Since local methods consider pairs of clusters, they can be used to decide if data should be clustered. The disadvantage of local methods is that they need a threshold value or significance level to decide whether the clusters should be amalgamated. The proposed approach combines both local and global approaches. At the local level, offspring clusters are checked for overlap and this information is converted into a global index.

The well-known methods for finding the number of clusters use within-cluster dispersion and/or inter-cluster distances. These 'distance' based methods are generally suitable when clusters are compact and well-separated but fail when sub-clusters exist. Our approach overcomes this limitation by giving no extra weightage for larger inter-cluster distances. In our approach, clusters lose or gain information based on intersection with other clusters. The actual distance between the clusters is not taken into consideration. Furthermore, the cumulative way of measuring information content of a partition ensures that information increase as long as a non-intersecting cluster can be identified.

However, the proposed method has a limitation. It models clusters as hyper-spheres. Even though modeling clusters as hyper-spheres simplifies the task of finding cluster intersections, it may lead to incorrect results in case the clusters do not have a spherical shape. Nevertheless, this procedure consistently identified the 'correct' number of clusters suggesting, in part, the spherical shape of gene clusters.

Here, the proposed method is evaluated using k-means clustering algorithm with Pearson correlation as distance metric for the Yeast cell-cycle and lymphoma datasets. The standard correlation coefficient (dot product of normalized vectors) is used for the Serum dataset. These two metrics are bounded: the minimum and maximum distances are 0 and 2 respectively. On the other hand metrics such as Euclidean distance and Manhattan distance are unbounded. Hence, the affect of outliers will be high while estimating the cluster radii. This may lead to incorrect estimation of number of clusters. This can be overcome by suitably normalizing the data or selecting other ways to find

cluster radius that are less sensitive to outliers. Further study using various distance metrics and clustering techniques is needed to further evaluate the method.

Generally computational time is an important issue in determining the number of clusters. In this study, we used 100 replicates of k-means algorithm for all datasets. The time required for finding the best number of clusters is less than 10 minutes for all datasets on a $Pentium4$ with 2.8 GHz processor.

7. SIMILARITY IN PRINCIPAL COMPONENT SUBSPACES FOR DETERMINING DISTINCT CLUSTERS IN GENE EXPRESSION DATA

7.1 Introduction

In this chapter, a method for finding the maximum number of 'distinct' clusters in gene expression data is described. This method uses Principal Component Analysis (PCA) for measuring the distinctness of clusters in a given partition and marks the clusters as 'distinct' or 'indistinct'. This is transformed to an index using information theory. 'Distinct' clusters contribute positively to the index whereas the 'indistinct' clusters contribute negatively. The partition with highest value for the index contains maximum number of 'distinct' clusters. This method for finding number of clusters is the extension of NIFTI method described in Chapter 6. NIFTI models each cluster as hyper-sphere. The use of PCA eliminates the problem associated with shape of cluster. The proposed method has one more advantage compared to NIFTI. The NIFTI requires the series of partitions with number of clusters from $k = 1$ to $k = k_{max}$ as it uses the rearrangement of objects to measure the quality of partition. The method proposed in this chapter does not have such limitation. The proposed method can also be used to compare the quality of results from different clustering algorithms with the same number of clusters.

The proposed method for determining distinct clusters is based on the definition of cluster: objects within the cluster are similar to one another (homogeneity) while be-

ing dissimilar to objects in other clusters (separation or distinctness). A partition with distinct, homogenous clusters is preferable to other partitions. Here, we use PCA similarity factor, S^{λ}_{PCA}, to identify such partition.

Initially a number of candidate partitions are generated, for example, by using different clustering techniques and/or by specifying different number of clusters, k, in each partition. Then the similarity between all the pairs of clusters in each partition is calculated using S^{λ}_{PCA} and the 'distinct' clusters are identified. A 'distinct' cluster shows low similarity to all other clusters in that partition whereas an 'indistinct' cluster shows high similarity to at least one of the other clusters. This information is summarized into an index called Net Principal Subspace Information (NEPSI) Index. 'Distinct' clusters contribute positively to this index whereas 'indistinct' clusters contribute negatively. A higher value of the index indicates the higher contribution of 'distinct' clusters over 'indistinct' clusters; hence the partition with highest NEPSI Index is selected as the 'best' partition.

7.2 Methods

7.2.1 Principal Components Analysis and S^{λ}_{PCA}

Here, we use PCA based similarity factor, S^{λ}_{PCA}, to measure the degree of similarity between two clusters. In this section, we briefly describe the PCA and S^{λ}_{PCA}. PCA is a multivariate technique that finds the principal components (directions) of variability in the data, and transforms the related variables into a set of uncorrelated ones. These principal components (PCs) are the linear combinations of original variables (Jackson,

1991). PCA is traditionally used to reduce the dimensionality of data.

Let $X_{N\times m}$ be the dataset containing N observations on m variables, $\mathbf{v} = \{v_1, v_2...v_m\}$. In gene expression dataset, observations (rows) are the genes and the variables (columns) are the assays. Each element, x_{ij}, is the expression level of i^{th} gene in j^{th} assay. Given the matrix X, the objective of PCA is to find a new set of uncorrelated variables, $z_j (j \leq m)$. The first PC z_1 is given by:

$$z_1 = w_{11}v_1 + w_{12}v_2 + ... + w_{1m}v_m \tag{7.1}$$

The coefficients, $\mathbf{w_{ij}} = \{w_{11}, w_{12}, ..., w_{1m}\}$ are selected such that the variance of the first PC is greatest while satisfying the constraint $\mathbf{w_1} \cdot \mathbf{w_1}' = \{w_{11}^2 + w_{12}^2 + ... + w_{1m}^2\} = 1$. The second PC, $z_2 = \mathbf{w_2} \cdot \mathbf{v}'$, has the greatest variance satisfying the two conditions:$\mathbf{w_2} \cdot \mathbf{w_2}' = 1$ and $\mathbf{w_2} \cdot \mathbf{w_1}' = 0$ (so that PCs are uncorrelated). Similarly, the j^{th} PC, $z_j = \mathbf{w_j} \cdot \mathbf{v}'$, has the greatest variance satisfying $\mathbf{w_j} \cdot \mathbf{w_j}' = 1$ and $\mathbf{w_j} \cdot \mathbf{w_i}' = 0$ $(\forall i \leq j, i \neq j)$. The variance of z_j is given by:

$$var(z_j) = \mathbf{w_j} \cdot \mathbf{S} \cdot \mathbf{w_j'} \tag{7.2}$$

where S is the covariance matrix of the original variables. S is given by:

$$\mathbf{S} = \frac{X^T X}{N-1} \tag{7.3}$$

provided X is column-mean centered. The solution of $\mathbf{w_j}$ that maximizes the variance of z_j is the eigenvector of $\mathbf{S}$ corresponding to the j^{th} largest eigenvalue of $\mathbf{S}$. The eigenvalues of $\mathbf{S}$ are the roots of the equation

$$|\mathbf{S} - \lambda_j \mathbf{I}| = 0 \tag{7.4}$$

If $\lambda_1, \lambda_2, ...\lambda_m$ are the eigenvalues of $\mathbf{S}$ such that $\lambda_1 \geq \lambda_2 \geq \lambda_3 ... \lambda_{m-1} \geq \lambda_m$, then $\mathbf{w_j}$ are the eigenvectors of $\mathbf{S}$ obtained by the solution of the equations

$$|\mathbf{S} - \lambda_j \mathbf{I}|\mathbf{w_j'} = 0 \tag{7.5}$$

The first few PCs capture most of the variance in the data whereas the remaining PCs represent noise. The number of PCs, l, generally used to reconstruct the data is much smaller than the original number of variables. In this work, l is chosen such that the l PCs describes at least 93% of the total variance in each dataset. The degree of similarity between two clusters with the same number of variables can be estimated by comparing their PCs (Krzanowski, 1979). The similarity factor S_{PCA} measures the similarity between the clusters based on the angles between the space spanned by the l PCs. Let A and B be two clusters with m variables. PCA transformation is carried out on the clusters separately and their first l PCs are obtained. The similarity between the two clusters is quantified by comparing their subspaces L and M, which are the eigenvector matrices corresponding to the first l PCs, and is given by:

$$S_{PCA}(A, B) = \frac{1}{N} \sum_{i=1}^{l} \sum_{j=1}^{l} \cos^2 \theta_{ij} \tag{7.6}$$

where θ_{ij} is the angle between the i^{th} eigenvector of L and the j^{th} eigenvector of M. The similarity can also be expressed as (Krzanowski, 1979)

$$S_{PCA}(A,B) = \frac{trace(L'MM'L)}{l} \tag{7.7}$$

Note that while the two clusters are required to have the same number of variables (assays), they could have different number of observations (genes). Smaller values of S_{PCA} indicate the low similarity between the groups whereas values larger than a pre-specified threshold signify high similarity.

The advantage of S_{PCA} is that it quantifies the similarity between two groups by a single number. One shortcoming of S_{PCA} is that it gives equal weight for all the PCs used in its calculation while the amount of variation captured by them varies largely. To overcome this limitation, Singhal and Seborg. (2002) developed a modified form of S_{PCA}, denoted as S^{λ}_{PCA}, that weighs the PCs by the variation captured by them. S^{λ}_{PCA} is defined as:

$$S^{\lambda}_{PCA}(A,B) = \frac{\sum_{i=1}^{l}\sum_{j=1}^{l}\lambda_i^A\lambda_j^B\cos^2\theta_{ij}}{\sum_{i}^{l}\lambda_i^A\lambda_i^B} \tag{7.8}$$

where λ_i^A and λ_j^B are the i^{th} and j^{th} largest eigenvalues (variances of i^{th} and j^{th} PCs) of the covariance matrices of A and B, respectively. The S^{λ}_{PCA} takes values between [0 1]; a value close to 0 indicates that A and B are dissimilar while a value close to 1 indicates that they are similar (Srinivasan *et al.*, 2004).

The expression profile of a gene is a point in m-dimensional assay space. A group of genes form a cluster of points. Application of PCA on a cluster identifies the directions of largest variations in that cluster and rotates the axes to these principal directions. If only the first l PCs are used to represent a cluster, the points can be considered as embedded in the l-dimensional PC subspace of the m-dimensional original space (Krzanowski, 1979). The PC sub-space thus captures the structure of the cluster of points. The S^{λ}_{PCA} measures the similarity between the sub-spaces embedding two gene clusters by measuring the angle between the sub-spaces. When the structures of the two clusters are similar, their PC subspaces coincide and $S^{\lambda}_{PCA} \approx 1$.

In gene expression data clustering, genes are generally grouped together based on the 'shape' of their expression profile *i.e.* genes are grouped into the same cluster even if their expression profiles differ in magnitude. In many applications, genes with inversely correlated expression profiles are also grouped together; this is achieved by using the squared Pearson correlation metric during clustering. Since S^{λ}_{PCA} measures the angles between the PCs of the two groups, it effectively discounts differences in their magnitudes and instead measures the similarity of the correlations (positive or negative) between the constituent genes. It is therefore a suitable measure for finding similarity or distinctness of gene clusters.

7.2.2 Calculation of NEPSI Index

The *Net Principal Subspace Information (NEPSI) Index* for a given partition is derived from Information theory. Information theory measures the information contained in a message selected from a set of possible messages. If the message is certain or

known a priori, *i.e.* the set contains only one message, the information content is zero. The more uncertain (large set of possible distinct messages) a message the more the information content of the message. Shannon defined the entropy that measures the average information contained in a single observation of a random variable (Shannon, 1948). Let the random variable R takes the values $\{r_1, r_2, ...r_t\}$ with the probabilities $\{p(r_1), p(r_2), ..., p(r_t)\}$. Shannon entropy of R is given by:

$$E(R) = -\sum_{r} p(r) \log p(r) \tag{7.9}$$

While Shannon entropy was originally developed for communication technology, it is widely used in a variety of applications including gene expression data analysis (Fuhrman *et al.*, 2000). Here, we use Shannon entropy to measure the information content of a partition.

The information content of a partition $C = \{C_1, C_2, .., C_i, ..C_k\}$ with k clusters can be defined as (Li *et al.*, 2004):

$$E(C) = \sum_{i=1}^{k} p_i E(C_i) \tag{7.10}$$

where p_i is the probability and $E(C_i)$ is the entropy of cluster C_i. The probability of a cluster is given by:

$$p_i = \frac{|C_i|}{N} \tag{7.11}$$

where $|C_i|$ is the number of genes in cluster C_i.

Suppose each cluster C_i follows the m-dimensional Gaussian distribution with covariance matrix Σ_i . Then $E(C_i)$ is given by:

$$E(C_i) = \log(2\pi e)^{m/2} + 1/2 \log |\Sigma_i| \tag{7.12}$$

Substituting this in Equation 7.10 and discarding the constant term $\log(2\pi e)^{m/2}$ gives the expression for $E(C)$ as

$$E(C) = \frac{1}{2} \sum_{i=1}^{k} p_i \log |\Sigma_i| \tag{7.13}$$

Shannon entropy requires the messages in the set to be distinct. Adding a message that models the existing one does not increase the entropy. This can be accounted for by including the similarity between the clusters. We incorporate the 'distinctness' of the clusters, d_i, into the Shannon entropy. We call this measure as NEPSI Index defined as:

$$NEPSI_k = \sum_{i=1}^{k} d_i \cdot p_i \log |\Sigma_i| \tag{7.14}$$

Whenever there are highly similar clusters, the similarity metric S^{λ}_{PCA} identifies them as 'indistinct' clusters and the information contribution of those clusters is negative.

The crucial step in the proposed method is the identification of 'distinct' clusters in the given partition. As described previously, we use the PCA based similarity metric S^{λ}_{PCA} for this purpose.

Let $C = \{C_1, C_2, C_3, ...C_k\}$ be a partition with k clusters. Each cluster C_i is compared to other clusters $C_j (i \neq j)$. Cluster C_i is said to be 'distinct' if it shows lower similarity than a predefined similarity threshold, θ_T, to all other clusters in that partition. The distinctness of information contribution from this cluster, d_i, is given by:

$$d_i = \left\{ \begin{array}{l} +1 \, if \, \max \{S^{\lambda}_{PCA}(C_i, C_j) \forall j \neq i\} \leq \theta_T \\ -1 \, if \, \max \{S^{\lambda}_{PCA}(C_i, C_j) \forall j \neq i\} > \theta_T \end{array} \right\} \tag{7.15}$$

S^{λ}_{PCA} is symmetric, *i.e.* $S^{\lambda}_{PCA}(C_i, C_j) = S^{\lambda}_{PCA}(C_j, C_i)$. Hence, while finding similarities among clusters, we need to find only $\left(\begin{array}{c} k \\ \\ 2 \end{array} \right)$ similarities.

Next, we describe how maximizing NEPSI identifies the partition with most 'distinct' clusters. Let C_{opt} be the best partition with k_{opt} distinct clusters. Three different scenarios are possible:

1. Number of clusters k in a partition is greater than k_{opt} : some 'natural' clusters will be split into two or more clusters in this case. These offspring clusters will generally show high similarity to each other. The proposed method identifies these clusters as 'indistinct' clusters since S^{λ}_{PCA} will be high $(> \theta_T)$. Therefore $d_i = -1$ and these clusters contribute negatively to NEPSI Index. So $NEPSI_k < NEPSI_{k_{opt}}$ for $k > k_{opt}$.

2. The number of clusters in a partition $k < k_{opt}$: Some or all of the k clusters may be 'distinct'. Even if all of them are 'distinct' $NEPSI_k < NEPSI_{k_{opt}}$ since the Shannon entropy (equation 7.14) increases with k as long as additional 'distinct' clusters $(d_i = +1)$ can be found in other partition as will be the case with C_{opt}.

3. When $k = k_{opt}$, the partition contains the largest number of 'distinct' clusters. So NEPSI$_{kopt}$ is the largest.

The proposed method uses one parameter, θ_T for identifying distinct clusters. We have used $\theta_T = 0.55$ in all cases. The procedure for selecting threshold is as follows: we collected twenty five reported gene clusters (not from the datasets used for evaluation of proposed method) from publicly available datasets. For these 25 'training' clusters, we measured the similarity between all possible distinct pairs using S^{λ}_{PCA}. Then we artificially split each cluster into two sub-clusters (to generate 50 'indistinct' clusters) and found the similarity between them. Histograms of similarity scores for 'distinct' (shaded) and 'indistinct' (plain) clusters are shown in Figure 7.1. The similarity scores for 'distinct' clusters spanned the range from [0.1 0.6] whereas those of 'indistinct' clusters spanned [0.5 0.85]. The 'distinct' and 'indistinct' clusters can be separated in the range from [0.5 0.6]. Hence we selected $\theta_T = 0.55$

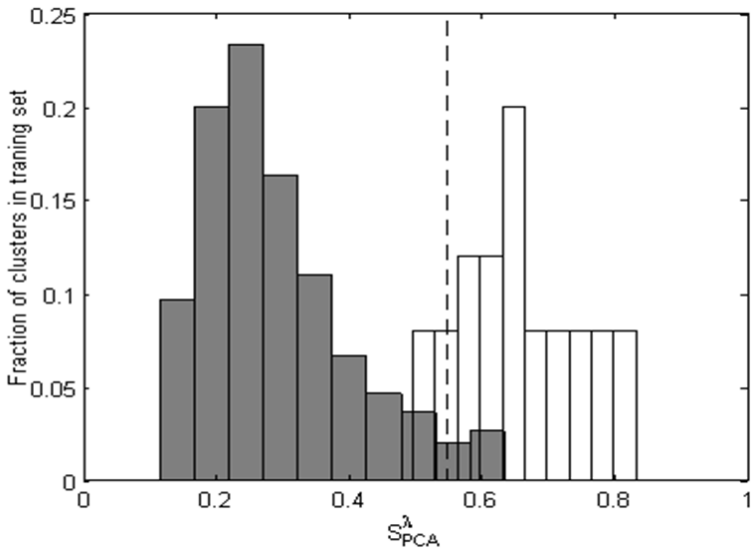

Fig. 7.1. Histograms of similarity scores for distinct (shaded) and indistinct (plain) clusters. Distinct clusters show a low similarity whereas indistinct clusters show high similarity.

7.3 Results

The NEPSI method for finding number of distinct clusters in gene expression data is evaluated using two gene expression datasets. The two datasets are standardized, *i.e.* the mean and standard deviation for each expression profile is set to zero and one respectively before clustering. The similarity metric used for clustering is Pearson correlation. Two different clustering techniques—k-means and model-based— are used for generation of partitions. More details of model-based clustering are available in Yeung *et al.* (2001). Specifically, we use the equal volume spherical (EI) models of Banfield and Raftery (1993) as implemented in the $Matlab^{TM}$ toolbox by Fraley and Raftery (1999). The toolbox also implements the Bayesian Information Criterion (BIC) to find the number of clusters. BIC compares two models using the fitness (likelihood) and the number of parameters. The higher the BIC score, the better the model. A score difference ≥ 10 is considered to be sufficient to say that a model is more favorable than others (Kass and Raftery, 1995).

7.3.1 Case Study 1: Yeast cell-cycle five-phase criterion dataset

This Yeast cell-cycle five-phase criterion dataset is described in Section 6.3.1. Results for Yeast cell-cycle five-phase criterion data are shown in Figure 7.2. The NEPSI Index correctly finds five distinct clusters using both k-means and model-based (EI) clustering algorithms. The BIC score shows no maxima within the given range of k (Figure 7.3). For all k values in this range, $BIC(k+1) - BIC(k) > 10$ indicating that there is no optimal number of clusters. However, BIC score shows inflection at $k = 4$ (visual observation) suggesting incorrectly 4 clusters in the data. The Silhouette

index, Dunn's index, and Davies-Bouldin index also identify only 4 clusters in this data (Figure 6.6). This clearly shows the efficacy of the proposed method in identifying the number of 'distinct' clusters in the data.

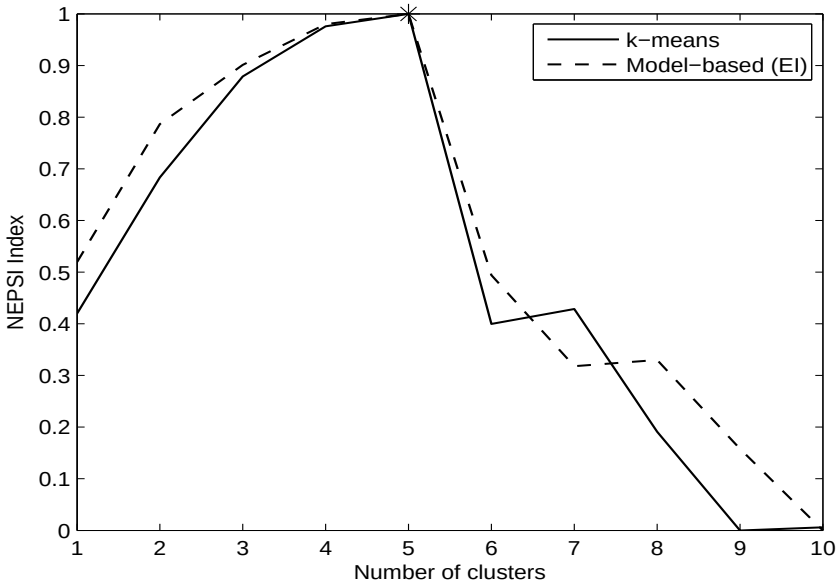

Fig. 7.2. Results for Yeast cell-cycle five-phase criterion data. The NEPSI index correctly finds 5 distinct clusters using both k-means and model-based (EI) clustering algorithms.

The JC for Yeast cell-cycle five phase criterion data as a function of number of clusters using k-means algorithm is shown in Figure 6.10. The JC takes a maximum value of 0.445 at number of clusters, $k = 5$ indicating that the extracted partition best matches with the reported one in the given range of k.

Table 7.1 shows the distribution of genes in the reported clusters over the 5 'distinct' clusters extracted by k-means. Rows represent the k-means clusters and columns the reported clusters. Three of the reported clusters (Early G1, Late G1, and M) are

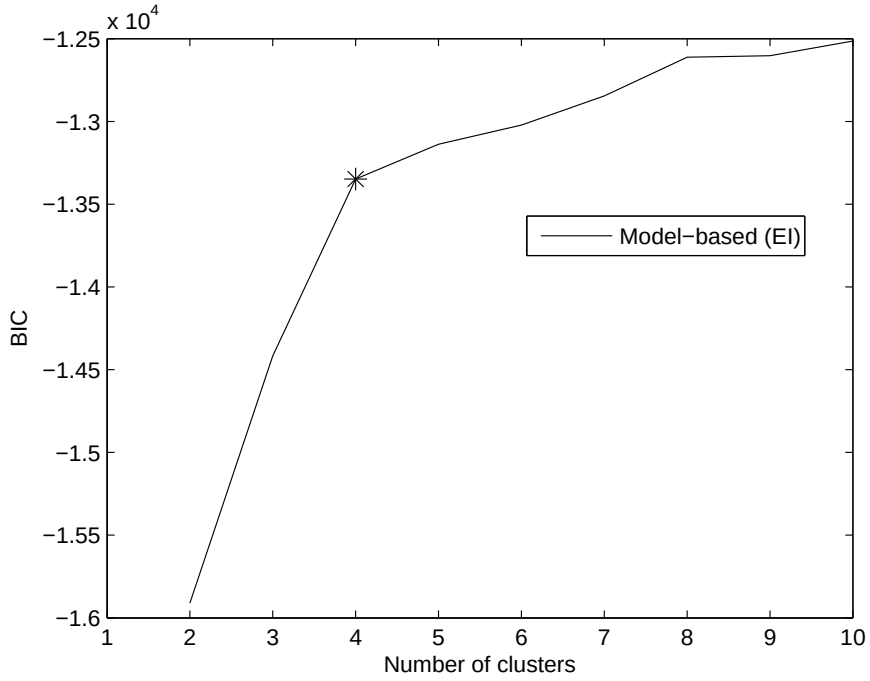

Fig. 7.3. Results for Yeast cell-cycle five-phase criterion data. BIC incorrectly reports 4 clusters with model-based (EI) clustering.

accurately identified by k-means. Some of the genes from other clusters (S and G2) are misclassified. To find the reason for this, we calculated the *average homogeneity (H_{avg})* and *average separation (S_{avg})* of both the reported and identified partitions as described in Sharan *et al.* (2003). The H_{avg} and S_{avg} are the average similarity of all the pairs of genes in the same and different clusters, respectively. A partition with high H_{avg} and low S_{avg} is preferable. The results are shown in Table 7.2. The H_{avg} and S_{avg} for the identified partition are significantly better than the reported partition. The homogeneity values of the S and G2 reported clusters are found to be 0.3316 and 0.4363, respectively. The low homogeneity of these clusters indicates that some genes in these clusters are not similar to other genes within the same clusters. This finding is also supported by another independent study (Lukashin and Fuchs, 2001). The clustering

algorithms used here might have placed these genes in more 'appropriate' (homogeneous) clusters resulting in the higher homogeneity and better separation. The heatmap (Eisen *et al.*, 1998) of the five distinct clusters obtained from k-means shows that the identified five 'distinct' clusters are enriched with similarly expressed genes (Figure 7.4).

Table 7.1
Comparison of distinct clusters identified using k-means against the reported clusters for the Yeast cell-cycle dataset shows that each distinct cluster is enriched with the genes from one of the reported clusters.

$\frac{Reported}{Identified}$	Early G1	Late G1	S	G2	M
1	67	5			
2		130	13		
3			46		
4			16	32	
5				20	55

Table 7.2
Comparison of distinct clusters identified using k-means and model-based (EI) clustering algorithms against the reported partition for the Yeast cell-cycle dataset. The proposed method correctly identified five clusters with k-means and model-based (EI) clustering. The average homogeneity and average separation are better than reported results.

Result	k^{opt}	H_{avg}	S_{avg}
Reported	5	0.5328	-0.0633
k-means	5	0.6615	-0.1125
Model-based	5	0.6615	-0.1125

7.3.2 Case Study 2: Yeast sporulation dataset

This data is reported by Chu *et al.* (1998). In this study, DNA microarrays are used to measure the expression levels of almost all the genes in Yeast Saccharomyces cere-

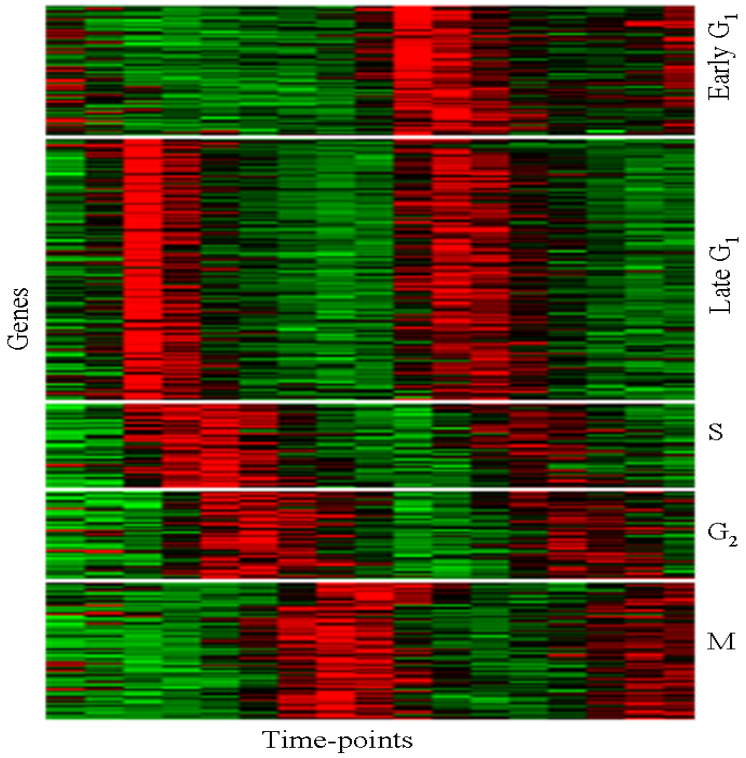

Fig. 7.4. Heat-map of five distinct clusters identified by k-means clustering in Yeast cell-cycle dataset. Each cluster is enriched with similarly expressed genes.

visiae during the temporal program of meiosis and spore formation. Expression levels are measured at seven time-points — 0, 0.5, 2, 5, 7, 9, and 11.5 hours. We collected the complete dataset and filtered the dataset to identify highly variant genes that show two-fold change at least once during the experiment. The final dataset contains 963 genes that qualified in this test.

Results for the Yeast sporulation dataset are shown in Figure 7.5. NEPSI Index finds 6 distinct clusters using both k-means and model-based (EI) clustering techniques. The BIC score for model-based (EI) is nearly flat after $k = 6$, thus indicating 6 clusters in this data (Figure 7.6). The Silhouette, Dunn's, and Davies-Bouldin indices suggest 4,

3, and 6 clusters, respectively (Figure 7.7).

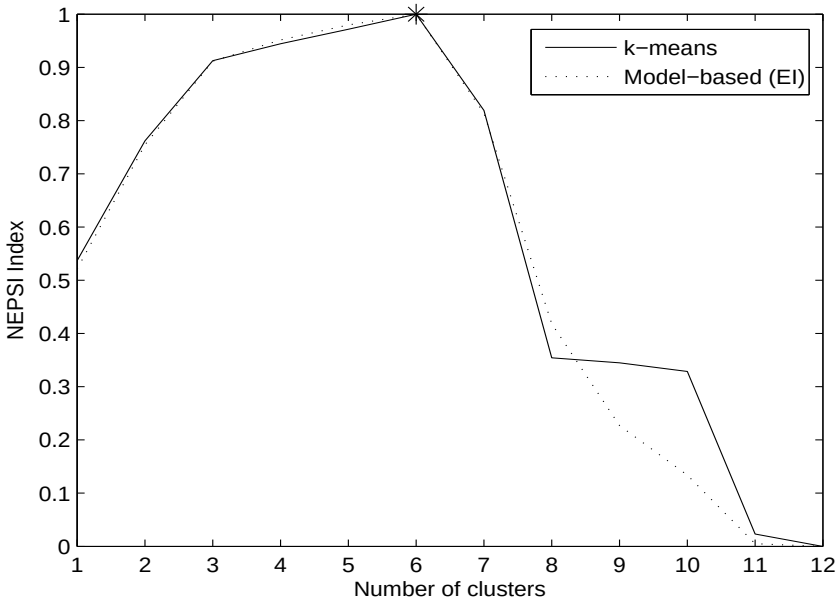

Fig. 7.5. Results for Yeast sporulation dataset. The NEPSI index identifies 6 distinct clusters using both k-means and model-based (EI) clustering algorithms. The BIC score for model-based (EI) clustering is flat after $k = 6$, thus also indicating 6 clusters in this dataset.

Since we don't have the 'true' solution in this case, we use *z-scores* proposed by Gibbons and Roth (2002) for evaluation of results. The *z-scores* of a partition indicates the relation between its clusters and the annotations relative to the random partition. The higher the score, the better the partition. It uses, for Yeast, the Saccharomyces Genome Database (SGD) annotation of yeast genes with the gene ontology developed by Gene Ontology Consortium (Ashburner *et al.*, 2000; Issel-Tarver *et al.*, 2002). Figure 7.8 shows the *z-scores* for this dataset as a function of number of clusters using k-means (solid line) and model-based (EI) (dotted line). The large positive values of the *z-scores* indicate that the clusters are significantly enriched with functionally re-

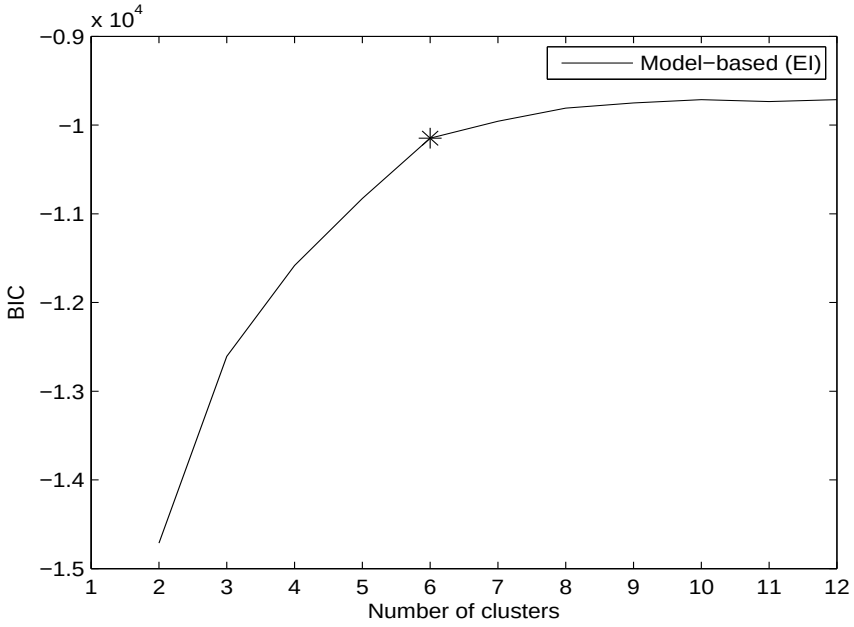

Fig. 7.6. Results for Yeast sporulation dataset. The BIC score for model-based (EI) clustering is flat after $k = 6$, thus also indicating 6 clusters in this dataset.

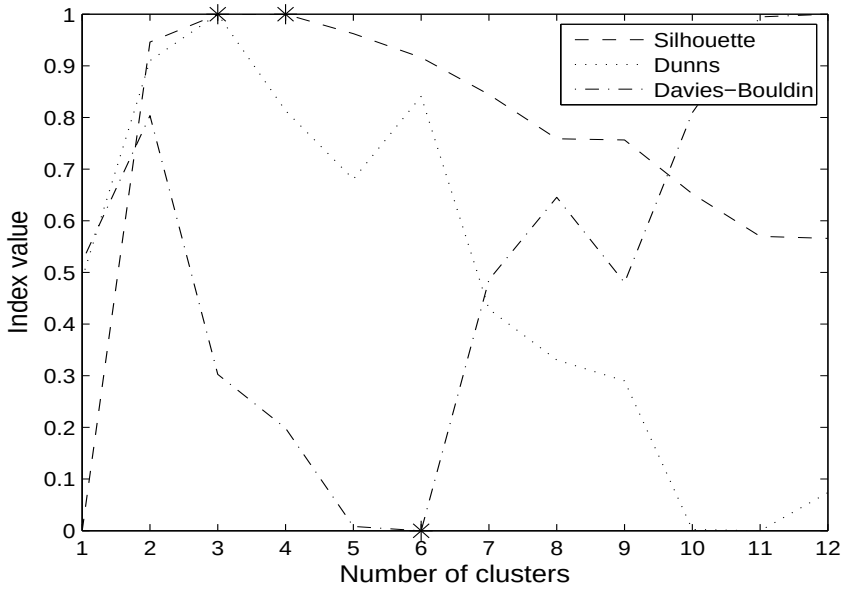

Fig. 7.7. Results for Yeast sporulation dataset. The Silhouette index finds 4 clusters, Dunn index finds 3 clusters, and Davies-Bouldin index selects the partition with $k = 6$.

lated genes than the random partitions. The *z-scores* for k-means are spanned from 36 to 58.6 with the maximum at $k = 6$. For model-based clustering, the *z-scores* are spanned from 31.7 to 57.8 with the maximum at $k = 7$. However, the *z-scores* for the partition with 6 clusters is equally good and only 0.4 lesser than the maximum. Hence, the partition with 6 clusters is reasonable.

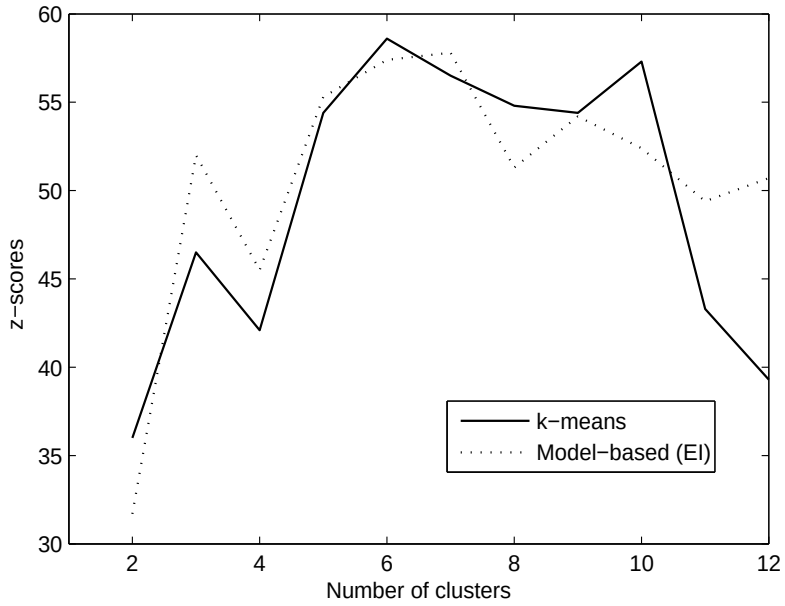

Fig. 7.8. z-scores as a function of number of clusters for Yeast sporulation dataset. The z-score is maximum for $k = 6$ for k-means clustering algorithm. z-scores for Model-based (EI) clustering are almost equal for $k = 6$ and $k = 7$.

We now analyze the partition with 'distinct' clusters extracted using k-means. The centers of the 6 clusters with error-bars are showed in Figure 7.9. Table 7.3 gives the enriched Gene Ontology (GO) processes for all the clusters. For each cluster, only 3 significantly enriched (based on p-value) processes are given. The number of genes in the cluster and the enriched process are given in the parenthesis. From Figure 7.9, it is

clear that each cluster has 'distinct' expression pattern. The first 3 clusters contain the genes that are activated during sporulation with difference in the time of activation and duration of activated state. For example, the first cluster contains the genes that are immediately and transiently activated after the change of environment. The remaining 3 clusters are formed with the repressed genes. Twenty three (out of 53) genes from this cluster 1 are involved in organic acid metabolism (P = 2.26 E-19), 19 involved in amine metabolism (P = 3.09 E-16), and 17 involved in amino acid and derivative metabolism (P = 1.91 E-14). Similarly, other clusters are also significantly enriched with the genes participating in sporulation (cluster 2), meiosis (cluster 3), macromolecular biosynthesis (cluster 4), alcohol and carbohydrate metabolism (cluster 5), and cell organization and biosynthesis (cluster 6), respectively. Each cluster is enriched with functionally related genes and the functions of genes in different clusters are different. This clearly shows the ability of proposed method to identify 'distinct' clusters with functionally enriched genes.

7.4 Discussion and Conclusions

Here, a method was proposed which uses Principal Component similarity factor to gauge distinctness of clusters in a partition. The proposed method selects the partition, out of a set of partitions, with maximum number of distinct clusters while satisfying the objectives of clustering. The efficacy of the proposed method was illustrated using two gene expression datasets and two clustering methods. In all the cases the proposed method performed reasonably well and showed better performance than other approaches.

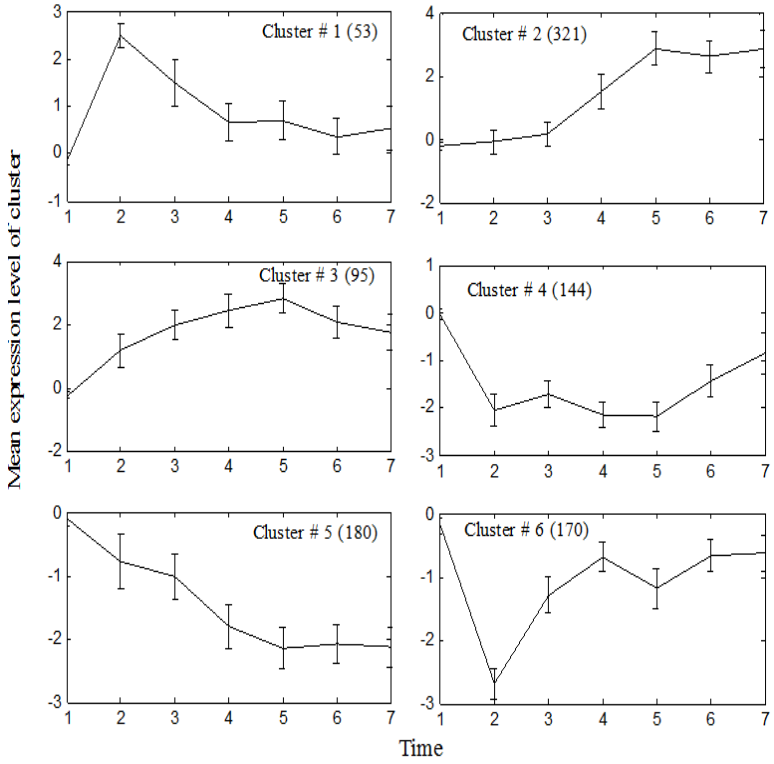

Fig. 7.9. Cluster centers of the 6 distinct clusters identified in Yeast sporulation data. Clusters 1, 2, and 3 are up-regulated and cluster 4, 5, and 6 are down-regulated. Each cluster is enriched with genes related to specific biological function.

The calculation of NEPSI requires the determinant of covariance matrix of clusters. If some of the input variables are highly correlated then the covariance matrix becomes ill-conditioned and determinant approaches zero. To avoid this, we calculated the determinant from the product of most significant eigenvalues that capture at least 93% of variance. The results were unchanged even when using all the eigenvalues in the case studies.

Since computational efficiency is an important requirement in cluster validation, we also evaluate the performance of the proposed method. To eliminate the effect of initial guess on k-means, we initiated k-means with 25 guesses for each value of k and

Table 7.3
Functional mapping of the 6 distinct clusters identified by k-means clustering algorithm in Yeast sporulation dataset. Clusters are enriched with genes with relevant functions and the function of each cluster of genes is different from those of others.

Cluster	GO process
C1(53)	Organic acid metabolism (23) Amine metabolism (19) Amino acid and derivative metabolism (17)
C2(321)	Sporulation (42,) Development (68) Cell-cycle (45)
C3(95)	Meiosis (29) Chromosome segregation (14) Meiotic gene conversion (6)
C4(144)	Macromolecule biosynthesis (83) Ribosome assembly (12) Cytoplasm organization and biogenesis (17)
C5(180)	Alcohol metabolism (20) Carbohydrate metabolism (23) Alcohol catabolism (10)
C6(170)	Cell organization and biogenesis (88) Biopolymer metabolism (77) Ribosome assembly (16)

used the best result for further analysis. The time taken for identifying distinct clusters using the proposed method in Yeast cell-cycle dataset with k-means and model-based clustering (including the time for clustering) were 0.6 and 2 mins, respectively in a Pentium 4, 2.8GHz class machine using $Matlab^{TM}$ 6.5.1. Similarly, the time taken for Yeast sporulation dataset was 1.2 and 10 mins, respectively. Excluding the time for clustering, the proposed method identifies best partition within one minute in both the datasets.

8. BAYESIAN APPROACH FOR INTEGRATING TRANSCRIPTION REGULATION AND GENE EXPRESSION DATA

8.1 Introduction

Cells carry-out their complex functions by altering the transcription rates of specific genes throughout the genome in timely fashion. The transcription rate of a gene is precisely regulated by the combined action of several activator and repressor proteins called Transcription Factors (TFs) that bind to the promoter regions of genes and regulate the expression of genes (Lee and Young, 2000). A primary goal of biological studies is to understand gene regulation and to identify which Transcription Factors regulate which genes. Such insights are essential to develop models that predict cell responses to novel conditions. Even though analysis of genome-wide expression profiles enhances our understanding of cellular processes, there are certain inherent challenges when assigning regulators for genes.

Microarray expression profiling does not distinguish between effect of direct binding of TF to a target gene and the indirect effect caused by intermediate TFs. So genes can have similar expression profile even though their regulators are different. Hence clustering of co-expressed genes is of limited use to reliably assign TFs to genes (Bar-Joseph *et al.*, 2003b). Segal *et al.* (2003) proposed a method to identify the targets of regulators using expression data. Their approach assumes that expression profile of regulated genes depend on expression of their regulators. This assumption is not always

valid. For example, during post-transcriptional modifications of TFs the expression of regulator does not change appropriately. Hence expression data alone is not adequate for identifying the regulators for genes.

However, there are other genomic data sources that provide complementary information about TF-gene interactions. For example, the genome-wide location analysis method (Ren *et al.*, 2000) identifies the direct TF-gene physical interactions at genome-scale by combining the chromatin immunoprecipitation (ChiP) procedure with microarrays. Though location data is highly useful, false positives and false negatives hinder the assignment of TFs to genes. For instance, there is only moderate agreement between the genome-wide location studies of *Saccharomyces Cerevisiae* by Iyer *et al.* (2001) and Simon *et al.* (2001) for the same TFs: mbp1, swi4, and swi6 (Futcher, 2002). However, by integrating gene expression and genome-wide location data one can extract useful and reliable information about regulation of genes.

There are two reported approaches to combine these two datasets. The first approach, proposed by Hartemink *et al.* (2001) uses Bayesian networks with the location data influencing the model prior and the expression data influencing the likelihood. The identified network provides the links between TFs and their target genes. As another approach, Bar-Joseph *et al.* (2003b) proposed a method that compliments the expression data with location data to overcome the false negatives in location data. In their approach, location data is used to classify genes into different sets such that genes in each set are bound by the same TFs. Then for each such group, a minimum radius sphere (capturing the genes within the set) is found in gene expression data. Then the

genes without any regulators (false negatives) in location data are classified into these sets if they fall in the sphere and have the combined probability of regulatory interactions lesser than the predefined threshold. One of the limitations of their method is the computational complexity of finding the minimum radius sphere in the high dimensional expression data. Furthermore, their method is not extendable to other datasets such as gene / promoter sequences. In this chapter, a Bayesian approach is proposed to integrate gene expression data with genome-wide location data in order to reliably assign TFs for genes.

8.2 Proposed Method

The proposed method uses the genome-wide location data and gene expression data in an incremental way to reliably assign regulators to genes. The method is schematically shown in Figure 8.1. A model is first developed using genes for which high confidence transcription factors are available in the location data. This model is then used for assigning TFs to the remaining genes (i.e. those without reliable transcription factor information) using expression similarity. There are three steps in the method: (1) Conversion of location data into binary values, (2) Model development for genes with TFs in location data, and (3) Bayesian classification of the remaining genes using the model identified in Step 2. We describe these three steps in the following sections.

8.2.1 Conversion of Location Data to Binary Values

The genome-wide location data contains the p-values for the TF-gene interactions. The lower the p-value, the higher the probability of interaction. These p-values have to

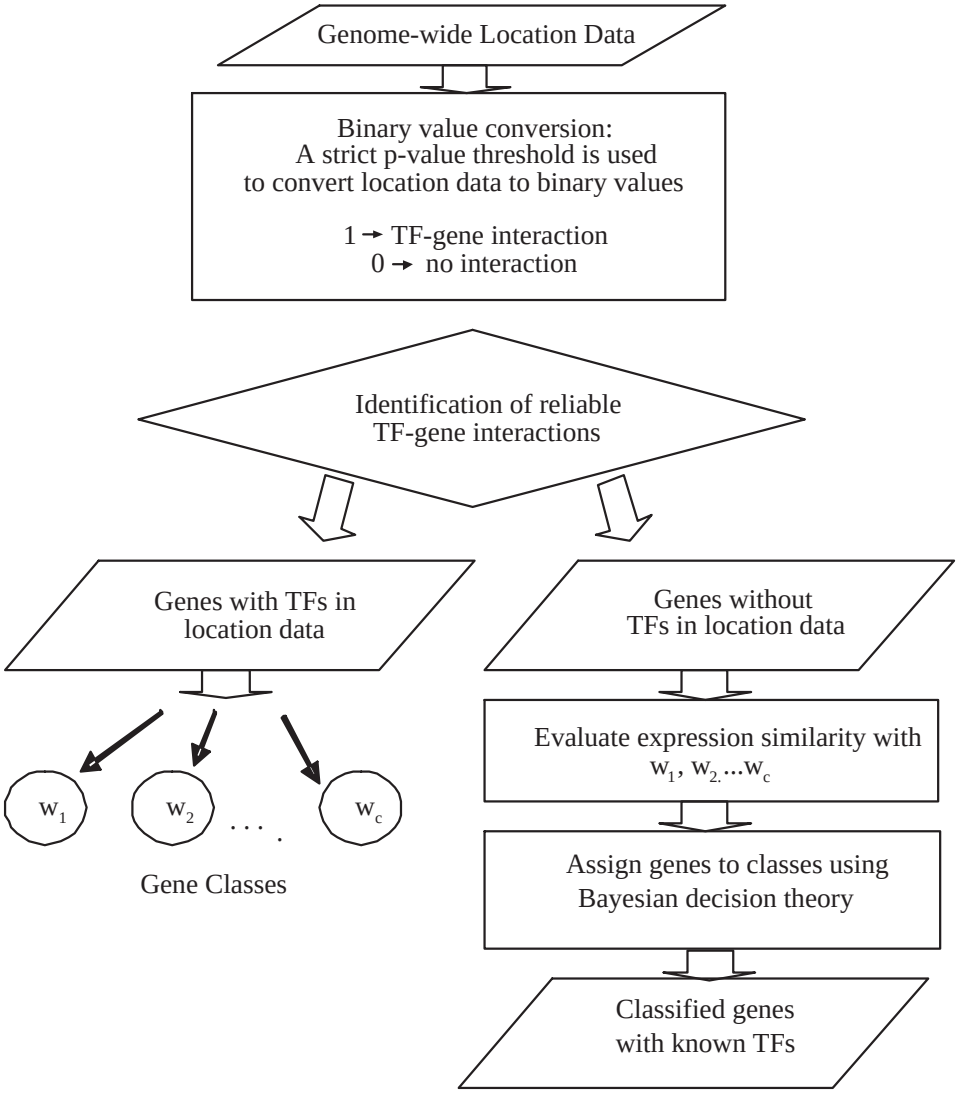

Fig. 8.1. Proposed methodology for integrating gene expression and genome-wide location data. Genes are first classified into several classes where each class of genes is bound by the same transcription factors (TFs). Unclassified genes are the assigned to one of the existing classes using Bayesian decision rule.

be converted to binary values to decide whether a particular TF binds to the gene or not. The value '1' indicates an interaction between a transcription factor and gene whereas the value '0' indicates no interaction. Binary conversion is carried out by selecting a suitable threshold for p-value.

Let $B_{m\times t}$ be the location data for m genes on t transcription factors where each element, b_{ij}, is the p-value for the interaction between gene i^{th} gene and j^{th} transcription factor. Then, we consider that the interaction between the i^{th} gene and j^{th} transcription factor occurs if b_{ij} is smaller than the p-value threshold P_T

$$b_{ij} = \left\{ \begin{array}{l} 1 \; if \; b_{ij} < P_T \\ \\ 0 \; otherwise \end{array} \right\} \tag{8.1}$$

8.2.2 Model Development for Genes with TFs in Location Data

A model consisting of TFs linked to expressions of the gene they regulate is developed in the next step. For this, by using a strict P_T, we can identify the most reliable interactions after binary conversion. Then different classes in the location data are identified such that all the genes within a class are bound by same TFs set. For this, the method searches for all the possible combinations of transcription factors (i.e. 2^{t-1} combinations are possible with t TFs). For each such set, our method finds the genes bound by all the TFs and considers them as a model component. Genes are allowed to be present in multiple model components. For example, a gene bound by regulators A, B and C in location data is allowed to be present in both the classes regulated by $\{$A, B$\}$, and C, respectively. However, it is not allowed in model components regulated by $\{$A$\}$ and $\{$B$\}$.

8.2.3 Model-based Bayesian Classification

After identification of reliable interactions and classification of genes, putative TFs are assigned to the remaining genes using the Bayesian classification rule. In general,

Bayesian rule updates the belief of a hypothesis in the light of new evidence. In the present context, the Bayesian rule updates the *a priori* probability (belief) that a previously unclassified gene belongs to the one of the classes (hypothesis) to *a posterior* probability using the expression similarity of the gene to the already classified genes (evidence).

Let $X_{m\times n}$ be the expression data matrix containing m genes measured at n time points. Assume that these m genes are classified into $w_i(1 \leq i \leq C)$ classes where all the genes in class w_i are bound by the same set of transcription factors. Given a new gene with expression profile represented by x , the probability that x belongs to the i^{th} class is given by Bayesian rule as (Duda and Hart, 1973):

$$P(w_i/x) = \frac{P(w_i).p(x/w_i)}{p(x)} \tag{8.2}$$

where $P(w_i/x)$ is the *a posterior* probability of x belongs to class w_i , $P(w_i)$ is the *a priori* probability that x belongs to class w_i , $p(x/w_i)$ is the probability density function of x given the class w_i , and $p(x)$ is the probability density function of x given by:

$$p(x) = \sum_{i=1}^{C} p(x/w_i).P(w_i) \tag{8.3}$$

The Bayes rule (8.2) shows how measuring the expression profile of a gene changes the *a priori* probability to *a posterior* probability. According to the Bayesian theory, to reduce the probability or error, a gene should be assigned to the class for which it has the highest posterior probability *i.e.* assign x to class w_j if

$$P(w_j/x) > P(w_i/x) \quad \forall i \neq j \tag{8.4}$$

The denominator in Equation 8.2 is a normalization factor which makes the sum of posterior probabilities equals to 1. For classification purposes, it is not necessary to have the normalized posterior probabilities; hence the denominator is normally discarded. Then the classification rule in Equation 8.4 becomes

$$p(x/w_j).P(w_j) > p(x/w_i).P(w_i) \quad \forall i \neq j \tag{8.5}$$

In practice, we can use any monotonic function of $p(x/w_i)P(w_i)$ that is convenient for calculation of posterior probabilities. In this work, we use the logarithm of $p(x/w_i)P(w_i)$ represented by $g_i(x)$. The conditional probability function of x for a given class w_i , $p(x/w_i)$, is assumed to be a multivariate normal distribution. Hence the Bayesian decision rule is given by:

$$g_j(x) > g_i(x) \quad \forall i \neq j \tag{8.6}$$

$$g_i(x) = -\frac{1}{2}(x-\mu_i)\Sigma_i^{-1}(x-\mu_i)^T - \frac{1}{2}\ln(|\Sigma_i|) + \ln P(w_i) \tag{8.7}$$

whereμ_i and Σ_i are the mean and covariance of class w_i , respectively. $P(w_i)$ is the fraction of genes in class w_i . The mean and covariance of a class are estimated using the samples in that class as:

$$\mu_i = \frac{1}{n_i}\sum_{x \epsilon w_i} x \tag{8.8}$$

$$\Sigma_i = \frac{1}{n_i} \sum_{x \epsilon w_i} (x - \mu_i)(x - \mu_i)^T \tag{8.9}$$

where n_i is the number of samples in class w_i.

In Equation 8.7, the Bayesian rule is used with the Mahalanobis distance between the expression profile of gene x and the mean of the class μ_i to generate the a posterior probability that a gene belongs to a class. As a strict P_T is used in binary conversion (Equation 8.1) of the location data, the gene interactions are identified with high confidence (few false positives). False negatives may be induced by the strict threshold; the proposed method reduces such false negatives by complimenting with gene expression data. For each gene with no regulators in location data, the proposed method uses its expression similarity to the already classified genes as evidence and generates the probability that it belongs to these classes. Finally, the gene is assigned to the class (set of TFs) for which it has highest similarity. Hence the proposed method reliably assigns the TFs to genes.

8.3 Results

We evaluate the proposed Bayesian approach to identify the regulators for Saccharomyces Cerevisiae cell-cycle regulated genes reported by Spellman *et al.* (1998). Spellman *et al.* (1998) measured the expression levels of Yeast genes at 73 time points during three independent conditions: α factor arrest, elutriation, and arrest of cdc15. They identified approximately 800 cell-cycle regulated genes using periodicity and correlation algorithms. The genome-wide location data for these genes are collected from Simon *et al.* (2001). Simon *et al.* (2001) conducted the genome-wide location study for

nine known cell-cycle transcription factors: Fkh1, Fkh2, Ndd1, Mcm1, Ace2, Swi5, Mbp1, Swi4, and Swi6. Out of the 800 cell-cycle regulated genes location data is available for 794 genes. We use these 794 genes in this study.

We used the strict p-value threshold, P_T , of 0.001 to convert the p-values to binary values (Bar-Joseph *et al.*, 2003b). This means there is 0.1% probability that an interaction happened by chance. We then tested all the combination of the nine cell-cycle regulators for eligible gene classes. This procedure identified 28 classes containing 206 unique genes (out of these 794). Considering the false positives even at this strict threshold, we considered only the classes containing at least 5 genes. The first three columns of Table 8.1 show the classes, class sizes and their regulators. In the third step, we calculated the probabilities for each of the remaining 588 genes belonging to all the 28 classes using Bayesian rule. The proposed approach needs the inverse of the covariance matrix of each class to generate the posterior probabilities (Equation 8.7). Since the dimensionality of the expression data is 73 (time points), we need at least 73 genes in each class to calculate a non-singular covariance matrix and hence the inverse. Given that a class has smaller number of genes than the minimum requirement, calculation of covariance matrix is untenable. To solve this problem we used Principal Component Analysis (PCA). PCA is a multivariate technique that finds the principal components (directions) of variability in the data, and transforms the related variables into a set of uncorrelated ones. These principal components (PCs) are the linear combinations of original variables (Jackson, 1991). The first few PCs capture most of the variance in the data whereas the remaining PCs represent noise. Hence the dimensionality of the data can be reduced by considering the first few PCs. We applied PCA on

the whole data before identifying the classes. Since the minimum size of the classes is 5, we used the first four PCs in order to have the non-singular covariance matrix for all classes. Then the 588 genes are assigned to one of these 28 classes using their highest posterior probabilities. The last column of Table 8.1 shows the number of novel genes assigned to the 28 classes. The number of gene assigned to different sets varies from zero to a maximum of 98. The distribution of normalized maximum a posterior probability of 588 genes is shown in Figure 8.2. 169 (out of 588) genes have the highest posterior probability of at least 0.5.

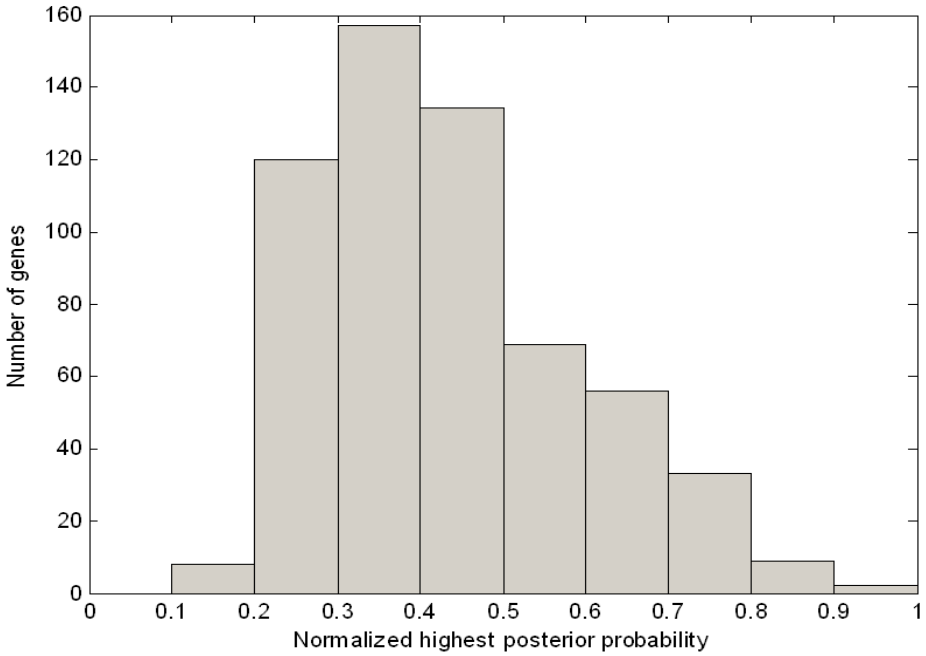

Fig. 8.2. Distribution of normalized maximum a posterior probability of the 588 genes whose regulators are predicted using the proposed method.

Here, we give a brief analysis of the classification results. Spellman *et al.* (1998) clustered cell-cycle regulated genes into different clusters based on their similarity in

Table 8.1
Prediction of class labels for genes without any transcription factors in genome-wide location data. Genes are assigned to the class with highest posterior probability.

Model development			Classification of novel genes
Class No.	Size	Transcription Factors (TFs)	No. of genes classified
1	5	Fkh2 Ndd1 Mcm1 Mbp1 Swi4 Swi6	0
2	6	Fkh2 Ndd1 Mbp1 Swi6	5
3	7	Fkh2 Ndd1 Swi4 Swi6	1
4	5	Fkh2 Mcm1 Swi4 Swi6	0
5	7	Fkh2 Ace2 Swi4 Swi6	1
6	12	Fkh2 Mbp1 Swi4 Swi6	14
7	5	Mcm1 Mbp1 Swi4 Swi6	9
8	5	Ace2 Swi5 Mbp1 Swi6	4
9	6	Ace2 Swi5 Swi4 Swi6	5
10	5	Ace2 Mbp1 Swi4 Swi6	7
11	5	Swi5 Mbp1 Swi4 Swi6	1
12	9	Fkh2 Ndd1 Mcm1	18
13	6	Fkh1 Fkh2	2
14	6	Fkh2 Ace2	3
15	6	Fkh2 Swi5	18
16	5	Fkh2 Swi4	19
17	7	Fkh2 Swi6	1
18	9	Ace2 Swi5	18
19	8	Mbp1 Swi4	19
20	13	Mbp1 Swi6	50
21	28	Swi4 Swi6	50
22	16	Fkh1	98
23	16	Ndd1	65
24	22	Mcm1	76
25	34	Swi5	35
26	7	Mbp1	30
27	16	Swi4	31
28	5	Swi6	8

expression over all experiments. We analyze the results for some of these clusters:

CLN2 Cluster: The CLN2 cluster contains 76 genes that show peak expression during mid-G1 phase in their expression. These genes are regulated by MBF (complex of Mbp1 and Swi6) and SBF (complex of Swi4 and Swi6) (Spellman *et al.*, 1998). TFs are available for 29 (out of 76) genes in location data, but no regulators are found for remaining 47 genes. The proposed method correctly identifies the regulators for these genes. Our approach assigns either MBF or SBF or both as the regulators for 37 of these genes. These genes include POL12, POL30, CDC9, and STB1 etc. For the remaining genes, one of the subunits of SBF and MBF are assigned.

CLB2 cluster: The CLB2 cluster contains 36 genes regulated by the complex formed the transcription factors Mcm1, Ndd1, Fkh1/Fkh2 (Koranda *et al.*, 2000; Zhu *et al.*, 2000). Regulators are not available for 15 of these genes in genome-wide location data. Our approach identifies all three TFs Mcm1, Ndd1, and Fkh1/Fkh2 as the regulators for 12 out of these 15 genes. These genes include CLB1, MOB1, and HOF1 etc. Ndd1 is assigned as a regulator for 2 genes and Fkh1 is assigned for the remaining one gene.

MCM cluster: The MCM cluster contains 34 genes regulated by Mcm1 (Spellman *et al.*, 1998). Our approach predicted the transcription factors for 23 out of these 34 genes. Comparing to the other clusters, the results for this clusters are not accurate. Mcm1 is assigned as a transcription factor for 9 genes and Ndd1 for 7 genes. One or more of the Fkh2, Ace2, Swi4, and Swi5 are assigned to the remaining genes.

Application of Bar-Joseph *et al.* (2003b) procedure for these same datasets yielded 34 classes with a mean class size of around 9. Only 22 of the 76 genes from CLN2 cluster are included in these 34 classes. Similarly 19 and 15 genes from CLB2 and MCM clusters included in these 34 classes. Moreover, these 22, 19, and 15 genes are distributed over several classes giving no clear clue of regulators for these genes.

8.4 Discussion and Conclusions

Here, a Bayesian approach to integrate genome-wide location data with gene expression data to predict the regulators for genes has been proposed. The proposed method has been evaluated by predicting the regulators for *Saccharomyces Cerevisiae* Cell Cycle regulated genes. The proposed method showed reasonable performance and correctly predicted the regulators for several genes. However, there are several issues to be addressed. The first one is the low sample situation. Out of these 794 genes used in this study, only 206 genes have reliable TFs in location data whose expression data is later used to identify the parameters (mean and covariance matrices). The minimum class size is 5. The estimation of the parameters generally needs more genes in each class to cancel the effect of noise in the data. This problem can be eliminated by using the same covariance matrix for all the classes. Then the parameters can be reliably estimated by pooling the genes from all the classes. This also eliminates the need for PCA. This needs further examination. Nevertheless, for the case considered here, the proposed method showed reasonable performance. All the genes with no regulators in location data are assigned to one of the predefined classes based on their posterior probability. Even though the results are reasonably correct, it is preferable to develop criterion to reject genes in case no significant evidence is available for classification.

From Figure 8.2, it is evident that some of the genes have the maximum posterior probability less than 0.5 indicating that they do not have significant evidence to be assigned to any class. Hence, it is better not to assign these genes to any of the classes.

9. INTEGRATIVE CASE STUDY: IMPROVEMENT OF AN *ESCHERICHIA COLI* STRAIN FOR PRODUCING RECOMBINANT PROTEIN

9.1 Introduction

The production of recombinant proteins in host organisms such as *Escherichia coli* has become indispensable for both research and industrial applications. DNA plasmids containing heterogenous genes are inserted into the host organisms for the production of proteins and metabolites. It has been demonstrated that plasmid bearing (PB) cells grow slower than the wild-type (WT) cells (without plasmids) due to additional metabolic burden on host cells by plasmid replication, DNA transcription, and plasmid-encoded mRNA translation (Bentley *et al.*, 1990). In this chapter, the data-driven framework proposed in Chapter 3 is employed to analyze the gene expression datasets from both WT and PB *Escherichia coli*. The analysis is aimed at understanding the genetic reprogramming due to expression of foreign genes in *Escherichia coli* and to identify genetic targets for recovering growth.

Expression of heterogenous genes in host organism has profound effects on their metabolism and phonotype. Especially, over-expression of recombinant proteins use large portion of cells resources and create a metabolic load on cells leading to the decrease in cell growth rate. Though the physiology of PB cells is more or less understood, the changes occurring at the genetic level are not clearly known (Diaz-Ricci *et al.*, 1995). It is essential to understand the genetic level changes in PB strain to

recover the growth rate and subsequently the yield and productivity of recombinant protein.

As described in Chapter 3, the proposed data-driven framework contains a series of data-mining methods which provide useful information about the functioning of cells. The first step in the proposed framework is to identify the genes differentially expression between the WT and PB cells. The differentially expressed genes (DEG) are identified using the method proposed in Chapter 4. These DEG provide the molecular level changes happened in PB cells compared to WT cells. The second step is to organize these DEG into different clusters such that genes within a cluster are similar in expression. Both clustering and cluster validation tools described in chapters 5 and 6 are used for this purpose. The last step in the proposed approach is to use the correlation information between the Transcription Factors (TFs) and genes to identify the key TFs which have to be modified to enhance the growth of PB strain.

9.2 *Escherichia coli* case study

Escherichia coli has been the most widely used organism for production of recombinant proteins as its molecular genetics, physiology and expression systems are relatively well characterized (Choi *et al.*, 2006). Fed-batch experiments are conducted for both WT *Escherichia coli* and the *Escherichia coli* producing beta-lactamase antibiotic resistant marker protein. The experiments were conducted at Bioprocessing Technology Institute (BTI), A*STAR, Singapore (Ow *et al.*, 2006, 2009). The experimental details are described below.

Strain

In this study *Escherichia coli K12* of strain DH5α was used. NS3 plasmids were constructed from pcDNA3.1 (Invitrogen, Carlsbad, CA) containing a beta-lactamase antibiotic selection marker and a non expressing 1.8 kb DNA fragment of Dengue virus. The plasmids were inserted into *Escherichia coli* (Ow *et al.*, 2006, 2009).

Growth Medium

The complex medium R25 was used in this study. The components in the growth medium were: yeast extract (12 gL^{-1}), tryptone (6 gL^{-1}), dipotassium phosphate (6 gL^{-1}), magnesium sulphate (0.48 gL^{-1}), and glucose (5 gL^{-1}) (Ow *et al.*, 2006, 2009).

Fed-batch experiments

Fed-batch experiments were conducted for both WT strain and the PB strain. The cell growth (in terms of Optical Density, OD_{600}) and the glucose concentrations were monitored over time. The concentrations are shown in Figure 9.1 and Figure 9.2 for WT and PB cells, respectively. Figure 9.1 and Figure 9.2 are drawn with same scales.

From Figures 9.1 and 9.2, it is apparent that the growth rate of PB cells is lower than that of WT cells. Also, PB cells have prolonged lag phase compared to the WT cells. Since the growth medium and conditions are similar between the WT and PB strain, the prolonged lag phase can be attributed to the maintenance of plasmid. Since the recombinant protein productivity is dependent on cell growth, lower growth of PB

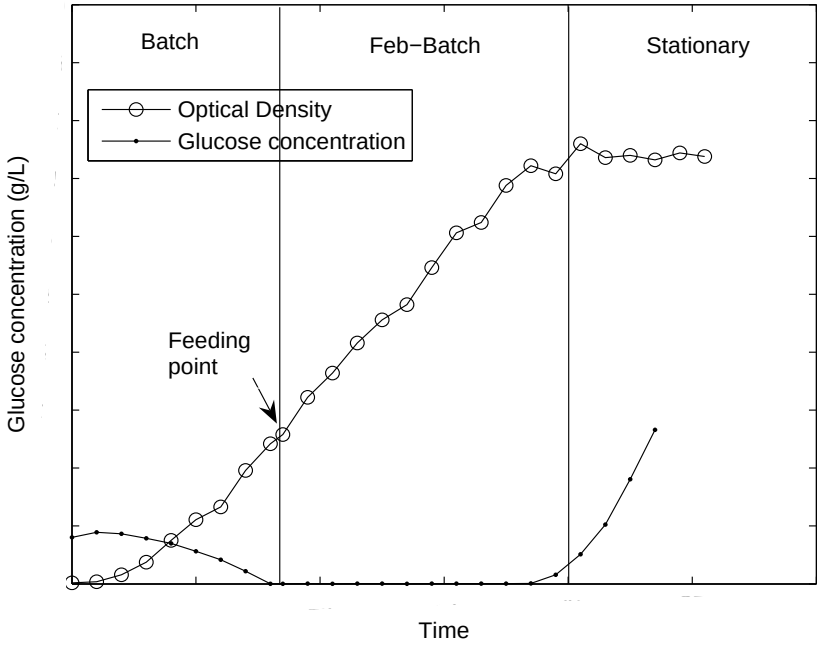

Fig. 9.1. Concentration of glucose and cell density for WT strain

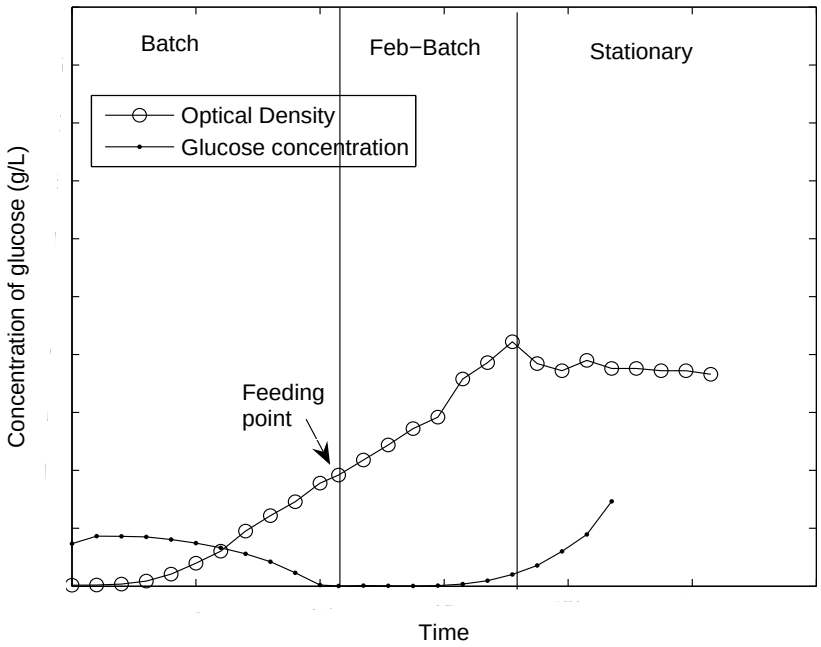

Fig. 9.2. Concentration of glucose and cell density for plasmid bearing strain

cells leads to smaller quantity of protein. In this study, we use the proposed data-mining framework to identify genetic targets to improve the PB strain performance.

Gene expression data

Microarrays were used to measure the expression of almost all the *Escherichia coli* genes in the WT cells and cells containing plasmid. The expression levels were measured at 8 time-points during the fermentation experiments. The time-points spanned all the phases including early exponential, before feeding, fed-batch stage, and stationary phase. The gene expression time-course data contain information about the changes during the fermentation. To understanding the dynamic changes at the gene expression levels, during the growth of the cells, these expression datasets have to be analyzed. Analysis of the gene expression datasets using the methods described in this thesis would provide understanding of cell functioning and provide the basis for selecting genetic targets.

9.3 Identifying differentially expressed genes

The proposed data-driven framework contains a series of data-mining methods for identifying DEG, clustering, finding number of clusters and for assigning TFs for genes. These methods extract the information about the functioning of cells which leads to selecting of genetic targets. First, PCA based algorithm described in Chapter 4 is used for identifying genes that are differentially expressed between the WT and PB cells. The cross validation approach for finding number of PCs to model the WT data returns 3 PCs (Figure 9.3). These 3 PCs capture 70.32% of total variation in data. The eigenval-

ues corresponding to these two PCs are 1.12, 0.89, and 0.53, respectively (Figure 9.4). The eigenvalues corresponding to the remaining PCs are very small. Hence, only three PCs are used to model the WT. Expression data from the PB strain is projected onto the PCA model developed from the WT gene expression data. Robust Mahalanobis distance described in Section 4.4 is used for finding whether a gene is differentially expressed.

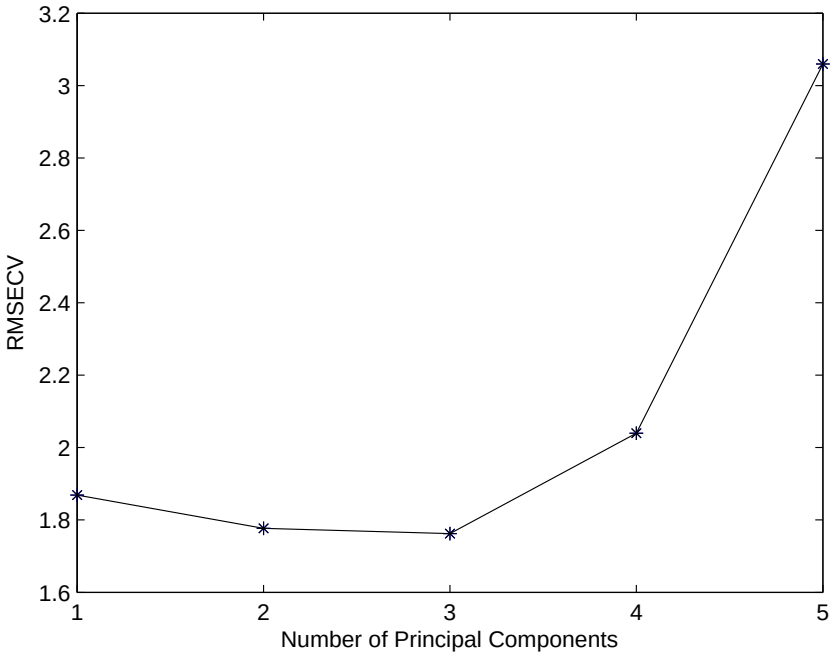

Fig. 9.3. Cluster validation result for WT gene expression data. RMSECV takes minimum at number of PCs 3

At a p-value threshold of 0.05, 534 genes are identified as differentially expressed between the WT and the PB strains. The difference of scores for all the genes on the 3 PCs used to model the data is shown in Figure 9.5. The DEG identified by the proposed method are shown by '*' on the scores plot. From Figure 9.5, it is clear that the proposed method identified genes that are away from the origin as differentially

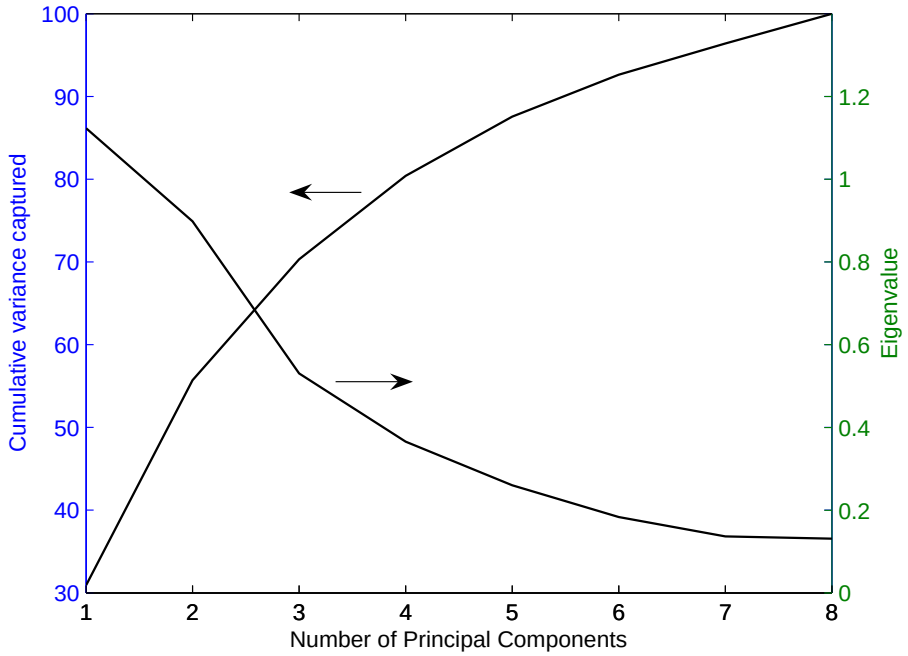

Fig. 9.4. Cumulative variance and Eigenvalues for WT gene expression data

expressed genes. In the next section, these DEG are used to understand the effects of plasmid on the host strain.

9.3.1 Mapping of DEG on the Central Metabolic Network

The Central metabolic network shown in Figure 9.6 comprising Glycolysis, TCA cycle and Phosphate Pentose pathways (PPP) provides the precursors, co-factors and energy for biosynthesis and other metabolic pathways (Mandelstam *et al.*, 1982). It is known that production of recombinant proteins results in differential expression of several genes from central metabolic network (Choi *et al.*, 2006). Since central metabolic network provides the precursors for biomass, disruption of it directly leads to low growth rates. So, the DEG identified above are first mapped on to the central metabolic network to explore the effect of plasmid on metabolism.

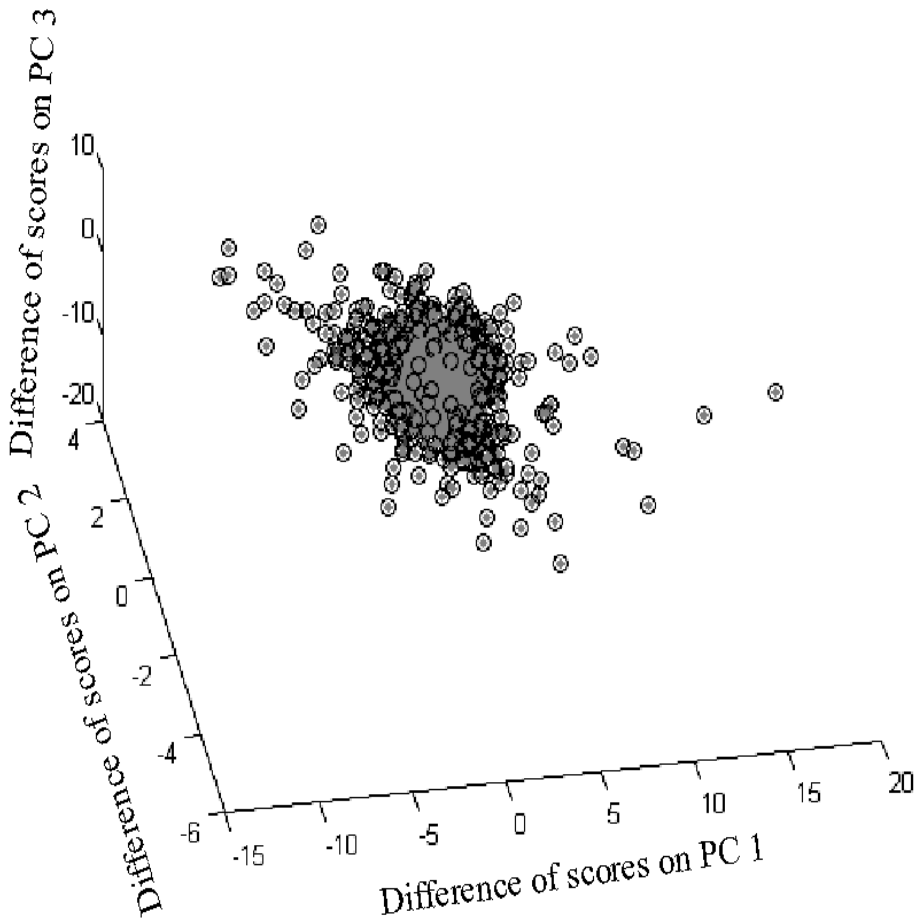

Fig. 9.5. Plot of difference of scores of all genes on 3 dominant PCs. Genes marked as '*' and identified as differentially expressed genes

Out of approximately 40 genes that catalyze the reactions in central metabolic network of *Escherichia coli K12*, five genes are found to be differentially expressed between the WT strain PB strain. Two of them, namely *pgi* and *fbaB*, are from the glycolysis pathway, one, *tktB*, from Phosphate Pentose Pathway and two, *acnA* and *icd*, from the TCA cycle.

The Glycolysis pathway is the primary catabolic route for degradation of carbohydrates to provide energy and precursor building blocks for the synthesis of other

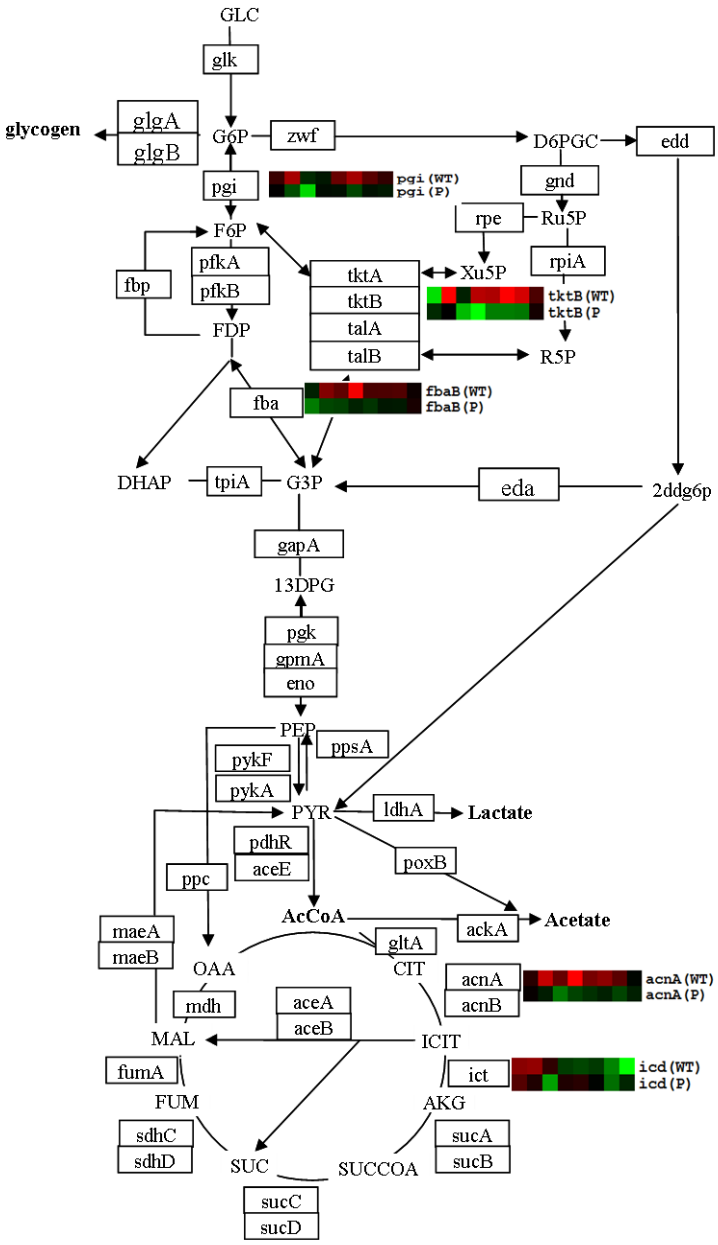

Fig. 9.6. The Central metabolic network of *Escherichia coli*

macromolecules. Glucose or other sugars are fed into glycolysis pathway and degraded to Pyruvate that is further degraded in the TCA cycle. Two enzymes in this pathway, namely *pgi* and *fbaB*, are found to be down-regulated in the PB strain compared to the WT strain. The expression of these two enzymes are shown as heatmaps (horizontal

bars of green and red colors) in Figure 9.6.

pgi catalyzes the reaction Glucose-6-Phosphate (G6P) to Fructose-6-Phosphate. Down regulation of *pgi* forces the glucose catabolism through Phosphate Pentose Pathway which converts glucose to Fructose-6-Phosphate (F6P) and Glyceraldehyde-3-Phosphate (G3P). As a consequence of this, NADPH production increases through PPP and creates NADPH imbalance in the cell leading to lower growth (Kabir and Shimizu, 2003). F6P and G6P produced through PPP is fed through the glycolysis pathway. However, the enzyme *tktB* which is an intermediate step in PPP is also found to be down-regulated. This indicates that the glucose intake is reduced which clearly explains the lower growth rate. Activity of *tktB* is also required to produce aromatic amino acids and *tktB* mutant strain of *Escherichia coli* requires supply of aromatic amino acids for growth (Zhao and Winkler, 1994).

The end-product of glycolysis and PPP is Pyruvate (PYR) which is further degraded in TCA cycle thus providing energy and building blocks for macromolecular biosynthesis. Pyruvate is first oxidized to Acetyl Coenzyme A (AcCoA) before it is fed to TCA cycle. In TCA cycle, complete oxidation of AcCoA takes place and energy is generated along with precursors for biosynthesis. In TCA cycle, both *acnA* and *icd* are found to be down-regulated in PB strain. These two reactions catalyze the second and third steps of the TCA cycle. Down-regulation of these two genes reduces the uptake rate of TCA, which results in conversion of AcCoA to acetate (El-Mansi and Holms, 1989). It is commonly observed that *Escherichia coli* cells excrete 10-30% of the carbon flux from glucose as Acetate (Farmer and Liao, 1997). Acetate accumulation is observed in

fermentation experiments for both WT and PB cells. In WT strain fermentation, Acetate concentration is observed to increase smoothly as the cells enters the exponential growth phase. Peak Acetate concentration was observed at the point where glucose concentration is limiting and feeding started. In the rest of the experiment, Acetate concentration is minimal during exponential growth and increased in stationary phase for WT cells. On the contrary, in PB strain Acetate concentration continues to increase throughout the experiment. The down-regulation of *acnA* and *icd* which reduces the uptake rate of TCA cycle might have diverted the AcCoA to Acetate. It is therefore clear from this mapping of gene expression onto the central metabolic network that central metabolic network is repressed in PB strain. Since the central metabolic network supplies the building blocks for other biosynthesis, cell growth was consequently reduced.

9.3.2 Effect of plasmid on Amino acid production

It is known that plasmid maintenance and recombinant protein production creates a metabolic burden on host cells resulting in down-regulation of macromolecular synthesis, especially the amino acid production. We now consider the amino acid production in PB strain. The amino acid production pathways for *Escherichia coli* are shown in Figure 9.7.

The expression of 8 genes involved in amino acid biosynthesis were down regulated in PB cells *(hisG, aroF, pheA, trpC, livC, dapA, metE* and *argF)*. *hisG* catalyzes the first reaction of histidine biosynthesis and down-regulation of *hisG* decreases the production of histidine. The production of aromatic amino acid pathway was completely

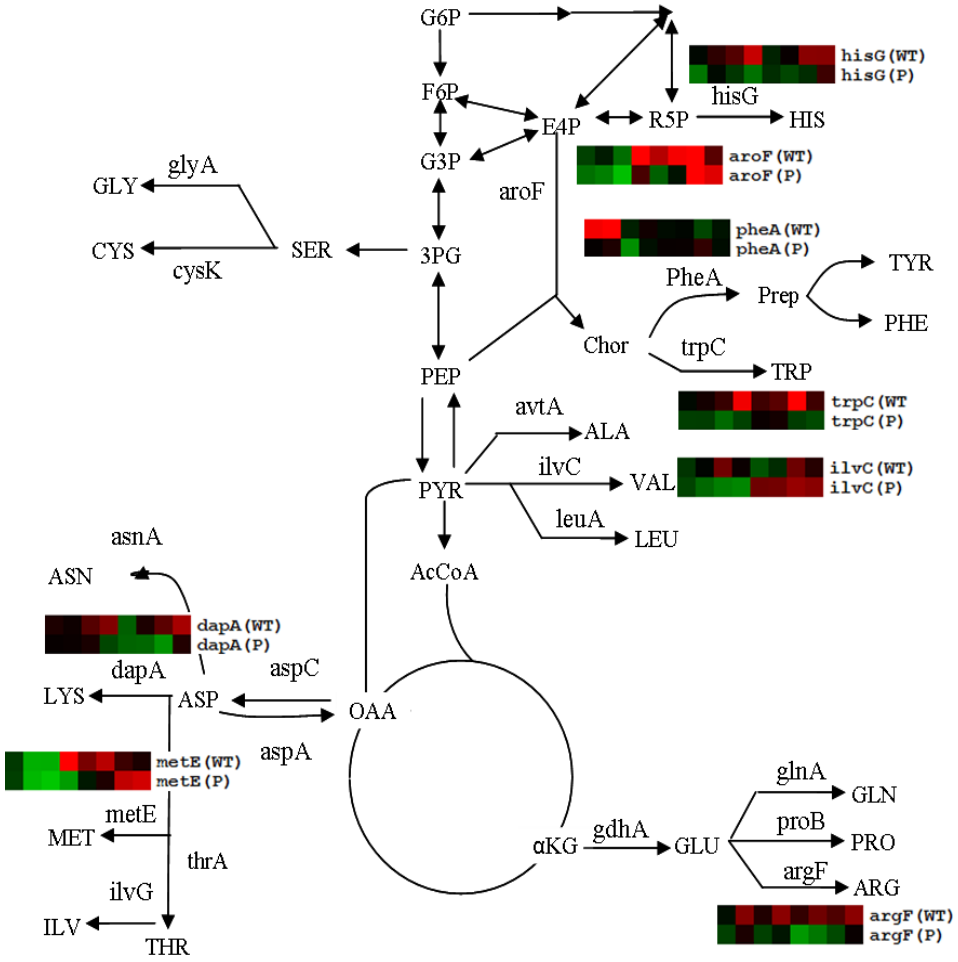

Fig. 9.7. The amino acid biosynthesis pathways of *Escherichia coli*

repressed. Aromatic amino acid production starts with erythrose-4-phosphate (E4P) and ends with the production of phenylalanine, tyrosine, and tryptophan. The enzymes *aroF*, *pheA* and *trpC* catalyze the intermediate steps in this pathway. Down-regulation of these genes decreased the aromatic amino acids production. The biosynthesis of amino acids valine, lysine, methionine and arginine is reduced in the PB cells due to the down-regulation of *livC, dapA, metE* and *argF*, respectively.

The above mapping of DEG on the central metabolic network and amino acid synthesis pathways provides understanding of the effects of the plasmid on the host strain.

It also reveals genetic targets to improve the growth of PB cells. In this case, the 5 genes that are down-regulated in the PB cells compared to WT cells are the potential targets. The performance of PB cells can be enhanced by over-expression of these five genes. However, there are many more genes that are differentially expressed. Analyzing 534 genes individually is tedious. We need methods for organizing these genes in a systematic fashion so that more information can be extracted. The rest of the steps in the proposed framework work on the difference of scores of these 534 genes.

9.4 Clustering and finding number of clusters

Clustering is a method to organize the genes into groups such that genes within a group are more similar to each other. Comparison of clusters of genes between WT cells and PB cells reveals reprogramming of metabolism in the later due to metabolic burden. Clustering generally revels higher level information of metabolic reprogramming such as alteration of biosynthesis pathways, regulation of ribosomal proteins, amino acids and nucleotide synthesis. It also reveals the strategies used by the cells to cope up with stress and changes in transportation genes. Such higher level information about metabolic reprogramming is useful to decide steps for strain improvement.

Before employing any clustering algorithm on a dataset, it is necessary to find the number of 'natural' clusters present in the data since clustering algorithms requires the number of clusters to be specified *a priori*. So the cluster validation procedure, NIFTI, described in Chapter 6 is employed on the 534 differentially expressed genes identified above. The results are shown in Figure 9.8. NIFTI using k-means clustering and Eu-

clidean distance as distance metric identifies 4 clusters in this dataset.

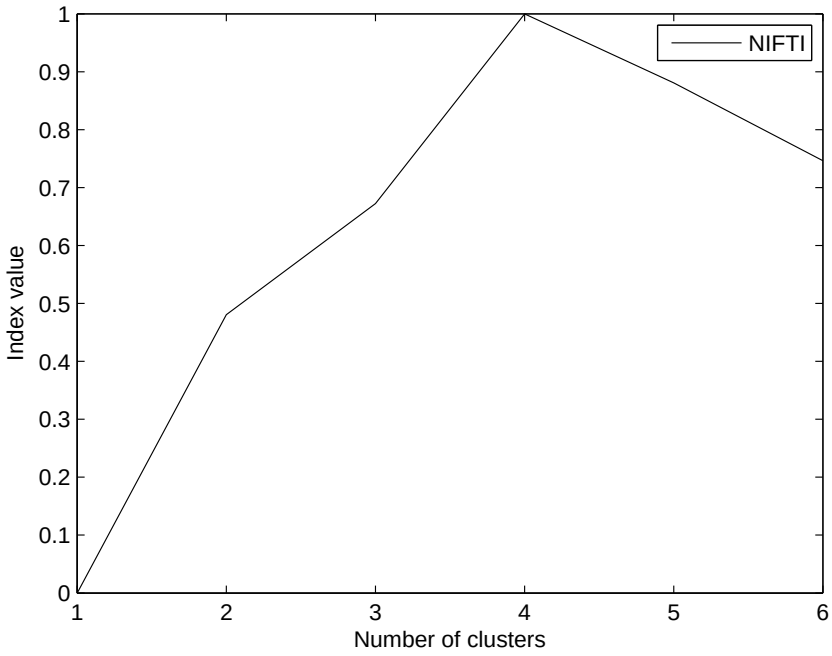

Fig. 9.8. Cluster validation results for differentially expressed genes in *Escherichia coli*

Next, the clustering method described in Chapter 5 is used cluster the 534 genes. The clustering algorithm uses Genetic Algorithms (GA) to cluster the genes based on the distance metric described in Section 5.2. The population size and the mutation probability are selected as 100 and 0.01, respectively. The number of generations is selected as 300. The performance of the GA in minimizing the objective function for clustering is shown in Figure 9.9. The objective function is constant after 19 iterations indicating that a minimum has been reached.

The heatmap of the four clusters are shown in Figure 9.10. From the heatmap, it is clear that the genes having similar expression profiles are clustered together. The mean

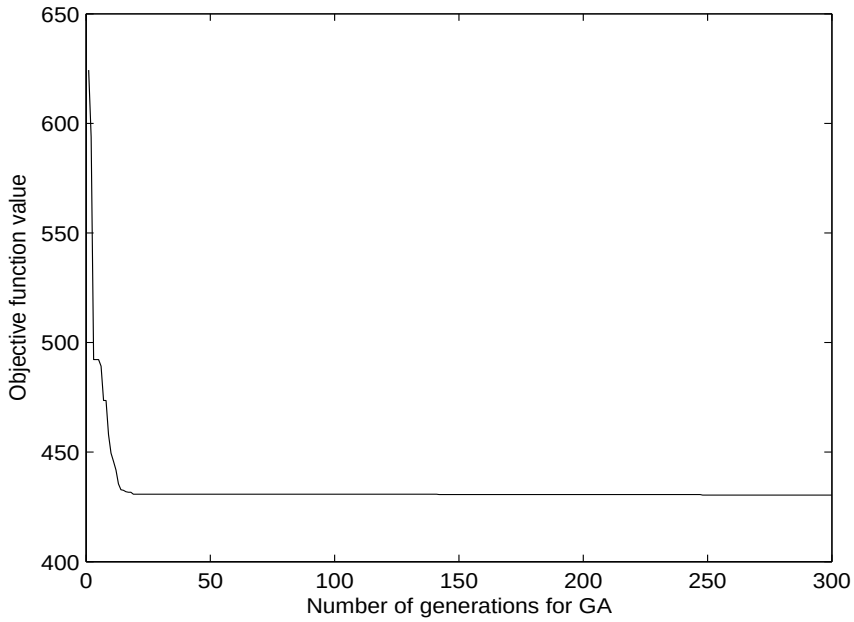

Fig. 9.9. Performance of GA for clustering differentially expressed genes

expression profile of the four clusters are shown in Figure 9.11. The mean expression profile for WT are shown as solid lines and that of plasmid are shown in dash line. A significant difference in mean expression profiles can be observed from Figure 9.11. These clusters can be analyzed further.

Cluster 1 contains 10 genes which have low expression in WT strain but have high expression in PB strain. This cluster contains transporter genes such as *yejF*, *ynfM*, and the Ferric uptake regulator *fur*. The up-regulation may be due to the incapability of strain to produce all the compounds required for biosynthesis. This cluster also contains genes related to protein degradation such as *tdcA*, *tdcG*. During exponential growth, WT cells do not degrade proteins. However, when amino acid production is decreased, higher degradation of amino acids is observed (Sussman and Gilvarg, 1969).

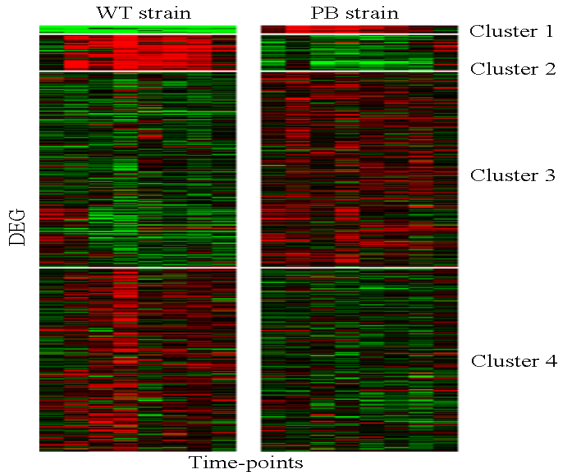

Fig. 9.10. Heatmap of differentially expressed genes. Clusters are enriched with similarly expressed genes.

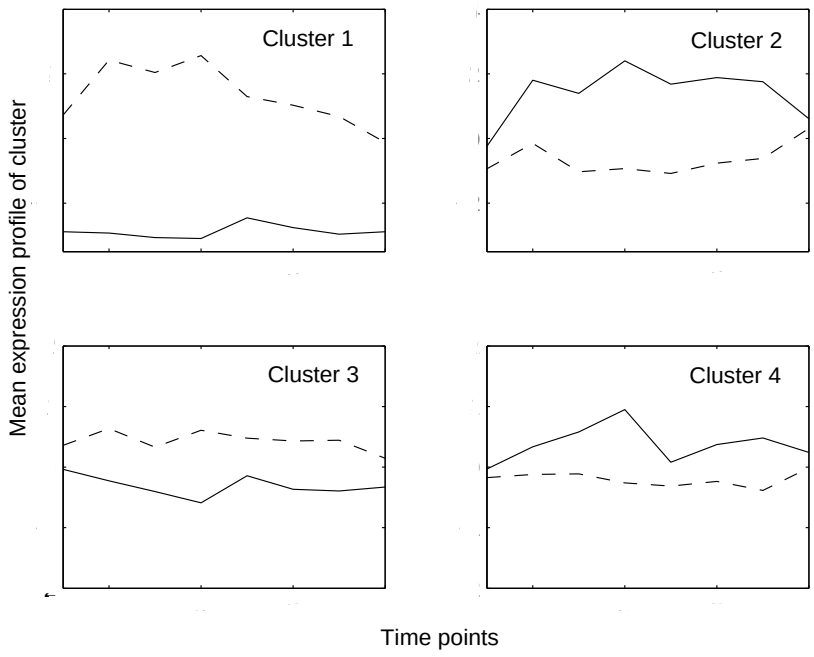

Fig. 9.11. Mean expression profiles of clusters. Solid lines represent the expression profiles of WT strain and dash lines represent the plasmid strain.

This phenomena also indicates that the cell is not capable of producing all compounds

it needs for growth.

Genes in Cluster 2 are expressed in large quantities in WT strain and repressed in plasmid strain. This cluster contains genes related to acid resistance response of cells. These genes include *gadA, gadB, gadC, gadE, gadX, and gadW*. It also contains *ydeO* which is a Transcription Factor that activates other acid resistance genes including *gadA* family. This cluster of genes explains the survival of WT strain during the low pH due to the presence of acetate in the medium. WT strain expressed the genes to resist the acidic condition of the growth medium. On the other hand, these acid resistance genes are down regulated in PB strain which is the reason for its low growth.

Cluster 3 comprises genes that are expressed in low levels in WT strain but over-expressed in PB strain. Similar to cluster 1, this cluster also contains genes related to transport. The difference between this cluster and the cluster 1 is the magnitude of the gene activation (Figure 9.11). The genes included in this cluster are *lacY, manY, malE, malK and melB*. It also contains *crp* which is a global regulator for many transporter genes (Keseler *et al.*, 2005).

The fourth cluster contains genes that are up-regulated in WT strains but down-regulated in PB strain. This cluster contains genes involved in utilization and degradation of several metabolic intermediates whose products are precursors for biosynthesis of other molecules. Well known genes in this cluster include *pgi* and *fbaB* from glycolysis pathway, and *tktB* from Phosphate Pentose pathway. Other genes include those participating in degradation of glycolate *(adlA)*, glyoxylate *(glcB)*, lysine *(ldcC)* , ala-

nine *(dadA)*, galactitol *(gatY)*, ethanol *(adhE)*, glycerol *(glpK)*, etc. All the biodegradation genes are down-regulated in PB cells leading to unavailability of energy and building blocks for biosynthesis which ultimately would affect the growth rate.

One of the important gene identified in cluster 4 is *acs* which is involved in acetate utilization. Since acetate production hinders the growth of *Escherichia coli* cells, it is interesting to identify the differences between the WT and PB cells in resisting the low pH due to increase in acetate concentration. The expression profile (data was slightly modified) of *acs* is shown in Figure 9.12.

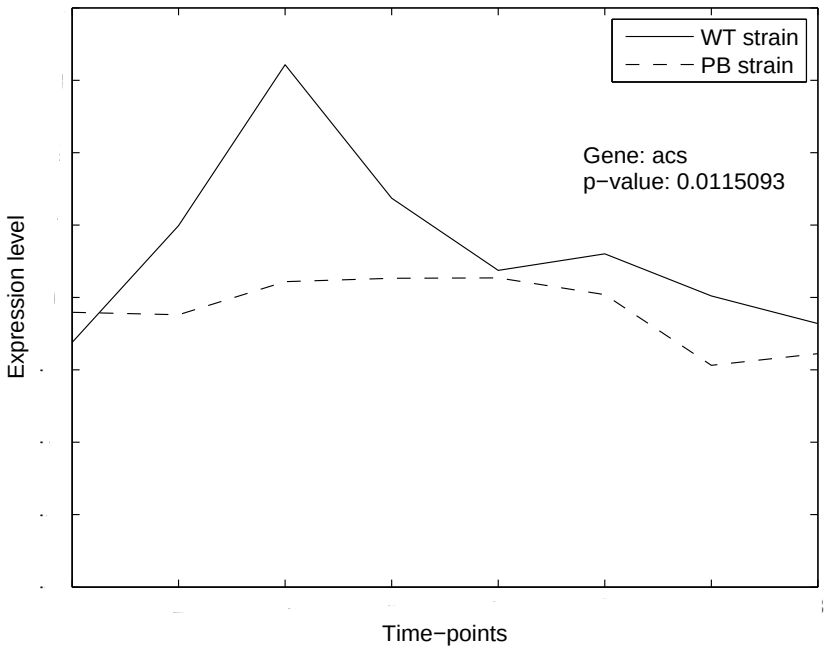

Fig. 9.12. Expression profile of the acetate utilization gene *acs*. Solid lines represent the expression profile of WT strain and dash line represent the plasmid strain.

As discussed above, during the fermentation of the WT cells, the concentration of Acetate gradually increase from start of exponential growth phase. To come up with

this acid medium, acid resistance genes are up-regulated in WT cells protecting the cells from low pH. Also, as the glucose concentration depletes, WT cells expressed the *ace* gene whose product converts the Acetate to AcCoA which feeds into the TCA cycle. This is the reason for decrease in the concentration of Acetate during WT fermentation. During fed-batch phase feeding is controlled and no acetate is produced. In contrast, the acid resistance genes are down-regulated in PB strain that reduces the growth rate. Also, the down-regulation of *acs* indicates that Acetate is not used as carbon source by PB strain.

In summary, through clustering the differentially expressed genes it is evident that (1) genes involved in catabolic reactions in glycolysis, PPP, TCA cycle and degradation of other carbon sources are down regulated in PB cells (Cluster 4). (2) as a consequence of this, PB cells loose their capability to produce building blocks for biosynthesis, hence transporter genes and amino acids degradation genes are activated (Cluster 1 and 3). (3) acid resistance genes are down-regulated in PB cells making them susceptible to low pH (Cluster 2).

9.5 Integration of TF-gene data and gene expression data

Integration of clustering results with TF-gene interaction data gives global regulators that regulate sets of genes. Identifying such global TFs is important since modifying a few of TFs brings the effect of modifying all the genes they regulate.

The next step in the data-driven approach for target selection is integration of TF-gene interaction data. The Bayesian approach described in Chapter 8 can be used to

reliably assign TFs (regulators) to genes. Then correlational analysis will be used to identify the global TFs. Though the genome-wide location data is not available for *Escherichia coli*, the TF-gene interaction network is available due to the vast amount of literature of *Escherichia coli*. Here, we use the RegulonDB database for this information (Gama-Castro *et al.*, 2008). RegulonDB contains information of 155 regulators (TFs) and their regulated genes. Our of these 155 TFs, gene expression data is available for 143 TFs. These 143 TFs are used for further analysis.

Out of the 143 TFs, 22 shows difference in expression between the WT and PB strain. The expression profiles (data were slightly modified) of these TFs are shown in Figure 9.13. All these TFs and other genes found to be differentially expressed are genetic targets that can be modified (over-expressed or down-regulated) to improve the PB strain. However, modifying a large number of genes is not desirable considering the experimental complexity. In the next step, the correlation between these TFs and the four gene clusters along with the understanding of cell functioning will be used to identify a reduced set of genetic targets.

The mean correlation between the 22 differentially expressed TFs in PB cell is given in Table 9.1. Since, gene expression data does not differentiate between direct regulation of gene by a TF or indirect regulation, we also find the number of genes that each of these 22 TFs bind to. This number is given in parenthesis along with the mean correlation. Mean correlation is important, as it cancels the noise associated with individual genes. Some TFs, called activators, activate the expression of genes whereas some TFs, called repressor, repress gene expression. There are some TFs that can work

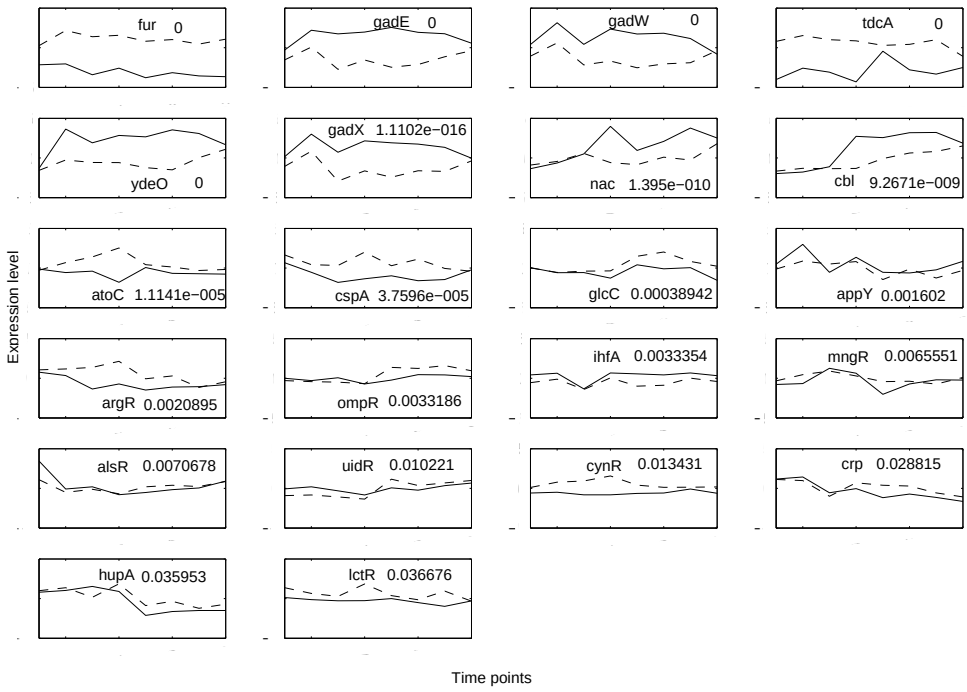

Fig. 9.13. Expression profile of TFs differentially expressed in PB strain compared to WT strain. TF names and corresponding p-values are also shown. Solid lines represent the expression profile in WT strain and dash line in PB strain.

as both activator and repressor based on the condition. The kind of action of each TF is given in the last column of the Table 9.1.

Considering the expression correlation and binding information, *tdcA* is selected as a global regulator of cluster 1. It has a correlation value of 0.978 and binds to 2 of the 10 genes in cluster 1. *tdcA* is an activator of operon containing *tdc* genes involved in amino acids degradation. As mentioned above, during starvation, cells degrade amino acids for manufacturing other compounds (Sussman and Gilvarg, 1969). *tdcA* is not expressed in WT cells but over-expressed in PB strain.

Table 9.1
Average correlation of differentially expressed TFs to four clusters

TF name	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Action
tdcA	**0.978(2)**	-0.9194(0)	0.7031(0)	-0.617(0)	Activator
gadE	-0.9434(0)	**0.9532(7)**	-0.704(2)	0.6233(1)	Activator
gadX	-0.9634(0)	**0.9613(7)**	-0.714(0)	0.6321(2)	Activator
ydeO	-0.9609(0)	**0.9517(1)**	-0.710(0)	0.6101(0)	Activator
fur	0.9578(1)	-0.9573(1)	**0.7113(4)**	-0.62(6)	Dual
crp	0.5928(3)	-0.6802(5)	**0.475(28)**	-0.40(30)	Dual
gadW	-0.9736(0)	0.95539(4)	-0.716(0)	0.616(0)	Repressor
nac	-0.8601(0)	0.9061(1)	-0.654(0)	0.6416(1)	Dual
cbl	-0.8607(0)	0.9101(0)	-0.657(1)	0.6292(1)	Activator
atoC	0.8249(0)	-0.795(0)	0.5978(0)	-0.612(1)	Activator
cspA	0.9654(0)	-0.9475(0)	0.7087(0)	-0.658(0)	Activator
glcC	0.7431(0)	-0.8185(0)	0.5813(1)	-0.528(1)	Dual
appY	-0.7822(0)	0.6914(0)	-0.549(0)	0.363(0)	Activator
argR	0.9149(0)	-0.9008(0)	0.6716(5)	-0.656(2)	Dual
ompR	0.4579(0)	-0.549(1)	0.3777(3)	-0.292(2)	Dual
ihfA	-0.8398(0)	0.8285(0)	-0.621(0)	0.4723(0)	Unknown
mngR	0.4163(0)	-0.3555(0)	0.291(1)	-0.087(0)	Repressor
alsR	-0.14(0)	-0.0346(0)	-0.038(0)	-0.074(2)	Repressor
uidR	-0.2428(0)	0.0944(0)	-0.121(0)	0.11(1)	Repressor
cynR	0.9552(0)	-0.9133(0)	0.691(1)	-0.642(0)	Dual
hupA	0.7291(0)	-0.7437(0)	0.550(0)	-0.403(0)	Dual
lctR	0.9705(0)	-0.9168(0)	0.699(1)	-0.628(0)	Repressor

gadE, gadX and ydeO are the global regulators for Cluster 2. Cluster 2 is very important as it contains the genes related to acid resistance. *gadE* and *gadX* are glutamate decarboxylase-dependent acid resistance transcription factors which activate genes related to acid resistance. *ydeO* also belongs to the same family and activates *gadE* which in turn activates *gadX* (Ma *et al.*, 2003).

The ferric uptake regulator *fur* and *crp* are considered as regulators for Cluster 3. Though their expression correlation is not high, they bind to more genes from Cluster

3. They are also known as transporter gene regulators (Keseler *et al.*, 2005).

No regulators are evident for Cluster 4 through correlational analysis and TF-gene binding information. This is not a surprise since Cluster 4 contains genes related to degradation of intermediate metabolites in central and other metabolic pathways. The central metabolic pathway is regulated via complex interactions that provide redundancy to pathways, so the organism can survive even if some TFs are mutated. So, it may not be possible to identify regulators for the whole cluster.

In summary, the growth of PB cells is lower than the corresponding WT cells without plasmid. The gene expression analysis identifies the reasons for this as (1) down-regulation of glycolysis and TCA cycle genes, (2) down-regulation of amino acid production, (3) incapability of PB strain to withstand low pH caused by high acetate, and (4) inability of the PB strain to utilize acetate as a substrate.

The gene targets identified through the proposed data-driven framework are *pgi, fbaB tktB, acnA, icd, ydeO* and *acs*. All these genes have to be over-expressed to increase the growth rate of PB cells. Up-regulation of *pgi, fbaB tktB, acnA, icd* is required to enhance the glucose uptake rate in glycolysis and subsequent conversion to energy and building blocks for other macromolecules. Experimental analysis of increase of glucose uptake rate by inactivating global regulator *FruR* shows 60% recovery of growth rate in PB cells (Ow *et al.*, 2007). Inactivation of *FruR* up-regulated several genes including *fbaA* which functions similar to *fbaB*. This clearly shows that the genetic targets identified proposed method are useful for strain improvement. The

experiment in which *FruR* is inactivated is carried only batch phase. Increase in acetate concentration is observed in the experiment. The proposed framework recommended over expression of *ydeO* and *acs* in order to cope up with decrease in PH and use of acetate as carbon source, respectively. Though the experiment with PB strain with overexpression of *ydeO* is not done yet, other studies have been reported expression of acid resistance genes when *ydeO* was overexpressed (Masuda and Church, 2003). The expression of *ydeO* could also lead to expression of *acs* indirectly (Rahman and Shimizu, 2008; Rahman *et al.*, 2006; Keseler *et al.*, 2005). Hence, *ydeO* should be over-expressed first followed by phenotype analysis and subsequent over-expression of other genes if necessary. Some other gene targets from amino acid biosynthesis are also important. However, the down-regulation of amino acid biosynthesis may be due to reduced uptake rate of TCA cycle. So, these genes should be considered after increasing the expression of TCA cycle genes.

9.6 Discussion and Conclusions

The proposed data-mining framework is used for identifying genetic targets for the improvement of *Escherichia coli* strain producing recombinant protein. The proposed framework contains different data-mining techniques for extracting information from gene expression data. Each data-mining technique provides different and complementary information about the functioning of cells which leads to identification of targets for strain improvement. In this case study, the proposed method for identifying DEG is able to identify the biologically significant genes which explain the phenotype of PB strain. Methods for clustering and finding number of clusters are also able to group the genes correctly. The correlation analysis and gene binding information correctly pre-

dicted TFs for clusters leading to reduced set of targets for strain improvement. The predictions from the framework have to be experimentally tested to further validate and refine the framework.

The proposed framework combines methods for identifying DEG, clustering and finding number of cluster in a systematic way which makes the framework suitable for target selection for strain improvement. However, the quality and availability of the data is an important issue to use this framework. The gene expression data can be generated relatively easily. But generation the genome wide location data is difficult (Ren *et al.*, 2000). Currently, the genome scale location data is currently available for yeast *Saccharomyces cerevisiae* only. Lack of such data hinders the identification of key regulator genes for strain improvement for other organisms. In such cases, the available literature related to the organism of interest should be compiled to get the TF-gene interaction data. Other important issue is the noise in genome scale datasets. Though the genome scale data provides global view of cell functioning, they are associated with high noise levels which reduce the accuracy of predictions (Kothapalli *et al.*, 2002). Improvement of technology and careful design of experiments is necessary to improve the quality of data.

10. CONCLUSIONS AND FUTURE WORK

In this thesis, different statistical data-mining methods have been proposed and validated. These methods include identification of differentially expressed genes in time-course gene expression datasets, clustering of gene expression profiles and cluster validation methods. These data-mining methods comprise the prime data-mining methods for gene expression data analysis. A Bayesian approach for integration of gene expression data and genome-wide location data has been proposed. All these methods were validated separately using artificial and real gene expression datasets and results are compared with other methods for the same purpose. The proposed methods have shown reasonably good performance in all the case studies. Finally, these methods were combined in a principled way to identify genetic targets for strain improvement. Here, the possible extensions of proposed method to further improve their performance are discussed.

10.1 Conclusions

In Chapter 4, a PCA based method was proposed for identifying differentially expressed genes in time-course gene expression data measured between two different conditions. For this purpose, a model is developed for the expression data from one of the datasets (generally the control condition) using the dominant PCs. The expression data from the other condition is projected onto the model. A hypothesis test using Mahalanobis distance is used to evaluate the significance of the differences in the scores to identify differentially expressed genes. The proposed method is validated using two real gene expression datasets. In both cases, the proposed method identified differen-

tially expressed genes which are biologically significant.

Clustering is an important aspect of gene expression data analysis. Clustering organizes a large number of genes into a few clusters such that genes within a cluster are more similar to each other. It gives the overall view of reprogramming of the gene expression due to change in biological condition of cells. A novel clustering method has been proposed and validated in Chapter 5. The proposed method identifies ellipsoidal clusters in gene expression. The proposed method uses PCA and divides the gene expression space into *PCA subspace* and *residual subspace*. The *PCA subspace* is spanned by dominant PCs and residual space is spanned by the remaining PCs. The division of original space facilitates identification of ellipsoidal clusters even if their covariance matrices becomes singular as in the case of gene expression data clustering. To address the issues with local minima in optimizing the clustering objective function, the proposed method uses Genetic Algorithms for minimizing the objective function. The proposed clustering method is validated using real gene expression datasets. The results are compared with already published results. The proposed method successfully identifies groups of genes that are functionally enriched.

Cluster validation methods identify number of clusters, k, in the dataset. Identification of number of clusters is important as many clustering algorithms require this to be specified a priori. A wrong specification leads to incorrect results. Two different cluster validation procedures, namely NIFTI and NEPSI, were proposed in chapter 6 and 7, respectively. NIFTI evaluates a cluster partition based on separability of resultant clusters. A statistical test is proposed for testing the separability of clusters. NIFTI in-

creases if clusters are separable and decreases otherwise. NEPSI finds the maximum number of distinct clusters in the data. PCA is used for determining whether a cluster is distinct or not. NEPSI increases if a cluster is distinct and decreases otherwise. In both methods, the value of the indices are calculated for different number of clusters and the k corresponding to the maximum value the of index is selected. Both NIFTI and NEPSI are validated using the gene expression data and results are compared with literature and results from other methods. Both the methods correctly identify number of clusters in gene expression data and outperform other methods.

In Chapter 8, a Bayesian approach for integration of gene expression data with TF-genes interaction data was proposed and validated using real genomic datasets. The proposed method models genes for which TFs are known from TF-gene interaction data and uses the developed models to predict the TFs for genes of unknown TFs using their expression similarity to modeled genes. The proposed Bayesian approach is used to combine yeast *Saccharomyces Cerevisiae* gene expression data and genome-wide location data. The proposed method correctly assigns TFs to genes for which no TFs are currently available.

The data-mining methods described in this thesis are combined systematically to identify genetic targets for strain improvements. A case study of this data-driven framework is give in Chapter 9 where genetic targets for improvement of growth rate of plasmid bearing strain. The data-mining tools provide information to understand the functioning of cell and subsequently leads to identification of genetic targets.

10.2 Future work

In this section, the possible extensions of methods proposed in this thesis are discussed.

The most important extension of proposed method for identifying DEG is to include replicates information. Replicates are very important in gene expression data analysis due to the inherent variability of gene expression data (Lee *et al.*, 2000). Replicates are of two types, biological replicates and technical replicates. Biological replicates indicates the samples collected from different populations of the same organism maintained at same conditions. Technical replicates indicate multiple experiments from the same sample. Replicates allows comparison variation in gene expression within each group and between groups and improve the reliability of identifying differentially expressed genes. The idea of using within and between group variation should be included in PCA analysis. The Multiway Principal Component Analysis (MPCA) which is routinely used to analyze data from multiple batches could be used (with modifications) to explicitly include replicates.

Another important improvement necessary is to improve the estimation of covariance matrix for the Mahalanobis distance calculation. Estimations of Covariance is prone to outliers in the data. Differentially expressed genes are outliers in the PCA scores data. Hence, the covariance matrix is affected by these outliers. Methods are available for estimation of robust covariance matrix which is not effected by outliers in the data (Rousseeuw and Leroy, 1987). These methods use re-sampling approach

and identify minimum volume ellipsoid that capture predefined (say 75%) of the data points in multidimensional space. The covariance matrix corresponding to minimum volume ellipsoid is un-effected by outliers as outliers are excluded from analysis. The eigen-values of robust covariance matrix are generally smaller than eigen-values of covariance matrix estimated from whole sample data. Hence, the proposed significance test for identifying differentially expressed genes becomes more sensitive and identifies more genes as differentially expressed. This could increase the quality of results.

The first improvement suggested for clustering technique is to extend this method to identify clusters of different sizes along with different shapes. One of the constraints used in optimizing the objective function is to fix the cluster volume to 1. This constraint is necessary to have a non-trivial solution. However, this constraint forces the optimizer to identify clusters of equal volume. The possibility of using reduced distance which are independent of cluster volumes can be explored in the future. Another improvement is to reduce the computational time for clustering using deterministic optimization algorithms.

One of the problems associated with cluster validation methods is the selection of maximum value of k to be tested. NIFTI checks whether offspring of a parent cluster overlap or not by modeling them as spheroids. NIFTI increases if there is no overlap, and decreases otherwise. For large values of k, no overlap can be detected even when a natural cluster is artificial broken. At the theoretical maximum number of clusters *i.e* when each gene is identified as a cluster, no overlap can be detected as the radii of all the clusters is zero. The phenomena of a continuous increase in NIFTI at large

value of k has been observed in some tests. Similar problem occurs for NEPSI also. The possible improvement for these methods is to add a new component that penalizes large values of k similar to penalizing higher order curves in regression. However, the penalty component should be selected in such as way that it does not hinder these methods from finding the correct number of clusters at the small values of k.

The proposed Bayesian approach is used to assign TFs to genes for yeast *Saccharomyces Cerevisiae* by combining gene expression data and genome-wide location data. The proposed Bayesian method assigned all the genes with no TFs in location data to one of the predefined classes based on their posterior probability. Even though the results are reasonably correct, it is preferable to develop criterion to reject genes in case no significant evidence is available for classification. From Figure 8.2, it is evident that some of the genes have the maximum posterior probability less than 0.5 indicating that they do not have significant evidence to be assigned to any class. Hence, it is better not to assign these genes to any of the classes. Also, the proposed method needs regulator information for some genes which is used to find TFs for other genes. Such information can also be extracted from other sources, such as literature search and promoter sequence analysis. Since the proposed Bayesian approach uses any new evidence to convert *a priori* probabilities to *a posterior*, it is relatively easy to extend this method to other complementary datasets. For example, if we know that a particular gene has a regulatory element similar to that of a set of other genes, we can use this as additional evidence (similar to expression profile similarity). The same procedure can be used for other complimentary data. These points have to be explored in future.

The data-mining methods described in this thesis are combined systematically to identify genetic targets for strain improvements. One of the bottleneck for such approaches is collecting information of individual genes that show differential expression between WT and PB strain. The redundancies in of metabolic network connectivity and presence of isoenzymes provide cells with the capability to exhibit a large number of phenotypes. This makes deriving of conclusions very difficult. It is necessary to have more information about genes. Integration of a text mining tool that searches relevant literature and provides information about genes would increase the understanding of cell function. Apart from this, the connectivity of metabolic networks seems to play a major role in selecting genetic targets. So, integration of graph-theoretic approaches for checking the connectivity and redundancy of metabolic networks are also helpful. Finally, development of a software package with a graphical user interface for data analysis and visualization is essential for successful genetic target selection by biologists.

Bibliography

Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Lossos, I. S., Rosenwald, A., Boldrick, J. C., Sabet, H., Tran, T., Yu, X., Powell, J. I., Yang, L., Marti, G. E., Moore, T., Hudson, J., Lu, L., Lewis, D. B., Tibshirani, R., Sherlock, G., Chan, W. C., Greiner, T. C., Weisenburger, D. D., Armitage, J. O., Warnke, R., Levy, R., Wilson, W., Grever, M. R., Bird, J. C., Botstein, D., Brown, P. O., and Staudt, M. (2000). Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.

Alter, O., Brown, P. O., and Botstein, D. (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, **97**, 10101–10106.

Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nature Genetics*, **25**, 25–29.

Babuska, R., Van Der Veen, P. J., and Kaymak, U. (2002). Improved covariance estimation for gustafson-kessel clustering. *Proceedings of IEEE International Conference on Fuzzy Systems*, **2**, 1081–1085.

Bachmann, R. (2005). Making the bio-based economy happen: changes and successful management approaches in the chemical industry. *Renewable Resources Biorefineries conference*, pages 19–21.

Banfield, J. D. and Raftery, A. E. (1993). Model-based gaussian and non-gaussian clustering. *Biometrics*, **49**, 803–821.

Bar-Joseph, Z., Gerber, G., Simon, I., Gifford, D. K., and S, J. T. (2003a). Comparing the continuous representation of time-series expression profiles to identify differentially expressed genes. *Proceedings of the National Academy of Sciences*, **100**, 10146–10151.

Bar-Joseph, Z., Gerber, G. K., Rinaldi, N. J., Yoo, J. Y., Robert, F., Gordon, D. B., Fraenkel, E., Jaakkola, T. S., Young, R. A., and Gifford, D. K. (2003b). Computational discovery of gene modules and regulatory networks. *Nature Biotechnology*, **21**, 1337–1342.

Bar-Joseph, Z., Gerber, G. K., Gifford, D. K., Jaakkola, T. S., and Simon, I. (2003c). Continuous representation of time series gene expression data. *Journal of Computational Biology*, **10**, 341–356.

Bartlett, M. S. (1950). Tests of significance in factor analysis. *The British journal of Psychology*, **3**, 77–85.

Ben-Hur, A., Elisieeff, A., and Guyon, I. (2002). A stability based method for discovering structure in clustered data. In *Pacific Symposium of Biocomputing*, pages 6–17.

Bentley, W. E., Mirjalili, N., Andersen, D. C., Davis, R. H., and Kampala, D. S. (1990). Plasmid-encoded protein: the principal factor in the metabolic burden associated with recombinant bacteria. *Biotechnology and Bioengineering*, **35**, 668–681.

Beranova-Giorgianni, S. (2003). Proteome analysis by two dimensional gel electrophoresis and mass spectrometry: strengths and limitations. *Trends in Analytical Chemistry*, **22**, 273–281.

Bezdek, J. C. and Pal, N. R. (1998). Some new indexes of cluster validity. *IEEE Transactions on Systems, Man and Cybernetics B*, **28**, 301–315.

Bolshakova, N. and Azuaje, F. (2003). Cluster validation techniques for genome expression data. *Signal Processing*, **83**, 825–833.

Brauer, M. J., Saldanha, A. J., Dolinski, K., and Botstein, D. (2005). Homeostatic adjustment and metabolic remodeling in glucose-limited yeast cultures. *Molecular Biology of the Cell*, **16**, 2503–2517.

Bro, C. and Nielsen, J. (2004). Impact of 'ome' analyses on inverse metabolic engineering. *Metabolic Engineering*, **6**, 204–211.

Brown, P. O. and Botstein, D. (1999). Exploring the new world of genome with dna microarrays. *Nature Genetics*, **21**, 33–37.

Calvano, S. E., Xiao, W., Richards, D. R., Felciano, R. M., Baker, H. V., Cho, R. J., Chen, R. O., Brownstein, B. H., Perren Cobb, J., Tschoeke, S. K., Miller-Graziano, C., Moldawer, L. L., Mindrinos, M. N., Davis, R. W., Tompkins, R. G., and Lowry, S. F. (2005). A network-based analysis of systemic inflammation in humans. *Nature*, **437**, 1032–1037.

Cheng, C., Ma, X., Yan, X., Sun, F., and Li, L. (2006). Mard: A new method to detect differential gene expression in treatment-control time courses. *Bioinformatics*, **22**, 2650–2657.

Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J., and Davis, R. W. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Biology of the Cell*, **2**, 65–73.

Choi, J. H., Lee, S. J., Lee, S. J., and Lee, S. Y. (2003). Enhanced production of insulin-like growth factor i fusion protein in escherichia coli by coexpression of the down-regulated genes identified by transcriptome profiling. *Applied and Environmental Microbiology*, **69**, 4737–4742.

Choi, J. H., Keum, K. C., and Lee, S. Y. (2006). Production of recombinant proteins by high cell density culture of escherichia coli. *Chemical Engineering Science*, **61**, 876–885.

Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Bostein, D., Brown, P. O., and Herskowitz, I. (1998). The transcriptional program of sporulation in budding yeast. *Science*, **282**, 699–705.

Conesa, A., Nueda, M. J., Ferrer, A., and Talon, M. (2006). masigpro : a method to identify significantly differential expression profiles in time-course microarray experiments. *Bioinformatics*, **22**, 1096–1102.

Davies, D. L. and Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **1**, 224–227.

Demain, A. L. (2000). Small bugs, big business: The economic power of the microbe. *Biotechnology Advances*, **18**, 499–514.

Dembele, D. and Kastner, P. (2003). Fuzzy c-means method for clustering microarray data. *Bioinformatics*, **19**, 973–980.

DeRisi, J. L., Iyer, V. R., and Brown, P. O. (1997). Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

Diaz-Ricci, J. C., Bode, J., Il Rhee, J., and Schugerl, K. (1995). Gene expression enhancement due to plasmid maintenance. *Journal of Bacteriology*, **177**, 6684–6687.

Duda, R. O. and Hart, M. P. (1973). *Pattern classification and scene analysis*. Wiley, NY.

Dudoit, S. and Fridlyand, J. (2002). A prediction-based resampling method to estimate the number of clusters in a dataset. *Genome Biology*, **3**, RESEARCH0036.

Dunn, J. (1974). Well separated clusters and optimal fuzzy partitions. *Cybernetics and Systems*, **4**, 95–104.

Edwards, J. S. and Palsson, B. O. (2000). The escherichia coli mg1655 in silico metabolic genotype: Its definition, characteristics, and capabilities. *Proceedings of the National Academy of Sciences*, **97**, 5528–5533.

Efron, B., Tibshirani, R., Storey, J. D., and Tusher, V. (2001). Empirical bayes analysis of a microarray experiment. *Journal of the American Statistical Association*, **96**, 1151–1160.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, **95**, 14863–14868.

El-Mansi, E. M. T. and Holms, W. H. (1989). Control of carbon flux to acetate excretion during growth of e. coli in batch and continuous culture. *Journal of General Microbiology*, **135**, 2875–2883.

Farmer, W. R. and Liao, J. C. (1997). Reduction of aerobic acetate production by escherichia coli. *Applied and Environmental Microbiology*, **63**, 3205–3210.

Fielden, M. R., Matthews, J. B., Fertuck, K. C., Halgren, R. G., and Zacharewski, T. R. (2002). In silico approaches to mechanistic predictive toxicology: An introduction to bioinformatics to toxicologists. *Critical reviews in toxicology*, **32**, 67–112.

Fraley, C. and Raftery, A. E. (1999). Mclust: Software for model-based cluster analysis. *Journal of Classification*, **16**, 297–306.

Friedman, N., Linial, M., Nachman, I., and Pe'er, D. (2000). Using bayesian networks to analyze expression data. *Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)*, pages 127–135.

Fuhrman, S., Cunningham, M. J., Wen, X., Zweiger, G., Seilhamer, J. J., and Somogyi, R. (2000). The application of shannon entropy in the identification of putative drug targets. *BioSystems*, **55**, 5–14.

Futcher, B. (2002). Transcriptional regulatory networks and the yeast cell cycle. *Current Opinion in Cell Biology*, **14**, 676–683.

Gama-Castro, S., Jacinto, V. J., Peralta-Gil, M., Santos-Zavaleta, A., Pealoza-Spindola, M. I., Contreras-Moreira, B., Segura-Salazar, J., Rascado, L. M., Martinez-Flores,

I., Salgado, H., Bonavides-Martinez, C., Abreu-Goodger, C., Rodriguez-Penagos, C., Miranda-Rios, J., Morett, E., Merino, E., Huerta, A. M., and Collado-Vides, J. (2008). Regulondb (version 6.0): gene regulation model of escherichia coli k-12 beyond transcription, active (experimental) annotated promoters and textpresso navigation. *Nucleic Acids Research*, **36**, Database issue:D120–D124.

Gath, I. and Geva, A. B. (1989). Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, 773–780.

Gibbons, D. F. and Roth, F. (2002). Judging the quality of gene expression-based clustering methods using gene annotation. *Genome Research*, **12**, 1574–1581.

Gordon, A. D. (1999). *Classification*. Chapman and Hall/CRC, Boca Raton.

Gustafson, D. E. and Kessel, W. C. (1979). Fuzzy clustering with a fuzzy covariance matrix. *IEEE Conference on Decision and Control*, **17**, 761–766.

Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of Intelligent Information Systems*, **17**, 107–145.

Hartemink, A. J., Gifford, D. K., Jaakkola, T. S., and Young, R. A. (2001). Combing location and expression data for principled discovery of genetic regulatory network models. *Pacific Symposium on Biocomputing*, pages 422–433.

Hartigan, J. A. (1975). *Clustering algorithms*. Wiley, New York.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, MI.

Holter, N. S., Mitra, M., Maritan, A., Cieplak, M., Banavar, J. R., and Fedoroff, N. V. (2000). Fundamental patterns underlying gene expression profiles: simplicity from complexity. *Proceedings of the National Academy of Sciences*, **97**, 8409–8414.

Iacobuzio-Donahue, C., Maitra, A., Olsen, M., Lowe, A. W., Van Heek, N. T., Rosty, C., Walter, K., Sato, N., Parker, A., Ashfaq, R., Jaffee, E., Ryu, B., Jones, J., Esh-

leman, J. R., Yeo, C. J., Cameron, J. L., Kern, S. E., Hruban, R. H., Brown, P. O., and Goggins, M. (2003). Exploration of global gene expression patterns in pancreatic adenocarcinoma using cdna microarrays. *American Journal of Pathology*, **162**, 1151–1162.

Ideker, T., Thorsson, V., Ranish, J. A., Christmas, R., Buhler, J., Eng, J. K., Bumgarner, R., Goodlett, D. R., Aebersold, R., and Hood, L. (2001). Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. *Science*, **292**, 929–934.

Issel-Tarver, L., Christie, K., Dolinski, K., Andrada, R., Balakrishnan, R., Ball, C. A., Binkley, G., Dong, S., Dwight, S. S., and Fisk, D. G. (2002). Saccharomyces genome database. *Methods in Enzymology*, **350**, 329–346.

Iyer, V. R., Eisen, M. B., Ross, D. T., Schuler, G., Moore, T., Lee, J. C. F., Trent, J. M., Staudt, L. M., Hudson, J. J., Boguski, M. S., Lashkari, D., Shalon, D., Botstein, D., and Brown, P. O. (1999). The transcriptional program in the response of human fibroblasts to serum. *Science*, **283**, 83–87.

Iyer, V. R., Horak, C. E., Scafe, C. S., Botstein, D., Snyder, M., and Brown, P. O. (2001). Genomic binding sites of the yeast cell-cycle transcription factors sbf and mbf. *Nature*, **409**, 533–538.

Jackson, J. E. (1991). *A User's Guide to Principal Components*. John Wiley, NY.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys*, **31**, 264–323.

Jiang, D., Pei, J., and Zhang, A. (2003). Dhc: A density-based hierarchical clustering method for time-series gene expression data. *Proceedings of Third IEEE Symposium on Bioinformatics and Bioengineering*, pages 393– 400.

Jiang, D., Tang, C., and Zhang, A. (2004). Cluster analysis for gene expression data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, **16**, 1370–1386.

Jonnalagadda, S. and Srinivasan, R. (2004). An information theory approach for validating clusters in microarray data. *Presented in Intelligent Systems for Molecular Biology ISMB*.

Kabir, M. M. and Shimizu, K. (2003). Gene expression patterns for metabolic pathway in pgi knockout escherichia coli with and without phb genes based on rt-pcr. *Journal of Biotechnology*, **105**, 11–31.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of American Statistical Association*, **90**, 773–795.

Kerr, M. K. and Churchill, G. A. (2001.). Statistical design and the analysis of gene expression microarray data. *Genetical Research*, **77**, 123–128.

Keseler, I. M., Collado-Vides, J., Gama-Castro, S., Ingraham, J., Paley, S., Paulsen, I. T., Peralta-Gil1, M., and Karp, P. D. (2005). Ecocyc: a comprehensive database resource for escherichia coli. *Nucleic Acids Research*, **33**, D334–D337.

Kim, D. W., Lee, K. H., and Lee, D. (2005). Detecting clusters of different geometrical shapes in microarray gene expression data. *Bioinformatics*, **21**, 1927–1934.

Koch, C., Schleiffer, A., Ammerer, G., and Nasmyth, K. (1996). Switching transcription on and off during the yeast cell cycle: Cln/cdc28 kinases activate bound transcription factor sbf (swi4/swi6) at start, whereas clb/cdc28 kinases displace it from the promoter in g2. *Genes and Development*, **10**, 129–141.

Koranda, M., Schleiffer, A., Endler, L., and Ammerer, G. (2000). Forkhead-like transcription factors recruit ndd1 to the chromatin of g2/m specific promoters. *Nature*, **406**, 94–98.

Kothapalli, R., Yoder, S. J., Mane, S., and Loughran, T. (2002). Microarray results: how accurate are they? *BMC Bioinformatics*, **http://www.biomedcentral.com/1471-2105/3/22**.

Krishna, K. and Murty, M. N. (1999). Genetic k-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics*, **29**, 433–439.

Krishnapuram, R. and Kim, J. (1999). A note on the gustafson-kessel and adaptive fuzzy clustering algorithms. *IEEE Transactions on Fuzzy Systems*, **7**, 453–461.

Krzanowski, W. J. (1979). Between-groups comparison of principal components. *Journal of American Statistical Association*, **74**, 703–707.

Lander, E. S. (1996). The new genomics: global views of biology. *Science*, **274**, 536–539.

Leach, S. and Hunter, L. (2000). Comparative study of clustering techniques for gene expression microarray data. *Presented in Fourth Annual International Conference on Computational Molecular Biology, RECOMB*.

Lee, M. L. T., Kuo, F. C., Whitmorei, G. A., and Sklar, J. (2000). Importance of replication in microarray gene expression studies: Statistical methods and evidence from repetitive cdna hybridizations. *Proceesings of the National Academy of Sciences*, **97**, 9834–9839.

Lee, S. Y., Lee, D. Y., and Kim, T. Y. (2005). Systems biotechnology for strain improvement. *Trends in Biotechnology*, **23**, 349–358.

Lee, T. I. and Young, R. A. (2000). Transcription of eukaryotic protein-coding genes. *Annual Review of Genetics*, **34**, 77–137.

Li, H., Zhang, K., and Jiang, T. (2004). Minimum entropy clustering and applications to gene expression data. In *Proceedings of IEEE Computational Systems Bioinformatics Conference (CSB 04)*, pages 142–151.

Lipshutz, R. J., Fodor, S. P. A., Gingeras, T. R., and Lockhart, D. J. (1999). High density synthetic oligonucleotide arrays. *Nature Genetics*, **21s**, 20–24.

Lukashin, A. V. and Fuchs, R. (2001). Analysis of temporal gene expression profiles: clustering by simulated annealing and determining the optimal number of clusters. *Bioinformatics*, **17**, 405–414.

Ma, Z., Gong, S., Richard, H., Tucker, D. L., Conway, T., and Foster, J. W. (2003). Gade (yhie) activates glutamate decarboxylase-dependent acid resistance in escherichia coli k-12. *Molecular Microbiology*, **49**, 1309–1320.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability*, **1**, 281–297.

Mandelstam, J., McQuillen, K., and Dawes, I. (1982). *Biochemistry of bacterial growth*. Blackwell Scientific Publications, NY.

Mao, J. and Jain, A. K. (1996). A self-organizing network for hyperellipsoidal clustering. *IEEE Transactions on Neural Networks*, **7**, 16–29.

Masuda, N. and Church, G. M. (2003). Regulatory network of acid resistance genes in escherichia coli. *Molecular Microbiology*, **48**, 699–712.

McMillan, D. R., Xiao, X., Shao, L., Graves, K., and Benjamin, I. J. (1998). Targeted distruption of heat shock transcription factor 1 abolishes thermotolerance and protection against heat-inducible apoptosis. *Journal of Biological Chemistry*, **273**, 7523–7528.

Milligan, G. W. and Cooper, M. C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, **50**, 159–179.

Nasmyth, K. and Dirick, L. (1991). The role of swi4 and swi6 in the activity of g1 cyclins in yeast. *Cell*, **66**, 995–1013.

Nau, G. J., Richmond, J. F. L., Schlesinger, A., Jennings, E. G., Lander, E. S., and Young, R. A. (2002). Human macrophage activation programs induced by bacterial pathogens. *Proceedings of the National Academy of Sciences*, **99**, 1503–1508.

Nielsen, J. (1998). Metabolic engineering: techniques for analysis of targets for genetic manipulations. *Biotechnology and Bioengineering*, **58**, 125–132.

Ow, D. S. W., Nissom, P. M., Philp, R., Oha, A. K., and Yap, M. G. (2006). Global transcriptional analysis of metabolic burden due to plasmid maintenance in *escherichia coli* $dh5\alpha$ during batch fermentation. *Enzyme and Microbial Technology*, **39**, 391–398.

Ow, D. S. W., Lee, R. M., Nissom, P. M., Philp, R., Oh, S. K., and Yap, M. G. (2007). Inactivating frur global regulator in plasmid-bearing escherichia coli alters metabolic gene expression and improves growth rate. *Journal of Biotechnology*, **131**, 261–269.

Ow, D. S. W., Yap, M. G., and Oh, S. K. (2009). Enhancement of plasmid dna yields during fed-batch culture of a frur-knockout escherichia coli strain. *Biotechnology and Applied Biochemistry*, **52**, 53–59.

Pal, N. R. and Bezdek, J. C. (1995). On cluster validity for fuzzy c-means model. *IEEE Transactions on Fuzzy Systems*, **3**, 370–379.

Pan, W. (2002). A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. *Bioinformatics*, **18**, 546–554.

Pan, W., Lin, J., and Le, C. T. (2003). A mixture model approach to detecting differentially expressed genes with microarray data. *Functional and Integrative Genomics*, **3**, 117–124.

Park, T., Yi, S. G., Lee, S., Lee, S. Y., Yoo, D. H., Ahn, J., and Lee, Y. S. (2003). Statistical tests for identifying differentially expressed genes in time-course microarray experiments. *Bioinformatics*, **19**, 694–703.

Price, N. D., Reed, J. L., and Palsson, B. O. (2004). Genome-scale models of microbial cells: Evaluating the consequences of constraints. *Nature Reviews*, **2**, 886–897.

Rahman, M. and Shimizu, K. (2008). Altered acetate metabolism and biomass production in several escherichia coli mutants lacking rpos-dependent metabolic pathway genes. *Molecular BioSystems*, **4**, 160–169.

Rahman, M., Rubayet Hasan, M., Oba, T., and Shimizu, K. (2006). Effect of rpos gene knockout on the metabolism of escherichia coli during exponential growth phase and early stationary phase based on gene expressions, enzyme activities and intracellular metabolite concentrations. *Biotechnology and Bioengineering*, **94**, 585–595.

Raychaudhuri, S., Stuart, J. M., and Altman, R. B. (2000). Principal components analysis to summarize microarray experiments: application to sporulation time series. *Pacific Symposium on Biocomputing*, **5**, 452–463.

Redner, R. A. and Walker, H. F. (1984). Mixture densities, maximum likelihood and the em algorithm. *Society for Industrial and Applied Mathematics Review*, **26**, 195–239.

Ren, B., Robert, F., Wyrick, J. J., Aparicio, O., Jennings, E. G., Simon, I., Zeitlinger, J., Schrei-ber, J., Hannett, N., Kanin, E., Volkert, T. L., Wilson, C. J., Bell, S. P., and Young, R. A. (2000). Genome-wide location and function of dna binding proteins. *Science*, **290**, 2306–2309.

Reverter, A., Ingham, A., Lehnert, S. A., Tan, S. H., Wang, Y., Ratnakumar, A., and Dalrymple, B. P. (2006). Simultaneous identification of differential gene expression and connectvity in inflammation, adipogenesis and cancer. *Bioinformatics*, **22**, 2396–2404.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, **20**, 53–65.

Rousseeuw, P. J. and Leroy, A. (1987). *Robust Regression and Outlier Detection*. John Wiley, New York.

Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, **26**, 43–49.

Schafer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Bioligy*, **4**, Article 32.

Schena, M., Shalon, D., Davis, R. W., and Brown, P. O. (1995). Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, **270**, 467–470.

Segal, E., Shapira, M., Regev, A., Peer, D., Bostein, D., Koller, D., and Friedman, N. (2003). Module networks: identifying regulatory modules and their condition specific regulators from gene expression data. *Nature Genetics*, **34**, 166–176.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, **27**, 379–423.

Sharan, R., Moron-Katz, A., and Shamir, R. (2003). Click and expander: a system for clustering and visualizing gene expression data. *Bioinformatics*, **19**, 1787–1799.

Simon, I., Barnett, J., Hannett, N., Harbison, C. T., Rinaldi, N. J., Volkert, T. L., Wyrick, J. J., Zeitlinger, J., Gifford, D. K., Jaakkola, T. S., and Young, R. A. (2001). Serial regulation of transcriptional regulators in the yeast cell cycle. *Cell*, **106**, 697–708.

Singhal, A. and Seborg., D. (2002). Pattern matching in historical batch data using pca. *IEEE Control Systems Magazine*, **22**, 53–63.

Slonim, D. K. (2002). From patterns to pathways: gene expression data analysis comes of age. *Nature Genetics*, **32**, 502–508.

Small, N. J. H. (1978). Plotting squared radii. *Biometrika*, **65**, 657–658.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M., Brown, P. O., Botstein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, **9**, 3273–3297.

Srinivasan, R., Wang, C., Ho, W. K., and Lim, K. W. (2004). Dynamic principal component analysis based methodology for clustering process states in agile chemical plants. *Industrial and Engineering Chemistry Research*, **43**, 2123–2139.

Steinhoffand, C. and Vingron, M. (2006). Normalization and quantification of differential expression in gene expression microarrays. *Briefings In Bioinformatics*, **7**, 166–177.

Stephanopoulos, G. (2002). Metabolic engineering: perspective of a chemical engineer. *AIChE journal*, **48**, 920–926.

Storey, J. D., Xiao, W., Leek, J. T., Tompkins, R. G., and Davis, R. W. (2005). Significance analysis of time course microarray experiments. *Proceedings of the National Academy of Sciences*, **102**, 12837–12842.

Sussman, A. and Gilvarg, C. (1969). Protein turnover in amino acid-starved strains of escherichia coli k-12 differing in their ribonucleic acid control. *The Journal of Biological Chemistry*, **244**, 6304–6308.

Tabibiazar, R., Wagner, R. A., Ashley, E. A., King, J. Y., Ferrara, R., Spin, J. M., Sanan, D. A., Narasimhan, B., Tibshirani, R., Tsao, P. S., Efron, B., and T, Q. (2005). Signature patterns of gene expression in mouse atherosclerosis and their correlation to human coronary disease. *Physiological Genomics*, **22**, 213–226.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S., and Golub, T. R. (1999). Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Sciences*, **96**, 2907–2912.

Tavazoie, S., Huges, J. D., Campbell, M. J., Cho, R. J., and Church, G. M. (1999). Systematic determination of genetic network architecture. *Nature Genetics*, **22**, 281–285.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a dataset via gap statistic. *Journal of Royal Statistical Society B*, **63**, 411–423.

Trinklein, N. D., Murray, J. I., Hartman, S. J., Botstein, D., and Myers, R. M. (2004). The role of heat shock transcription factor 1 in the genome-wide regulation of the mammalian heat shock response. *Molecular Biology of the Cell*, **15**, 1254–1262.

Troyanskaya, O. G., Garber, M. E., Brown, P. O., Botstein, D., and Altman, R. B. (2002). Nonparametric methods for identifying differentially expressed genes in microarray data. *Bioinformatics*, **18**, 1454–1461.

Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences*, **98**, 5116–5121.

Van der Werf, M. J. (2005). Towards replacing closed with open target selection strategies. *Trends in Biotechnology*, **23**, 11–16.

Vinciotti, V., Liu1, X., Turk, R., Meijer, E. J., and Hoen, P. A. (2006). Exploiting the full power of temporal gene expression profiling through a new statistical test: Application to the analysis of muscular dystrophy data. *BMC Bioinformatics*, **7**, 183.

Walsh, G. (2006). Biopharmaceutical benchmarks 2006. *Nature Biotechnology*, **24**, 769–776.

Wicker, N., Dembele, D., Raffelsberger, W., and Poch, O. (2002). Density of points clustering, application to transcriptomic data analysis. *Nucleic Acids Research*, **30**, 3992–4000.

Wierckx, N. J. P., Ballerstedt, H., de Bont, J. A. M., de Winde, J. H., Ruijssenaars, H. J., and Wery, J. (2008). Transcriptome analysis of a phenol-producing pseudomonas putida s12 construct: genetic and physiological basis for improved production. *Journal of Bacteriology*, **190**, 2822–2830.

Wise, B. M. and Ricker, N. L. (1991). Recent advances in multivariate process control: Improving robustness and sensitivity. *IFAC Symposium on Advanced Control of Chemical Processes, Toulouse, France*.

Wodicka, L., Dong, H., Mittmann, M., Ho, M. H., and Lockhart, D. J. (1997). Genome-wide expression monitoring in saccharomyces cerevisiae. *Nature Biotechnology*, **15**, 1359–1367.

Wold, S. (1976). Pattern recognition by means of disjoint principal component models. *Pattern Recognition*, **8**, 127–139.

Yeung, K. Y., Fraley, C., Murua, A., Raftery, A. E., and Ruzzo, W. L. (2001). Model-based clustering and data transformations for gene expression data. *Bioinformatics*, **17**, 977–987.

Zhao, G. and Winkler, M. (1994). An escherichia coli k-12 tkta tktb mutant deficient in transketolase activity requires pyridoxine (vitamin b6) as well as the aromatic amino acids and vitamins for growth. *Journal of Biotechnology*, **176**, 6134–6138.

Zhu, G., Spellman, P. T., Volpe, T., Brown, P. O., Botstein, D., Davis, T. N., and Futcher, B. (2000). Two yeast forkhead genes regulate the cell cycle and pseudohyphal growth. *Nature*, **406**, 90–94.