

BIO
3941

Statistical Challenges for Imputing Missing Data in Bioinformatics



By

Saima Nawaz

*A thesis submitted in the partial
fulfilment of the requirements for the degree of*

MASTER OF PHILOSOPHY

In

BIOINFORMATICS

National Center for Bioinformatics

Faculty of Biological Sciences

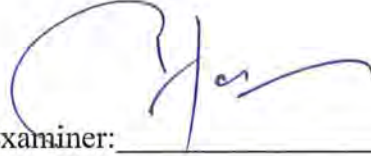
Quaid-i-Azam University

Islamabad, Pakistan

2015

CERTIFICATE

This thesis submitted by **Miss Saima Nawaz** from National Centre for Bioinformatics, Faculty of Biological Sciences, Quaid-i-Azam University, Islamabad, Pakistan, is accepted in its present form as satisfying the thesis requirement for the Degree of Master of Philosophy in Bioinformatics.



Internal Examiner: _____
Dr. Muhammad Naeem
Assistant Professor & Supervisor
Department of Biotechnology
Quaid-i-Azam University Islamabad



External Examiner: _____
Dr. Shaheen Shahzad
Department of Bioinformatics
International Islamic University,
Islamabad



Incharge: _____
Dr. Amir Ali Abbasi
Assistant Professor
National Centre for Bioinformatics
Quaid-i-Azam University Islamabad

Date:- *March 26, 2015*

Dedicated

To

*My beloved parents and siblings,
Because all that I am, or hope to be,
I owe my family*

DECLARATION

I Saima Nawaz, hereby declare that I have produced the work presented in this thesis, during the scheduled period of study. I also declare that I have not taken any material from any source except referred to wherever due. If a violation of HEC rules on research has occurred in this thesis, I shall be liable to punishable action under the plagiarism rules of the HEC.

Date: 4th May, 2015

Signature of the student:



Saima Nawaz

(02281311008)

CONTENTS

LIST OF FIGURESiv

LIST OF TABLESvi

LIST OF ABBREVIATIONSvii

LIST OF NOTATIONS.....viii

ACKNOWLEDGEMENTSix

ABSTRACT.....x

Chapter 1

INTRODUCTION..... 1

1.1 Background 1

1.2 Microarrays 2

1.2.1 Principle of Microarray 9

1.2.2 Applications of Microarray 9

1.2.3 Examining the Gene Expression 11

1.2.4 Microarray Expression Matrix 12

1.3 Missing Data 14

1.3.1 Occurrence of Missing Values 14

1.4 Microarray Data Processing and Image Analysis 16

1.4.1 Addressing 16

1.4.2 Segmentation..... 17

1.4.3 Information Extraction..... 17

1.4.4 Weak Spots or Missing Values 18

1.4.5 Expression Ratios (The Primary Comparison) 18

1.4.6 Normalization 18

1.5 Nature of Missing Data 19

1.5.1 Missing completely at random..... 19

1.5.2	Missing at random.....	19
1.5.3	Missing not at random	20
1.6	Statistical Problems due to Missing Data.....	20
1.7	Aim of Study.....	20
Chapter 2		
MATERIALS AND METHODS		21
2.1	Data Collection and Software	21
2.2	Drawbacks of Above Methods.....	22
2.3	Other statistical Methods to deal with Missing Values.....	22
2.3.1	K Nearest Neighbors (KNN) Imputation	22
2.3.2	Singular Value Decomposition Imputation	23
2.3.3	Bayesian Principal Component Analysis Imputation	25
2.3.4	Least Squares Imputation.....	26
2.3.5	Local Least Squares Imputation.....	27
2.3.6	missForest	28
2.4	Proposed Algorithm and its Working.....	29
2.4.1	Gibbs Bayesian Variable Selection Algorithm	29
2.4.2	Linear Regression to Estimate missing Values.....	31
2.4.3	Input	32
2.4.4	Imputing Missing Values by Mean Imputation	32
2.4.5	Gibbs Bayesian Variable Selection and Linear regression	33
2.4.6	Normalized root Mean Square Error.....	33
Chapter 3		
RESULTS		34
3.1	Results of Proposed Algorithm	34
3.2	Results of KNN.....	38
3.3	Results of SVD.....	39

3.4	Results of LLS Impute	40
3.5	Results of BPCA	42
3.6	Results of missForest	43
Chapter 4		
DISCUSSION		45
4.1	Conclusion.....	48
References		50
Webliography		56
Appendix.....		57

LIST OF FIGURES

Figure 1.1	Schematic Illustration of spotted genes on a glass slide array.....	3
Figure 1.2	A sketch of cDNA microarray technology.	5
Figure 1.3	A sketch showing initial steps of Oligonucleotide Genechip.	6
Figure 1.4	Six steps of microarray experiment.	8
Figure 1.5	Hybridization of two single stranded nucleic acid sequences to a double stranded helical complex.	9
Figure 1.6	Applications of Microarray.....	11
Figure 1.7	A $m \times n$ expression matrix X with m genes across n conditions and expression value of gene c in condition j is denoted as x_{ij}	12
Figure 1.8	A flow chart of a typical microarray experiment.....	15
Figure 1.9	Steps of Microarray image analysis and data processing.	16
Figure 2.1	Matrix of k most similar genes for the estimation of missing values.....	27
Figure 3.1	Results of proposed algorithm at 5% missing percentage.	35
Figure 3.2	Results of proposed algorithm at 10% missing percentage.	36
Figure 3.3	Results of proposed algorithm at 20% missing percentage.	36
Figure 3.4	Results of proposed algorithm at 40% missing percentage.	37
Figure 3.5	Results of proposed algorithm for given missing percentages (5%, 10%, 20%, 40%).....	37
Figure 3.6	Results of KNN for given missing percentages (5%, 10%, 20%, 40%).....	39
Figure 3.7	Results of SVD for all missing percentages (5%, 10%, 20%, 40%).....	40
Figure 3.8	Results of LLS impute for given missing percentages (5%, 10%, 20%, 40%).....	41
Figure 3.9	Results of BPCA for given missing percentages (5%, 10%, 20%, 40%).....	43
Figure 3.10	Results of missForest for given missing percentages (5%, 10%, 20%, 40%).....	44
Figure 4.1	Results of proposed algorithm along with increasing percentage of missing values.....	46
Figure 4.2	Comparison of results of already existing methods (KNN, SVD, BPCA, missForest) with the proposed algorithm (GibbsBvs+Reg)	47

Figure 4.3 Results of LLSImpute method for all percentages of missing values. .48

LIST OF TABLES

Table 1.1	Complete Microarray Data Matrix.	13
Table 1.2	Data matrix containing missing values.....	14
Table 2.1	Complete dataset.....	32
Table 3.1	NRMSE values of proposed algorithm for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).	35
Table 3.2	NRMSE values of KNN for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).....	38
Table 3.3	NRMSE values of SVD for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).....	39
Table 3.4	NRMSE values of LLS Impute for first 500 iterations for given missing percentages (5%, 10%, 20%, and 40%).....	41
Table 3.5	NRMSE values of BPCA for first 500 iterations for given missing percentages (5%, 10%, 20%, 40%).	42
Table 3.6	NRMSE values of missForest for first 500 iterations for given missing percentages (5%, 10%, 20%, 40%).	44
Table 4.1	Average NRMSE values of proposed algorithm (GibbsBvs + Regression), KNN, SVD, BPCA, missForest and LLSImpute, for given missing percentages (5%, 10%, 20%, and 40%)	46

LIST OF ABBREVIATIONS

AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
BPCA	Bayesian Principal Component Analysis
cDNA	Complementary DNA
C_p	Mallow's criterion
Cy3	Cyanine 3
Cy5	Cyanine 5
DNA	Deoxyribonucleic acid
GibbsBvs	Gibbs Bayesian Variable Selection
KNN	K Nearest Neighbours
LLS	Local Least Squares
LS	Least Squares
mRNA	messenger RNA
MVs	Missing values
MCAR	Missing Completely at Random
MCMC	Markov Chain Monte Carlo
MAR	Missing at Random
MNAR	Missing Not at Random
NRMSE	Normalized Root Mean Square Error
OOB	Out-of-Bag estimation error
PC	Principal Component
SVD	Singular Value decomposition
SNP	Single nucleotide polymorphism
SSVS	Stochastic Search Variable Selection
RF	Random Forest
RMSE	Root Mean Square Error
RNA	Ribonucleic acid

LIST OF NOTATIONS

X	potential predictors
Y	dependent variables
g	genes having missing values
n	number of experiments or samples
$V_{g,n}$	Missing value of gene in experiment
$V_{g,i}$	Missing value of gene on index i
k	genes having similar expression profiles
m	number of genes
i	index of gene
j	index of missing value
$p(\theta)$	prior distribution
σ^2	variance of samples
β	coefficient vector
γ	latent variable
ε	Error term
c_i	Prior parameters
τ_i	Prior parameters
w_i	Vector of entries not missing for gene
V^T	Matrix of eigengenes

ACKNOWLEDGEMENTS

In the name of **ALLAH**, the Most Gracious and the Most Compassionate, all praises to Him for giving me ability and courage to explore His bestowed world as a student of research. Also all praises to His **prophet (Hazrat Muhammad peace be upon him)** for enlightening our conscious with faith in Allah.

The successful completion of this dissertation was only made possible through valuable contribution of number of people. Formal way to say "Thank you" is not enough to pay my gratitude.

I express my sincere gratitude to my esteemed supervisor, **Dr. Muhammad Faisal**, for his encouragement, able guidance and support at each level, without which this dissertation would never have been completed. I am thankful for his timely and prompt response to my queries, new ideas and his enthusiasm to complete this research work. I am also thankful to **Dr. Muhammad Naeem** for his contribution finally to upgrade this thesis.

I am also thankful to **Dr. Wasim Ahmed**, Chairman of National Centre for Bioinformatics and Dean of Faculty of Biological Sciences, QAU, Islamabad and **Dr. Amir Ali Abbasi**, Incharge of National Centre for Bioinformatics, QAU, Islamabad, for being very kind and cooperative.

My appreciation extends to my supportive lab colleagues **Syed Aleem Haider** and **Raiha Mumtaz** and caring friends **Noor-us-Schar**, **Madiha Hafeez**, **Fouzia Altaf**, **Irum Javaid** and **Madiha Shabbir** for their kindness and moral support during my study. I pay heartiest gratitude to **Rabia Tahir** who guided me in every step of my research.

Finally, my family deserves special mention for their inseparable support and care. My loving father, **Ahmad Nawaz Malik**, caring and supportive mother, **Bilqees Nawaz** and siblings **Masood Ahmed**, **Bushra Almas**, **Aamir Nawaz** and **Uzma Nosheen** for their unconditional support and motivation throughout my studies. I would like to thank for all their prayers, patience and appreciation.

Saima Nawaz

ABSTRACT

The study of gene expression in cells and tissues has become a major tool nowadays in different fields such as identification of diseases and SNPs, development of drugs etc. DNA microarrays are being widely used for simultaneously measuring the expression levels of thousands of genes in a set of biological samples. The drawback of DNA microarray experiments is that they produce multiple missing expression values due to numerous reasons. Many algorithms developed for gene expression analysis require complete data matrix as an input. Hence to analyze the data it is necessary to estimate the missing values. Either row or column mean of the genes having missing values is incorporated at the place of missing values or the genes having missing values are to obtain a complete data matrix. This results in the deletion of important information required for analysis hence producing misleading results. Thus accurate methods are needed for the estimation of Missing Values. For this purpose a method using Gibbs Bayesian Variable Selection and Linear regression has been developed in the current study to estimate missing values. This method selects predictors i.e., genes having important impact upon the data and then calculates the missing values on the basis of linear regression. Normalized Root Mean Square Error was used as a metric for testing the accuracy of results generated by the developed algorithm. The Normalized Root Mean Square Error values show that the developed algorithm calculates missing values more accurately as compared to some other previously developed methods.

Chapter 1
Introduction

INTRODUCTION

1.1 Background

Cells present in any organism possess identical genetic material. Similar genes are not necessarily active in every cell. Scientists need to study the activity of genes in different types of cells in order to understand the normal function of cells as well as effect of abnormal gene function upon cells. It was challenging for scientists to conduct genetic analysis on large scale but with the advent of technology it became easy to carry out large scale genetic analysis. DNA has a structural arrangement similar to a ladder twisted into a helix.

Cells are known as building blocks or fundamental units of the living organism. Each cell consists of a nucleus which contains the chromosomes. Chromosomes carry the instructions that are needed to control and govern the cell activities resulting in the production of proteins by the help of DNA (deoxyribonucleic acid). DNA stores biological information of all living organisms. DNA has a structural arrangement similar to a ladder twisted into a helix. Molecules of sugar and phosphate form the sides of ladder while pairs of nucleotide bases joined by the hydrogen bonds form the rungs of ladder. The nucleotide bases include Adenines, Thymine, Cytosine, and Guanine. Adenine always pairs with thymine while Cytosine pairs with Guanine in base pairing. Each strand of double helix consists of nucleotide sequence. A nucleotide is made up of a molecule of sugar, one molecule of phosphate and one of the four bases. DNA sequence is specific order of the bases which are arranged along the sugar phosphate backbone. DNA sequence encodes the genetic code. The genetic code is actually a set of instructions by the help of which the information encoded in the genetic material (either DNA or RNA sequences) is translated to produce proteins in living organisms. The complete set of DNA of an organism is known as genome, containing all the information required to build and maintain an organism. The size of genomes vary widely across organisms. All the cells of an organism contain same DNA and hence same set of instructions even then cells are different from each other. The reason is that DNA segments activate in some particular conditions and not in other conditions. These DNA segments are known as genes and the process by which they activate is called their expression (Sebastiani et al., 2003).

The concept of gene expression emerged in 1961 along with the discovery of messenger RNA (mRNA). The gene expression level is defined as an integer value or the continuous measure hence providing a quantitative description of gene expression by measuring the amount of the intermediary molecules that are produced during this process. Expression profile of a gene is a set of expression levels that are measured for that gene across different conditions (Sebastiani et al., 2003).

Microarray technology offers a powerful approach for the analysis of gene expression on large scale. Thousands of different genes can be simultaneously analyzed using microarray in histological or cytological experiments. Microarrays can be used for wide variety of purposes including identification of SNPs, analysis of alternative RNA splicing, analysis of transcription factors binding to promoters, cancer classification, discovering the unknown gene function, identifying the effects of specific therapy, study of gene regulation, discovery of bio marker, diagnosis and prognosis of a disease and drug development. Microarray experiments are complex, time consuming and often very expensive. They generate large and complicated data sets that require significant effort to analyze and validate (Gan et al., 2006; Molaezadeh and Moradi, 2006; Yoon et al., 2007; Liew et al., 2011).

1.2 Microarrays

Microarray is a series of single stranded DNA molecules or target sequences that have been immobilized on to a carrier surface by the process of biochemical synthesis. Carrier is a solid surface and it can be a glass slide, a silica chip or a nylon membrane. Microarrays are also known as DNA chips, bio-chips, gene chips, DNA microarrays or simply the arrays. Most microarrays contain probes for 10,000 to 40,000 different genes. Microarrays may have test sites ranging from hundreds to many thousands of 10 to 500 microns size range. Test sites of high density microarrays can be upto 10^6 in an area of 1 to 2 cm². DNA target sequences are immobilized in an orderly and logical fashion on the solid surface. Nucleic acid probes which are derived from a diseased cell are then hybridized to target sequences (Heller, 2002). According to the nomenclature recommended by Phimister, 1999 “probe” is defined as a tethered nucleic acid having known sequence while “target” is defined as a free nucleic acid sample whose identity or abundance is to be detected. Series of steps are followed in a typical microarray experiment in a defined order of array fabrication, target

preparation, hybridization, image capture and data analysis. A successful microarray experiment requires all the steps to be performed consistently and accurately in order to produce reliable and significant results (Majtan et al., 2004; Liew et al., 2011).

Generally, two types of microarrays have been developed; Spotted DNA microarrays and Oligonucleotide genechips.

For Spotted DNA microarrays DNA probe may be single or double stranded DNA delivered on to the array.

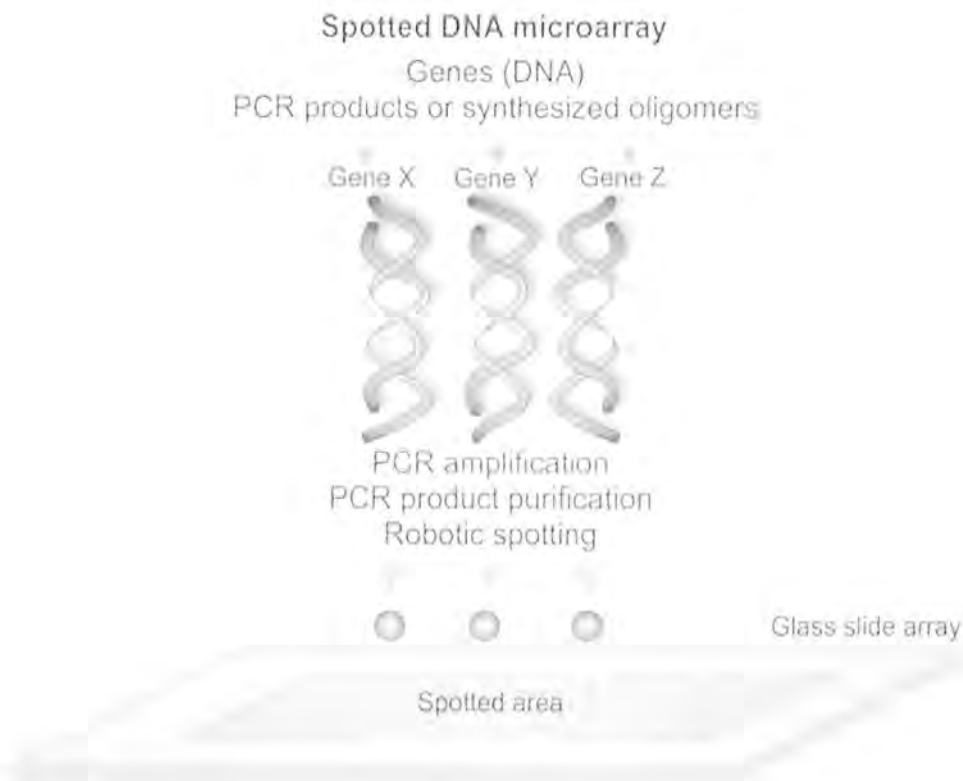


Figure 1.1 Schematic Illustration of spotted genes on a glass slide array.

Robotic spotters are used by glass slide arrays to spot genes on the glass slide. Each spot on the array represents a particular contiguous genes fragment. Adapted from Omidi et al., 2011.

Spotted DNA Microarrays are of two types; Spotted cDNA Microarrays and Oligonucleotide arrays.

Spotted cDNA microarrays consist of a solid surface (e.g., a microscope glass slide, a silica chip or a nylon membrane) upon which small amounts of nucleotide sequences are placed in grid like arrangement. Each spot represents a specific gene, an expressed sequence tag or a clone. Expressed Sequence tags are fragments of cDNA sequence which provides a tag for a gene whose full sequence or function is not known. Probes or drops of cDNA are printed on the surface of the array. Target mRNA is hybridized against spot which serves as a probe. The length of cDNA probe ranges from 500 to 2500 base pairs. DNA probe may be single or double stranded DNA delivered on to the array (Heller, 2002; Dubitzky et al., 2003; Sebastiani et al., 2003; Ehrenreich, 2006; Ness, 2006).

Sometimes significant amount of variation in size and shapes of spots occurs. These variations can occur among corresponding spots on different microarrays or between different spots present on the same microarray. Size of spot affects the probe amount that is available for hybridization. Shape of the spot affects the image analysis. Intensity of fluorescence labeling is affected by the inhomogeneous distribution of sample across the surface of cDNA microarray (Ness, 2006).

The use of cDNAs of longer length (greater than 300 base pairs) has its advantages and disadvantages. The hybridization of target and probe is strong due to longer length of cDNA. As a result there will be little or no effect of point mutations or small deletions on hybridization results. Hence large set of genes of human patients having minor differences among their genes can be studied. cDNA microarrays are less costly because a single PCR reaction generates enough purified DNA for the production of many thousand microarrays. Furthermore, due to long length of cDNA probes there is possibility of detection of all the transcripts produced either through alternative RNA splicing or alternative promoter use (Ness, 2006).

Considerably large amount of cost and effort is required for assembling large libraries correctly identified and annotated purified cDNA sequences. cDNA can hybridize to closely related gene families due to the possible presence of repeated sequences. Hence it does not provide enough specificity for many applications (Ness, 2006).

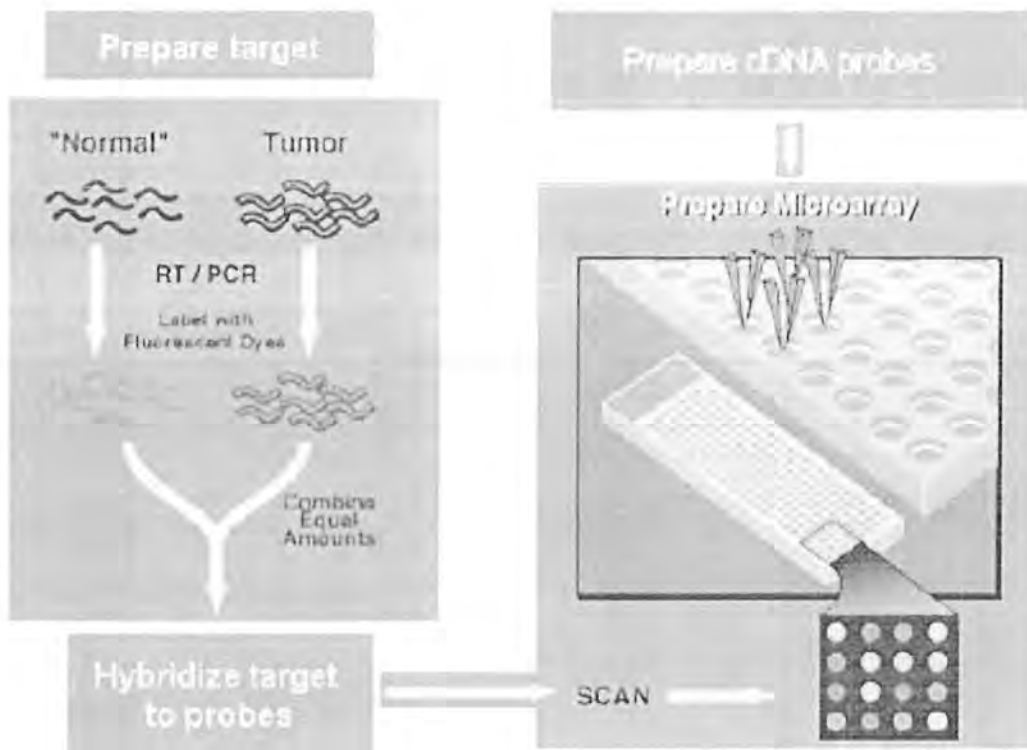


Figure 1.2 A sketch of cDNA microarray technology.

A robot is used to precisely apply tiny droplets containing clones of cDNA to the glass slides. Then fluorescent labels are attached to the mRNA extracted from the cell of interest. The labeled target is then allowed to hybridize with cDNA strands on slides. After the completion of hybridization is completed, scanning microscope measures the brightness of each fluorescence dot. Adapted from Sebastiani et al., 2003.

Oligonucleotide arrays use oligonucleotides as probes which range from 40 to 60 mer or 50 to 70 base pairs in length. Oligonucleotides are rather short fragments of single stranded DNA or RNA synthesized on the basis of sequence of existing genes (Heller, 2002; Majtan et al., 2004; Ehrenreich, 2006; Ness, 2006).

Properly designed oligonucleotides can help to overcome the specificity and cDNA GC content problems. The major problem is relatively high cost to purchase oligonucleotides and large scale bioinformatics support required to design specific DNA fragments with matched GC content for each gene. Hence the difference between spotted cDNA microarrays and oligonucleotide arrays is that spotted cDNA microarrays use predetermined probes while oligonucleotide arrays facilitates to design the probe sequences (Ness, 2006).

Oligonucleotide genechips or Affymetrix genechips are the commercially available oligonucleotide genechips that help to carry out parallel synthesis of oligonucleotides on large scale. DNA probe is synthesized *in-situ* on the surface of DNA chip. For the preparation of gene chips matched sets of short oligonucleotide pairs (currently 12 pairs of probe sets) are synthesized using a photolithographic process (Ness, 2007). Oligonucleotide pair consists of one perfect matched oligonucleotide and one with a single mismatch. Perfect matched probe has a sequence identical to that of the target gene (Dalma-Weiszhausz et al., 2006; Ness, 2007). The mismatch probe differs from perfect matched probe by a single base present in the middle of sequence. Oligonucleotides of the length 25 bases per probe are used by genechips. Each probe spot has diameter of approximately 18 micron meter hence facilitating almost 500,000 probes per array (Ehrenreich, 2006; Ness, 2007).

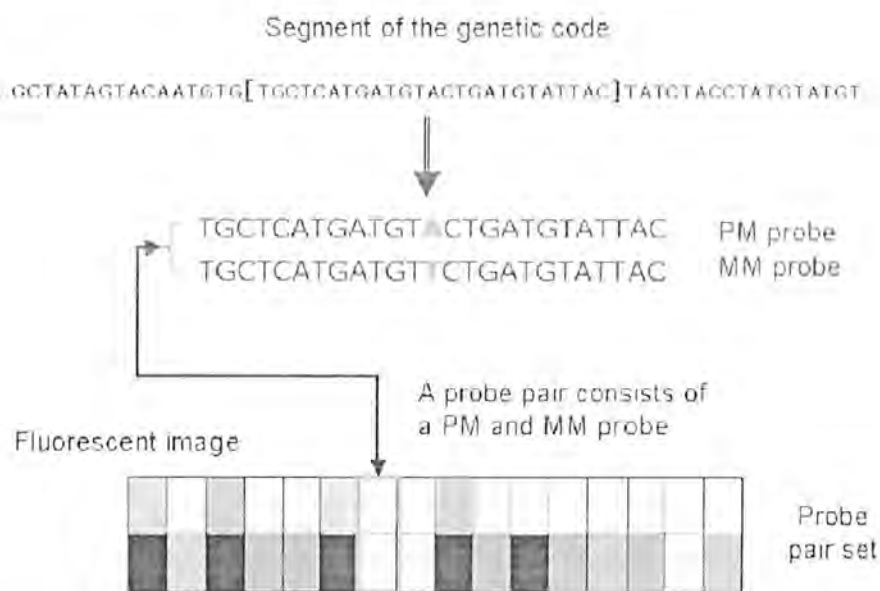


Figure 1.3 A sketch showing initial steps of Oligonucleotide Genechip.

PM probe is Perfect Matched probe having sequence identical to target gene. MM probe is Mismatch Probe having a difference of single gene from PM probe. Adapted from Sebastiani et al., 2003.

If a part of genechip surface is damaged even then enough probe sets will be readable to carry out the experiment. Furthermore, statistical analysis can be performed due to the presence of multiple probe sets, so that for each gene an expression level and p value of expression can be reported (Ness, 2006).

Following are the steps of typical microarray experiment as illustrated by Nguyen et al., (2002):

1. Samples of known probes are immobilized or spotted onto a glass slide or microarray, each spot corresponding to a gene or Expressed sequence Tags.
2. mRNA of interest was purified from cell populations. Pools of purified mRNA are then reverse transcribed into cDNA and then labeled with either red or green fluorescent dyes (mostly Cy3 or Cy5 is used) to distinguish between mRNAs of normal cells and treated or diseased cells.
3. These two populations of fluorescently labeled cDNAs are then combined and hybridized to the probes on the array. Unbound cDNA is washed off.
4. Next, the amount of hybridization that takes place is measured by a scanner.
5. This intensity is translated into a table with numerical measures and saved into a text file. This text file contains data about the level of fluorescence of each spot and background or foreground intensities. Foreground intensity corresponds to spots of interest in microarray. Background intensity is the noise resulting from high salt and detergent concentrations during hybridization of target.
6. These text files are then computationally and statistically analyzed according to the need of the researcher.

Brightness of each fluorescence spot reveals the amount of specific DNA fragment present in the target.

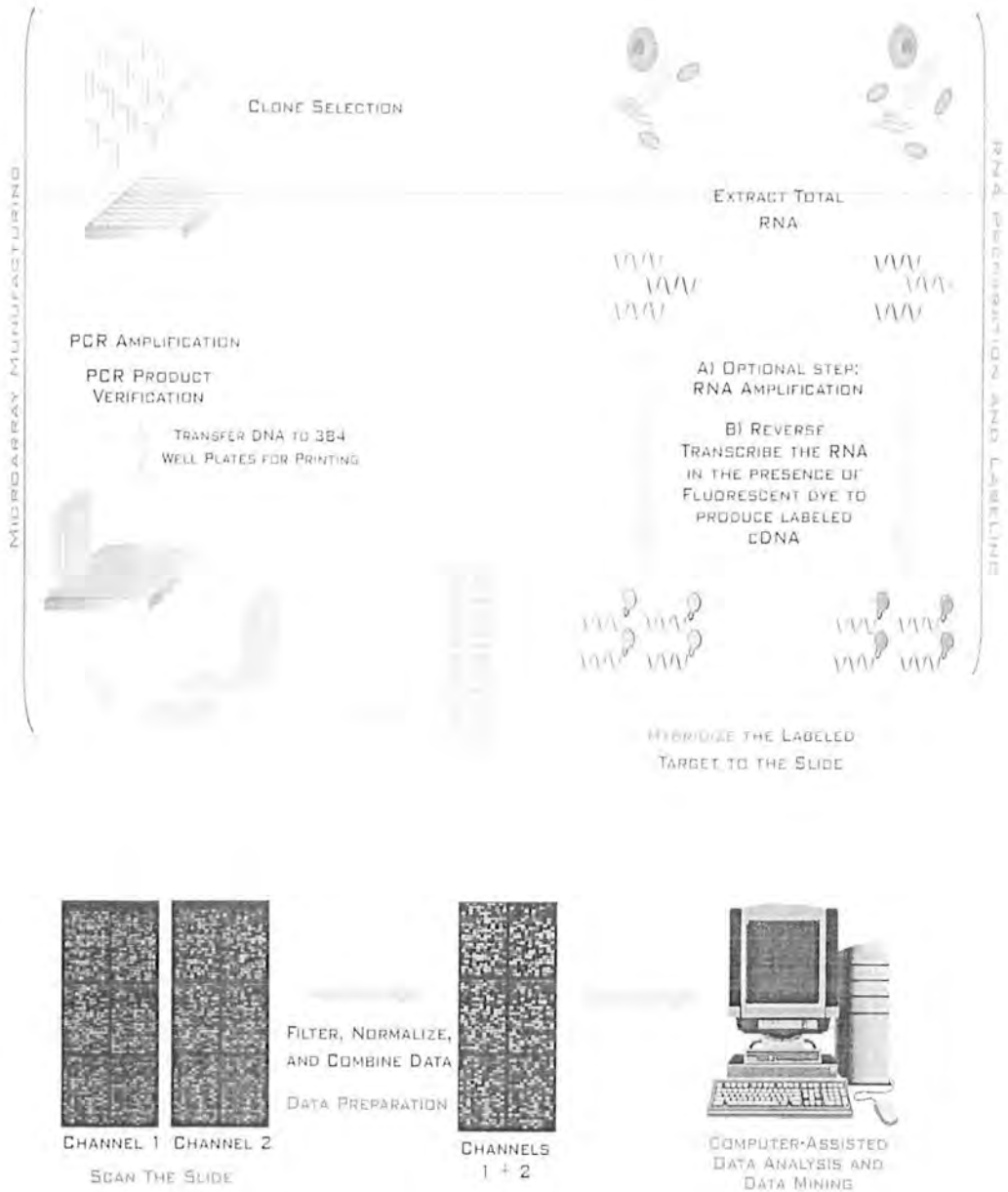


Figure 1.4 Six steps of microarray experiment.

Adapted from Macgregor et al., 2002.

1.2.1 Principle of Microarray

Microarrays function upon the principal of DNA hybridization. According to Watson and Crick rule, two DNA strands hybridize or pair with each other by the formation of hydrogen bonds between the complementary nucleotide base pairs. There would be tighter and stronger non covalent bonding between the complementary nucleotides if complementary base pairs are higher in number. Microarray chips are washed off after hybridization hence only strongly paired strands remain hybridized. Intensity of hybridization is measured by the intensity of the signal produced. Intensity of signal depends upon the amount of target sample hybridized to probe (Gabig and Wegrzyn, 2001).

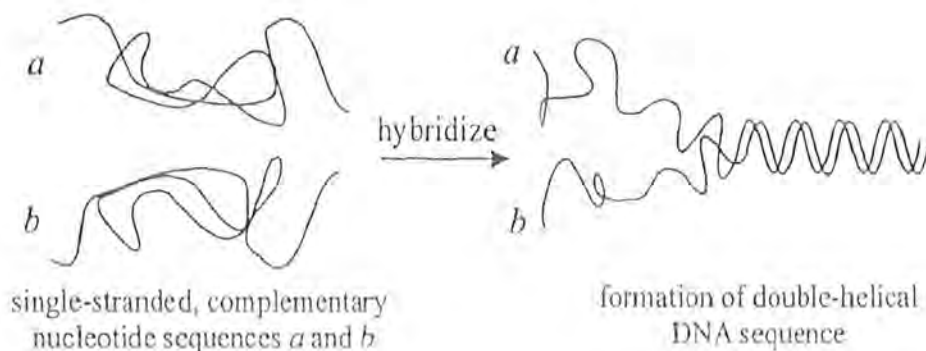


Figure 1.5 Hybridization of two single stranded nucleic acid sequences to a double stranded helical complex.

Microarrays work upon the principle of hybridization. Adapted from Werner Dubitzky, 2003

1.2.2 Applications of Microarray

Microarray experiment technique has been widely used in variety of biological studies including cancer classification, discovery of unknown genes, and identification of effects of a specific therapy and to measure the expression level of thousand of genes in a single experiment explained by Majtan et al., (2004).

1.2.2.1 Gene Discovery and Mapping or Gene Expression Profiles

For a particular organism, a specific “gene expression profile” can be produced at different developmental stages using microarrays. Hence different patterns that are specific for a growth condition, developmental stage or drug treatment are generated.

These profiles are a diagnostic tool for numerous applications such as drug discovery and fermentation process optimization.

1.2.2.2 Gene Regulation Studies

Based on the assumption that genes regulated in parallel share common control mechanisms, microarrays are helpful in identification of regulons (group of operons that are transcriptionally co-regulated by the same regulatory machinery) and description of complex cellular pathways.

1.2.2.3 Comparative Genomics and Genotyping

Microarrays serve as platform for genomic hybridization experiments of whole genome array. This helps in comparing genomes of closely related organisms, identification of virulence factors, exploration of molecular phylogeny, improvement of diagnostics, development of vaccines and identification of changes in genetic contents of same strain.

1.2.2.4 Drug Discovery

Microarray allows the comparison of expression of many genes between “disease” and “normal” tissues and cells. This is helpful in identification of multiple potential drug candidates.

1.2.2.5 Predicting Biochemical Pathways

Information generated by DNA microarrays can be helpful in prediction of biochemical pathways in several ways:

- Identification of genes involved in production process.
- Differences in genetic contents.
- Difference of expression profiles among wild type and improved strains.

1.2.2.6 Identification of SNPs

Single nucleotide polymorphisms are the most occurring mutations in DNA sequence. They occur at about one per 500–1000 base pairs in the human genome. Different approaches have been developed for the detection of SNPs by DNA microarrays being helpful in association studies of complex diseases, pharmacogenetics.

population genetics and physical mapping (Snijders et al., 2000; Miller and Tang, 2009).

1.2.2.7 Pathogen Detection

DNA microarray technology is most commonly used in epidemiological and clinical importance for the detection of pathogen. Apart from the identification of species and strains, microarrays are being used to characterize specific characteristics of microbial pathogens. This ability of microarray includes investigation and study of microbial virulence factors and antibiotic resistance genes using signature sequences and characteristic genes (Kostrzynska and Bachand, 2006; Rasooly and Herold, 2008).

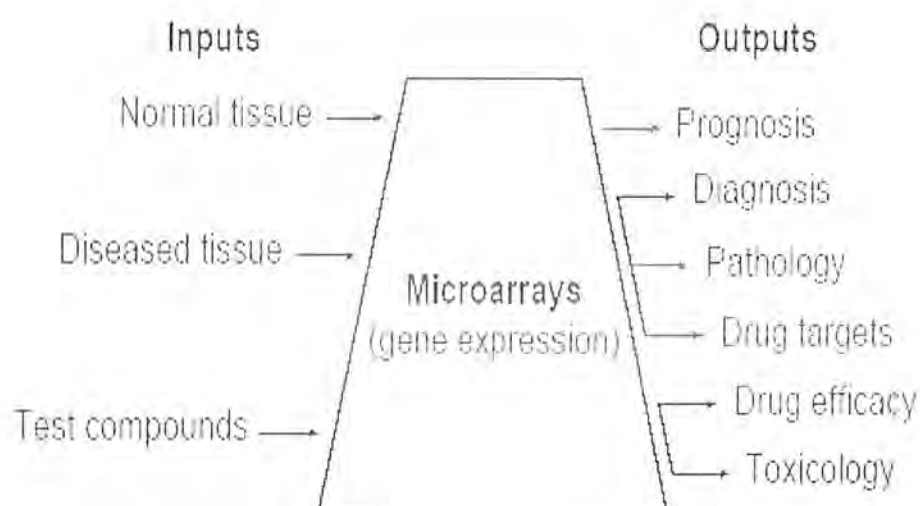


Figure 1.6 Applications of Microarray.

1.2.3 Examining the Gene Expression

Gene expression can be examined in two different ways: static and dynamic. In static microarray experiments, gene expression is an image at a single time. In time course experiments or dynamic microarray experiments, the expression profiles of genes are measured again and again over a time period. Biological processes are dynamic and complex hence dynamic microarray experiments are quite effective in exploring gene's functions, interaction of gene with their products as well as in the study of gene expression profile levels over a period of time. Hence time course microarray experiments or dynamic microarray experiments are a powerful tool for detecting the genes that are expressed periodically as well as to understand the temporal patterns of

gene expression meaning that it is possible to monitor temporal variations in gene expression (Rasooly and Herold, 2008; Tchagang et al., 2010; Liew et al., 2011).

1.2.4 Microarray Expression Matrix

The data generated from microarray experiment is usually in the form of large matrices consisting of rows representing genes and columns representing experimental conditions (Babu, 2004).

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix}$$

Figure I.7 A $m \times n$ expression matrix X with m genes across n conditions and expression value of gene c in condition j is denoted as x_{ij} .

Table 1.1 Complete Microarray Data Matrix.

Time	X40	X50	X60	X70	X80	X90	X100	X110	X120	X130	X140	X150	X160	X170	X180	X190	X200	X210	X220
YAL001C	-0.07	-0.25	-0.1	0.05	-0.04	-0.12	-0.25	-0.44	-0.05	0.12	0.06	-0.04	0.51	0.55	0.34	-0.28	-0.09	-0.44	0.35
YAL014C	0.215	0.09	0.025	-0.04	-0.04	-0.02	-0.51	-0.02	0	0.46	-0.15	0.04	0.18	-0.3	-0.38	0.07	-0.04	0.13	-0.06
YAL016W	0.15	0.15	0.22	0.29	-0.1	0.15	-0.79	0.15	-0.15	0.29	-0.08	0.16	-0.05	0.12	-0.17	0.11	-0.15	0.03	-0.26
YAL020C	-0.35	-0.25	-0.215	-0.15	0.16	-0.12	0.26	0	0.15	-0.2	-0.07	-0.05	0.43	0.07	0.61	-0.2	0.49	-0.43	0.8
YAL022C	-0.415	-0.59	-0.58	-0.57	-0.09	-0.34	0.45	0.32	1.15	0.2	0.31	-0.28	-0.01	-0.48	-0.4	-0.59	0.54	-0.09	1.03
YAL036C	0.54	0.33	0.215	0.1	-0.27	0.45	0.62	0.37	-0.6	0.27	-0.27	0.21	-0.4	-0.11	-0.55	0.24	-0.19	0.27	-0.5
YAL035W	-0.625	-0.6	-0.4	-0.2	-0.13	0.15	-0.55	0.26	-0.18	0.35	0.13	0.34	0.26	0.16	0.15	0.11	0.57	0.32	0.23
YAL039C	0.05	-0.24	-0.19	-0.14	-1.21	-0.16	-0.21	0.2	0.47	0.35	0.11	0.52	-0.48	-0.13	-0.26	-0.03	-0.2	0.11	-0.05
YAL040C	0.335	0.05	-0.04	-0.13	0.02	0.04	-0.14	0.34	0.91	0.42	0.58	-0.2	0.03	-0.55	-0.31	-0.56	0.17	-0.3	0.36
YAL044C	-0.43	-0.46	-0.39	-0.32	-0.66	0.03	-0.79	0.1	-0.89	0.05	-0.97	0.17	-0.31	0.18	-0.14	0.46	0.4	0.37	-0.24
YAL045C	0.135	0.23	0.125	0.02	0.09	0.16	-0.16	-0.02	-0.17	0.1	-0.14	-0.09	0.02	-0.27	0.16	-0.25	0.19	0.08	-0.11
YAL048C	0.005	0.02	-0.05	-0.12	0.1	0.1	0.11	0.01	-0.01	-0.39	-0.45	1.92	0.02	-0.13	-0.05	-0.35	-0.02	-0.25	0.53
YAL049C	-0.2	-0.32	-0.32	-0.32	-0.53	-0.26	0.09	-0.23	0.37	0.04	-0.15	0.82	-0.07	0.25	0.08	-0.18	-0.42	-0.08	0.02
YAL051W	0.155	0.2	0.25	0.26	-0.04	0.34	-0.31	-0.37	-0.25	0.27	-0.35	0.3	-1.15	0.19	-0.44	0.11	0.24	0.17	0.27
YAL056W	-0.135	-0.25	-0.205	-0.16	0.09	0.06	0.25	0.14	0.21	-0.19	-0.11	-0.23	-0.5	-0.19	0.25	-0.29	0.29	-0.45	0.53
YAL063C	0.42	0.43	0.325	0.22	0.03	0.2	-0.58	0.21	-0.34	0.06	0.11	-0.54	0.04	0	0	-0.04	0.28	-0.08	-0.1
YAL064W	0.1	0.07	-0.11	-0.29	0.56	-0.4	0.24	0.05	0.29	-0.07	0.33	0.99	0.28	-0.15	-0.06	-0.56	-0.02	-0.37	-0.03
YAL065C	-0.07	-0.12	0.075	0.27	0.22	0.15	0.19	0.18	0.13	0.19	-0.04	-0.04	0.05	-0.12	-0.02	-0.45	0.04	-0.08	0.22
YAR007C	1.26	1.55	0.71	-0.11	-0.59	-1.21	-1.43	-1.37	-1.26	0.72	0.94	0.15	0.65	0.99	0.15	-0.53	-0.66	-0.55	-1.06
YAR052C	0.25	0.32	-0.02	-0.36	0.45	-0.09	0.24	0	0.30	-0.02	0.23	-0.1	0.16	-0.27	0.14	-0.18	0.16	-0.61	0.31
YAR060C	0.55	0.65	-0.1	-0.86	0.44	-0.48	0.57	-0.4	-0.42	-0.22	0.45	-0.38	0.25	-0.42	0.14	-0.65	0.06	-0.92	0.38
YAR070C	0	0	-0.15	-0.3	0.11	-1.4	0.69	0.05	0.3	-0.58	0.32	-0.63	0.56	0.42	0.17	-0.1	0.38	0.34	0.57
YAR073W	0.26	0.21	0.155	0.1	0.47	-0.21	0.41	0.12	0.21	-0.46	-0.08	0.12	0.19	-0.37	0.09	-0.33	0.22	-0.1	0.75
YBL003C	0.215	1.26	1.375	1.43	1.05	0.15	-0.64	-1.35	-1.91	-1.12	-0.31	1.11	1.24	1.73	1.33	0.85	0.3	0.06	-0.44

Rows represent genes and columns represent experimental conditions. Each cell represents a gene expression value according to the experimental conditions.

1.3 Missing Data

One of the most important problems of microarray data is that it contains missing values.

Table 1.2 Data matrix containing missing values.

	X40	X50	X60	X70	X80	X90	X100	X110	X120	X130	X140	X150
YAL001C	NA	0.215	0.15	-0.35	NA	0.54	-0.625	0.05	0.335	-0.43	0.135	0.005
YAL014C	NA	NA	NA	-0.28	NA	0.33	-0.6	-0.24	0.09	NA	0.23	0.02
YAL016W	-0.1	0.025	NA	-0.215	-0.58	0.215	NA	NA	-0.04	NA	0.125	NA
YAL020C	NA	NA	0.29	NA	-0.57	NA	NA	-0.14	NA	NA	0.02	NA
YAL022C	-0.04	-0.04	-0.1	0.16	NA	-0.27	-0.13	-1.22	0.02	-0.66	NA	0.1
YAL036C	NA	NA	0.15	-0.12	-0.34	0.45	0.33	NA	NA	0.03	0.16	0.1
YAL038W	-0.28	-0.51	-0.73	NA	NA	NA	-0.53	NA	-0.14	-0.79	NA	0.11
YAL039C	-0.44	NA	NA	0	0.32	0.32	0.26	NA	NA	0.1	NA	0.01
YAL040C	-0.09	NA	-0.15	0.13	NA	-0.6	NA	-0.47	0.91	-0.89	-0.17	NA
YAL044C	NA	0.46	NA	-0.2	0.2	NA	0.35	NA	0.42	NA	0.1	-0.39
YAL046C	0.06	-0.19	-0.08	-0.07	0.31	-0.27	NA	NA	NA	NA	-0.14	-0.49
YAL048C	-0.04	NA	0.16	-0.05	-0.28	0.21	0.34	NA	-0.2	0.17	-0.09	NA
YAL049C	NA	0.18	NA	NA	-0.01	0.1	0.78	NA	NA	-0.31	0.02	0.02
YAL051W	0.59	-0.3	NA	NA	-0.48	NA	0.16	-0.13	-0.38	0.18	NA	NA
YAL056W	NA	-0.38	-0.17	NA	-0.4	-0.35	0.15	-0.26	NA	-0.14	0.16	-0.05
YAL063C	-0.28	NA	0.11	-0.2	NA	0.24	NA	NA	-0.56	NA	NA	-0.35
YAL064W	NA	-0.04	-0.15	NA	NA	-0.19	0.57	NA	0.17	NA	0.19	-0.01
YAL065C	-0.44	NA	NA	NA	NA	0.27	NA	NA	-0.3	0.87	0.08	NA
YAR007C	NA	-0.06	-0.26	0.8	NA	-0.5	NA	-0.05	0.36	-0.24	-0.11	NA

Rows represent genes and columns represent experimental conditions. Each cell represents a gene expression value according to the experimental conditions. Here NA represents the missing values.

1.3.1 Occurrence of Missing Values

There are various reasons for the occurrence of missing values including insufficient resolution, image noise and corruption, artifacts on microarray, dust or scratches on slide, hybridization failures and experimental errors during laboratory process. These technical limitations result in corrupted spots on microarray. When these corrupted spots are filtered during image analysis phase they produce missing data (Liew et al., 2011).

Most of the times 1 to 10% of microarray data is missing which effects up to 95% of genes (Hourani et al., 2009).

Several methods have been proposed to deal with missing values which either delete genes having missing values, replace missing values by zero or calculate missing values by average or median of corresponding rows or columns. These methods result in loss of useful information due to deletion of missing values. Change of variance occurs among the variables by the substitution of missing values with zeroes or row averages. Furthermore, correlation of data is not considered which leads to estimation errors (Ji et al., 2011; Mayer, 2013).

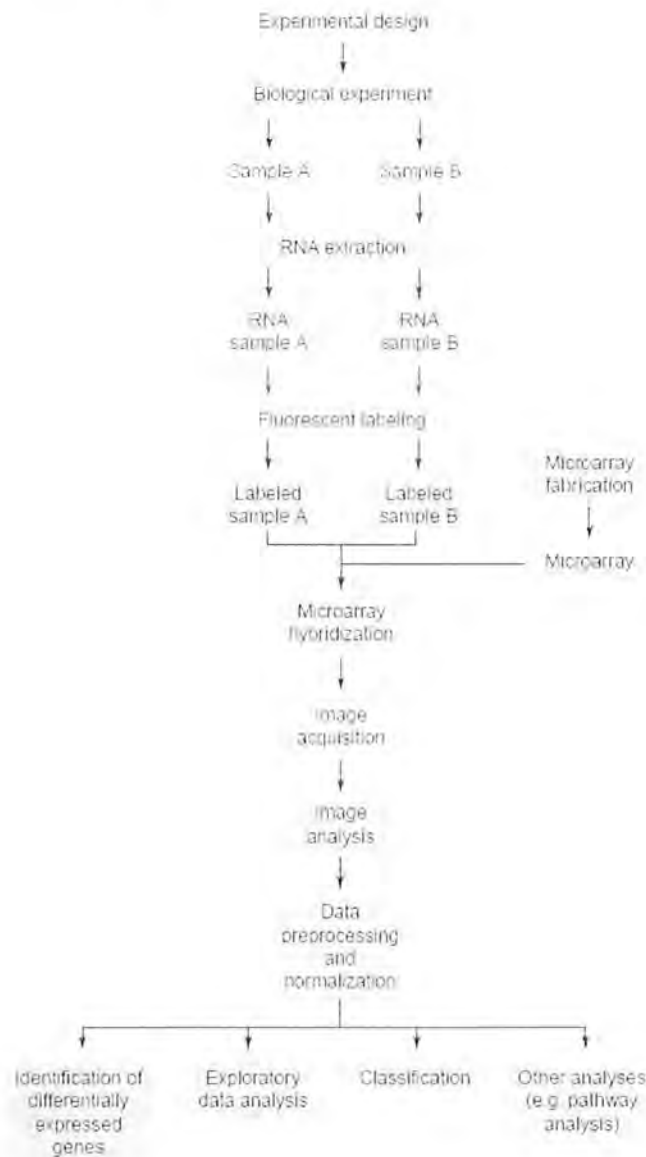


Figure 1.8 A flow chart of a typical microarray experiment.

1.4 Microarray Data Processing and Image Analysis

The data produced from microarray experiments is a raw data called “hybridized microarray images”. After the hybridization step, the spots produced are excited by laser. These spots are then scanned at suitable wavelengths in order to detect red and green dyes. Amount of bound nucleic acid is represented by the amount of fluorescence emitted at excitation. Red, green and yellow spots appear as a result of microarray experiments. The intensity of fluorescence level of these spots represents relative expression level of genes. If cDNA for a diseased gene is highly expressed, the spot would be red. If cDNA for normal gene is highly expressed, spot would be green. The spot would be yellow if gene is expressed to same extent in both conditions. Next step is “image quantitation” during which images are analyzed; intensity of each spot is measured and compared to background intensity. Microarray data processing and image analysis steps have been explained by Qin et al., (2005).

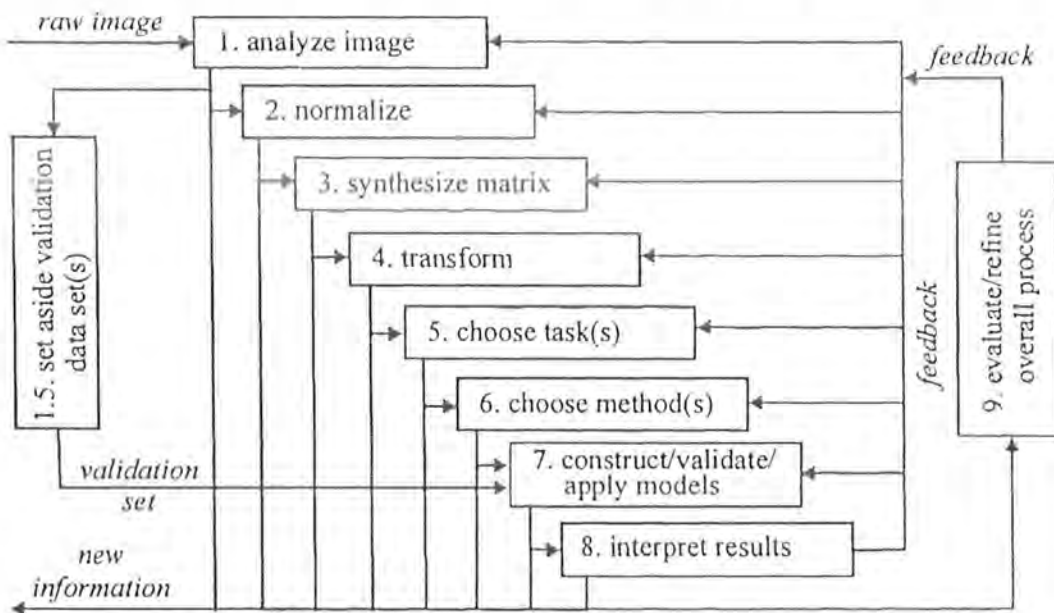


Figure 1.9 Steps of Microarray image analysis and data processing.

Adapted by Werner Dubitzky, 2003.

1.4.1 Addressing

The red and green fluorescence intensities are stored as a pair of 16 bit TIFF files of scanned images. They are typically from 2.5 to 20 MB in size. Different fluorescence dyes absorb and emit light at different wavelengths therefore scanners generate

excitation lights at different wavelengths and detect the different emission wavelengths. Image is analysed to identify spots after completion of image generation.

1.4.2 Segmentation

Segmentation of an image is defined as a process in which pixels of a microarray image are classified as foreground (within a spot) or background. This is done in order to calculate fluorescence intensities for each spotted DNA sequence as measures of abundance of transcript. A 'spot mask' is produced by this method comprising of a set of foreground pixels for each spot.

1.4.3 Information Extraction

Following is the process of information extraction from microarray image.

1.4.3.1 Spot Foreground Intensity

In a scanned image, each pixel represents the amount of hybridization that takes place. So for a particular spotted DNA sequence, total amount of hybridization is proportional to total fluorescence at the location of spot. Hence foreground intensity of spot is estimated as the average intensity of all foreground pixels.

1.4.3.2 Spot Background Intensity

Apart from level of hybridization of every target to probe, the measured intensity of spot also includes the amount of non specific hybridization and contaminations due to chemicals on glass slide, called background intensity. The standard method for the estimation of background intensity of spot is based on the assumption that the background level is same as the intensity in the close surroundings of spot. Most common method is to calculate sample mean or median of background pixels which is considered as background estimate or intensity of the spot. For this, an area near each spot is selected, background pixels in this area are identified and at last background intensity is calculated.

1.4.3.3 Background Correction

The next step of microarray image processing is background correction. As the measured fluorescence intensity of spot also includes the intensity due to

contamination on slides apart from hybridization of mRNA samples to spotted DNA, background correction is recommended to reduce bias. To accurately quantify the fluorescence intensity of spot, background intensity is subtracted from foreground intensity, by following equation:

$$I_t = I_f - I_b \quad (1)$$

Where I_t is intensity obtained after background correction, I_f is foreground intensity and I_b is background intensity.

1.4.4 Weak Spots or Missing Values

Even after background correction, a great number of low expression spots are expected to have negative values. As it is a common approach to define threshold and foreground intensity must exceed this threshold for the spot to be considered. Spot pixels are compared to background pixels. After comparison, if fraction of spot pixels is less than the given threshold and greater than median of background pixels, such spots are considered as weak spots. Hence the gene expression corresponding to this spot is set as missing. Furthermore if any spot has fluorescent intensity less than certain threshold, value of that spot is also defined as missing (Qin et al., 2005).

1.4.5 Expression Ratios (The Primary Comparison)

Microarray experiments are helpful in investigating relationships between related biological samples based on expression patterns and the simplest approach looks for differentially expressed genes. Ratios of red and green fluorescence are calculated to get a measure of expression changes. But these ratios have a drawback that they treat up and down regulated genes differently producing different expression ratios for gene upregulated and downregulated by same factor. Hence logarithm base 2 is widely used as alternative transformation of expression ratios. It produces continuous spectrum of values and treats upregulated and downregulated genes in a similar way (Babu, 2004).

1.4.6 Normalization

Normalization is the transformation applied to expression data. For meaningful biological comparisons to be made, normalization adjusts individual hybridization intensities so that they can be appropriately balanced. Normalization is carried out to

remove the bias and imbalance between the red and green dyes. The imbalances may arise due to various reasons like differences in print quality or position of the dye. It is applied to the log ratios of expression. Hence is important to eliminate variations among the data (Mehta, 2011).

After passing through the normalization procedure, the processed data called “gene expression matrix” is represented in the form of matrix. This data may contain missing values which are estimated by different imputation techniques (Qin et al., 2005).

1.5 Nature of Missing Data

It is required to empirically examine the pattern of missing data in the data set. Little and Rubin, (2014) has classified missing data into three main categories. These include:

- Missing completely at random (MCAR)
- Missing at random (MAR)
- Missing not at random (MNAR) or not missing at random (NMAR)

1.5.1 Missing completely at random

The data is said to be missing completely at random when probability of being missing is same for all cases. It means that causes of missing data are not related to the data. Hence in MCAR the probability of observation being missing, for a dependent variable, is not dependent upon observed and unobserved measurements (Durrant, 2009; Piyushimita, 2010; Van Buuren, 2012; Little and Rubin, 2014).

1.5.2 Missing at random

In this case missing values are not randomly distributed across all the observations rather they are randomly distributed within one or more groups or subsamples (Durrant, 2009; Piyushimita, 2010; Van Buuren, 2012; Little and Rubin, 2014).

1.5.3 Missing not at random

This condition exists when neither MCAR nor MAR hold. It means that probability of values of being missing varies for the reasons that are unknown (Durrant, 2009; Piyushimita, 2010; Van Buuren, 2012; Little and Rubin, 2014).

1.6 Statistical Problems due to Missing Data

Statistical analysis and interpretation of results of microarray data is difficult because of large and complicated datasets. The main purpose of statistical analysis of missing data is to estimate the significance of differential expression of genes, which is not possible due to the presence of missing data. The presence of missing data leads to the reduction of sample size that is available for analysis. Furthermore, statistical results based on missing data could be biased leading to erroneous results. Hence it is necessary to treat missing data appropriately in order to get accurate results.

1.7 Aim of Study

The main aim of this study was to find a trustworthy method for better imputation of missing data. The developed method uses Bayesian variable selection to extract the important genes and then use regression to estimate missing values. This method is compared to five different imputation methods to test their ability to predict missing data.

Chapter 2
Materials and Methods

MATERIALS AND METHODS

2.1 Data Collection and Software

The dataset used has been taken from Spellman et al., (1998). This is a time series dataset which contains samples taken from yeast cultures being synchronized by three separate approaches i.e., alpha factor arrest, elutriation and arrest of *cdc-15* temperature sensitive mutant. The dataset is available on ¹ (last accessed: 15 June, 2014) downloaded from the website named MINE; Maximal Information-based Nonparametric Exploration. It is in excel format containing 4381 genes in rows while 19 columns represent the expression of the genes at different times.

R version 3.0.2 (2013-09-25) (Statistical Package, 2009) was used for carrying out the simulations to impute missing data.

Microarray data contains missing values due to diverse reasons. Missing values affect the subsequent statistical analysis leading to erroneous results. Hence it is important to accurately estimate missing values.

One solution to avoid missing data is to repeat microarray experiments but this option is not feasible as it is costly and time consuming (Zhang et al., 2008). Several different approaches have been proposed to deal with missing values. Some simple methods are the following:

- One way is to delete expression vectors (genes) or eliminate data objects which contain missing values (Ghoneim et al., 2011).
- Another method is to replace all the missing values, present in the data, by zero (Friedland et al., 2006b).
- The corresponding row or column average or median is used to impute missing values (Friedland et al., 2006b).

¹ <http://www.exploredata.net/Downloads/Gene-Expression-Data-Set>

2.2 Drawbacks of Above Methods

Above described methods to impute missing values have some obvious drawbacks leading to erroneous results.

- Deletion of data objects or expression vectors having missing values cause loss of useful information.
- Deleting the variables having missing values result in the reduction of sample size that is available for analysis.
- Correlation of data is not considered by these methods (Troyanskaya et al., 2001).

2.3 Other statistical Methods to deal with Missing Values

Several different methods have been proposed to deal with missing values. Main approaches used are:

2.3.1 K Nearest Neighbors (KNN) Imputation

KNN impute is the most standard method used to impute missing values introduced by Troyanskaya et al., (2001). KNN is a local method and it makes use of the similarity structure of data for the imputation of missing values (Friedland et al., 2006a).

2.3.1.1 Steps of KNN

This method is divided into two steps:

2.3.1.1.1 *Selection of Similar Genes*

In first step, a set of genes having expression profiles similar to the genes having missing values are selected. Suppose there is a gene g having missing value in an experiment n . Let's say that $V_{g,n}$ is the missing value. KNN would find k other genes have expression profile similar to g in experiments (2-N). These k genes will have known values for experiment n (Troyanskaya et al., 2001; Ghoneim et al., 2011).

2.3.1.1.2 Predicting Missing Values

In second step missing values are predicted using observed values of selected genes. A weighted average using known values of k genes in experiment n are calculated and used as an estimate of missing values in gene g (Ghoneim et al., 2011).

Following equation is used to calculate the contributions of each gene to gene g on the basis of similar expression profiles (Hourani et al., 2009):

$$W = \frac{1/D_i}{\sum_{i=1}^k 1/D_i} \quad (2)$$

Where k is the number of genes selected on the basis of similar expression profiles. D_i is the distance between the i -th gene and the gene to be imputed.

2.3.1.2 Metrics Used for Gene Similarity

Commonly used metrics for gene similarity are Euclidean distance and Pearson correlation coefficient. Different researches show that Euclidean distance is more appropriate even though it is sensitive to outliers, as log transforming the data sufficiently reduces the effect of outliers on determination of gene similarity (Hourani and El, 2009; Sahu et al., 2011).

2.3.1.3 Computational Complexity of KNN

KNN impute method has computational complexity of approximately $O(m^2n)$ based upon the assumption that missing values are less than 20% (Sahu et al., 2011).

2.3.2 Singular Value Decomposition Imputation

SVD is a common technique for the analysis of multivariate data especially gene expression data. It is a global method. It is a method in which a set of mutually orthogonal expression patterns is obtained. These expression patterns can then be combined linearly so that expression of all genes in the data set can be approximated. These patterns are identical to principal components of gene expression matrix and

are also referred to as eigengenes. Hence, SVD involves linear transformation of expression data which is in the form of genes X arrays space which is reduced to eigengenes X eigenarrays space. Data is diagonalized in this space and each eigengenes is expressed only in its corresponding eigenarrays. At the end missing values are estimated by regressing genes against k most eigengenes (Alter et al., 2000; Troyanskaya et al., 2001).

Following equation is used to represent SVD:

$$A_{m \times n} = U_{m \times m} \Sigma_{m \times n} V_{n \times n}^T \quad (3)$$

Matrix V^T contains eigengenes. The contribution of these eigengenes to the expression is measured by Σ , which contains corresponding eigenvalues on the diagonal of the matrix. After that eigengenes are sorted by their eigenvalues to identify k most eigengenes. Suppose that the missing value is represented by j and gene is represented by i . After the selection of k most eigengenes from V^T matrix, to estimate missing values gene i containing missing values is regressed against k eigengenes. Then coefficients of regression are used to reconstruct j using linear combination of k eigengenes. As SVD can only be performed on complete matrices, A' is obtained by substituting row average for all the missing values present in matrix A . Then an expectation minimization method is used to obtain final estimate. In this method, above algorithm is used to estimate each missing value in A' . This procedure is repeated on newly obtained matrix. Each missing value of A' is estimated iteratively until RMSE between two consecutive A' falls below the threshold of 0.01 (Troyanskaya et al., 2001).

2.3.2.1 Computational Complexity of SVD

The computational complexity of SVD impute method is $O(n2mi)$ where i stands for the number of iterations performed before the threshold value is reached (Sahu et al., 2011).

2.3.3 Bayesian Principal Component Analysis Imputation

BPCA is a global method and it uses probabilistic Bayesian theory for missing value imputation. The whole gene expression data set is represented by a matrix Y . BPCA divides this data set into two data sets:

- Complete Data Set: having no missing value, Y^{obs}
- Incomplete data set: having missing values, Y^{miss}

2.3.3.1 Steps of BPCA

- There are three steps of BPCA:
 1. Principal component regression
 2. Bayesian Estimation
 3. Expectation minimization

2.3.3.1.1 *Principal component regression*

In PC regression the missing part Y^{miss} is estimated from observed part Y^{obs} . First, data matrix Y is normalized. Now transform independent variables X of data matrix Y to their principal components. Bayes theorem is used to calculate probabilistic PCA. So in this step a low rank approximation of data set is performed.

2.3.3.1.2 *Bayesian Estimation*

Bayesian estimation calculates posterior distribution of model parameter θ and input matrix X using:

$$p(\theta, X|Y) \propto p(X, Y|\theta)p(\theta) \quad (4)$$

- Where $p(\theta)$ is called as prior distribution.

Bayesian estimation is carried out on the assumption that the residual error and projection of each gene on the principal components behave as normal independent

random variables with parameters not known. Bayesian estimation algorithm is executed for both θ which is a model parameter and Y^{miss} . Then distributions for θ and Y^{miss} , $q(\theta)$ and $q(Y^{miss})$ are calculated using following equation:

$$q(Y^{miss}) = p(Y^{miss} | Y^{obs}, \theta_{true}) \quad (5)$$

Here θ_{true} is posterior of missing value. Last step is the imputation of missing values using following equation:

$$Y^{miss} = \int Y^{miss} q(Y^{miss}) dY^{miss} \quad (6)$$

2.3.3.1.3 *Expectation Minimization*

This is last step of BPCA. In this Bayesian estimation is followed by the iterations based on the Expectation Minimization of the unknown Bayesian parameters (Hourani and El, 2009; Liew et al., 2011; Sahu et al., 2011).

2.3.4 **Least Squares Imputation**

This method has been introduced by Bø et al., (2004). It uses multiple regression model based upon least squares principal. It uses correlation between genes and between arrays to estimate missing values. Two basic LSimpute methods are:

LSimpute_gene: This method uses correlation between genes.

LSimpute_array: This method uses correlation between arrays.

The gene expression matrix is represented by Y . For missing value estimation $V_{g,n}$ where g represents gene and n represents number of sample or experiments, first of all k most correlated genes are selected which have similar expression profile to target gene and does not contain missing values. At the end LS regression method is used for the estimation of missing values $V_{g,n}$. For the data having strong correlation LS impute performs better as it has the flexibility to adjust number of predictor genes k during regression (Bø et al., 2004; Gan et al., 2006; Hourani et al., 2009).

2.3.5 Local Least Squares Imputation

This method has been proposed by Kim et al., (2005). This algorithm uses linear correlation of the target gene with its k nearest neighbours to select the most correlated genes. Least squares formulation of k most related gene and the non missing values i.e., w_i of g_i are used to estimate missing values (Vg,i). w_i is a vector of non entries that are not missing for gene g_i . LLSimpute uses the KNN process to find out k nearest neighbour gene using Euclidean distance or Pearson correlation. These genes are said to be coherent to target genes. Row averages of respective rows are used to fill missing values in these coherent genes. Then two matrices A and B and a vector w are formed based upon KNN genes. For the estimation of missing values using k most correlated genes, every element of matrix A and B and a vector w is constructed as:

$$\begin{pmatrix} g_{s1} \\ g_{s2} \\ \dots \\ g_{sk} \end{pmatrix} = \begin{pmatrix} \alpha_1 & w_1 & w_2 & \dots & w_j & \alpha_2 \\ B_{11} & A_{11} & A_{12} & \dots & A_{1j} & B_{12} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ B_{k1} & A_{k1} & A_{k2} & \dots & A_{kj} & B_{k2} \end{pmatrix}$$

Figure 2.1 Matrix of k most similar genes for the estimation of missing values.

Here i represents the number of experiments, α_1 and α_2 represent missing values and k most similar genes to g are g_{s1}, \dots, g_{sk} . A linear coefficient vector x is established such that the square is minimized as:

$$\min A^T X - w|^2 \tag{7}$$

Here x denotes a vector where the square is minimized having x_i as the coefficients of linear combination. Hence the missing values present in g can be calculated as follows:

$$\alpha_1 = B_{11} + B_{21}x_2 + \dots + B_{k1}x_k \tag{8}$$

$$\alpha_2 = B_{12} + B_{22}x_2 + \dots + B_{k2}x_k \tag{9}$$

In the above equations a_1 and a_2 are actually the first and second missing values present in the target gene. So for missing value estimation of each gene, first matrices A and B and vector w are built up and then apply algorithm for all missing values.

2.3.6 missForest

This package has been introduced by Stekhoven and Bühlmann, (2012), developed in R language that can impute missing values for any type of input data i.e., mixed-type of variables, categorical data, continuous data, non linear relations, complex interactions and high dimensionality ($p \gg n$). It is basically a non parametric missing value imputation method. This algorithm uses random forest (RF) method. RF is trained on the observed part of data matrix. For each variable missForest fits the random forest on observed part to predict missing values. missForest algorithm continues to run iteratively until some stopping criterion is met. missForest continuously updates the imputed matrix variable wise and then assess the performance between iterations by considering the difference between previous and new imputation result.

2.3.6.1 Steps

- Consider $X=(X_1, X_2, X_3, \dots, X_p)$ to be an $n \times p$ dimensional data matrix.
- At start use mean imputation to estimate missing values of X .
- Sort variables of X according to the amount of missing values starting with the lowest value.
- Now use random forest to impute missing values.
- Stopping criterion is met as soon as difference between newly imputed data matrix and previous one increase for first time.
- Difference for set of continuous variables N is defined as:

$$\Delta N = \frac{\sum_{j \in N} (x_{new}^{imp} - x_{old}^{imp})^2}{\sum_{j \in N} (x_{new}^{imp})^2} \quad (10)$$

- And for set of categorical variables F as:

$$\Delta F = \frac{\sum_{j \in F} \sum_i^n |x_{new}^{(mp)} - x_{old}^{(mp)}|}{\#NA} \quad (11)$$

Here NA is number of missing values in categorical variable.

- After the imputation of missing values, performance of algorithm is assessed using normalized root mean square error.

An OOB (out-of-bag) estimation error is produced for the observed part of variable by fitting RF on that observed part.

2.4 Proposed Algorithm and its Working

The proposed approach uses Gibbs Bayesian variable selection algorithm to select the important predictors and then impute missing values using linear regression.

2.4.1 Gibbs Bayesian Variable Selection Algorithm

Selection of predictors is a crucial problem while building a regression model in statistics. A wide variety of methods have been proposed to find out the potential predictors like AIC, BIC and C_p and some step wise procedures like forward regression and backward regression. These methods include or exclude the variables sequentially based upon their R^2 considerations (George and McCulloch, 1993).

BVS identifies the important predictor variables from a gene expression microarray data set on the base of their posterior probabilities. George and McCulloch, (1993) proposed a Bayesian variable selection method i.e., Stochastic Search Variable Selection (SSVS) to identify the predictor variables on the basis of higher posterior probability. In the microarray gene expression data sets these predictor variables are to be chosen from the genes i.e., those genes which play an important role in cell regulation.

2.4.1.1 Bayesian Variable Selection Algorithm

Suppose there is $n \times p$ covariate matrix $X=(X_1 \dots X_p)$, also known as set of potential predictors and an n dimensional vector Y of dependent variables being linked by a normal linear model:

$$Y|\beta, \sigma^2 \sim N_n(X\beta, \sigma^2 I) \quad (12)$$

In the above equation $\beta = (\beta_1, \dots, \beta_p)$ is an unknown p vector and σ^2 is a scalar which is also unknown. This hierarchical model has a key feature that each component of β is modelled as obtained from a mixture of two normal distributions having different variances. A latent variable γ_i is introduced having value of either 0 or 1 and it determines about the inclusion or exclusion of β_i in the model, hence according to George and McCulloch, (1993) the normal mixture is represented by

$$\beta_i|\gamma_i \sim (1 - \gamma_i)N(0, \tau_i^2) + \gamma_i N(0, c_i^2 \tau_i^2) \quad (13)$$

And

$$P(\gamma_i = 1) = 1 - P(\gamma_i = 0) = p_i \quad (14)$$

c_i and m_i are the prior parameters. When $\gamma_i=0$ then $\beta_i \sim N(0, \tau_i^2)$ indicating the absence of covariates hence prior distribution of β_i is close to zero. When $\gamma_i=1$, $\beta_i \sim N(0, c_i^2 \tau_i^2)$ assuming that little prior information is available about β_i and hence the covariates are included in the model. At the end inverse gamma conjugate prior is used to calculate prior on residual variance σ^2 (George and McCulloch, 1993, 1997):

$$\sigma^2|\gamma \sim I(v_\gamma / 2, v_\gamma \lambda_\gamma / 2)G \quad (15)$$

2.4.1.1.1 Gibbs Sampler

It is an MCMC method to estimate the desired posterior distributions. It is an adaptation of Metropolis algorithm (George and McCulloch, 1993).

SVSS uses Gibbs sampler to avoid the computational complexities while calculating all the 2^p posterior probabilities $inf(\gamma|Y)$. It generates a sequence

$$Y^1, \dots, Y^m \tag{16}$$

hence converging rapidly in distribution to $\gamma \sim f(\gamma|Y)$. Such sequence is obtained efficiently. There are possibilities of this sequence to contain valuable information of variable selection due to high probability. The reason is that Y with high probability occur more frequently and hence can be easily identified and are of actual interest. While the Y which occur less often do not occur at all are not important. An auxiliary Gibbs sequence is generated using Gibbs sampler in which sequence in equation (16) is embedded.

$$\beta^0, \sigma^0, \gamma^0, \beta^1, \sigma^1, \gamma^1, \dots, \beta^j, \sigma^j, \gamma^j, \dots \tag{17}$$

B^0, σ^0 are initialized to be the least square estimates of equation (12). Y^0 is initialized as $\gamma^0 \equiv (\mathbf{1}, \mathbf{1}, \dots, \mathbf{1})$. Inverse gamma distribution is used for sampling and obtaining values of coefficient vector β^j and variance σ^j . In the final step Y^j is obtained by sampling using the conditional distribution. Repeated successive sampling results in prediction of Gibbs sequence. As a result length of subsequence increases and empirical distribution of the realized Y values will join the actual posterior $f(\gamma|Y)$. Now the sequence (17) contains relevant information to be used for variable selection.

When the sequence has reached an approximate stationarity then those values of Y which corresponds to the most promising subset having high frequency will appear. Their occurrence is due to largest probability under $f(Y|Y)$ (George and McCulloch, 1993; Li and Zhang, 2010).

2.4.2 Linear Regression to Estimate missing Values

For linear regression suppose that Y is a dependent variable and X_1, \dots, X_p is a set of covariates or potential predictors. Now based on the resulting model obtained from Gibbs Bayesian Variable selection, a regression model is fitted and used to impute missing values. For variable Y_j with missing values

$$Y = X_1^* \beta_1^* + \dots + X_q^* \beta_q^* + \epsilon \tag{18}$$

In the above equation $X_1^* \dots X_q^*$ are the selected subsets of $X_1 \dots X_p$. $\beta_1^* \dots \beta_q^*$ are the new parameters that have been drawn from posterior predictive distribution of missing data. At the end missing values are calculated by:

$$Y = \beta_0^* + \beta_1^*X_1 + \beta_2^*X_2 + \dots + \beta_q^*X_q \quad (19)$$

2.4.3 Input

The code takes two datasets as an input, the complete dataset and the incomplete dataset i.e., one having missing values in Excel format. 5%, 10%, 20% and 40% missing values have been produced artificially in the dataset using the `prodNA` function of `missForest` package. This function basically deletes the entries completely at random according to the specified amount in the complete data set.

Table 2.1 Complete dataset.

Time	YAL001C	YAL014C	YAL016W	YAL020C	YAL022C	YAL036C	YAL038W	YAL039C	YAL040C	YAL044C	YAL046C	YAL048C	YAL049C	YAL051W
X40	-0.07	0.215	0.15	-0.35	-0.415	0.54	-0.625	0.05	0.335	-0.43	0.135	0.005	-0.2	0.155
X50	-0.23	0.09	0.15	-0.28	-0.59	0.33	-0.6	-0.24	0.05	-0.46	0.23	0.02	-0.32	0.2
X60	-0.1	0.025	0.22	-0.215	-0.58	0.215	-0.4	-0.19	-0.04	-0.39	0.125	-0.05	-0.32	0.23
X70	0.03	-0.04	0.29	-0.15	-0.57	0.1	-0.2	-0.14	-0.13	-0.32	0.02	-0.12	-0.32	0.26
X80	-0.04	-0.04	-0.1	0.16	-0.09	-0.27	-0.13	-1.22	0.02	-0.66	0.09	0.1	-0.33	-0.04
X90	-0.12	-0.02	0.15	-0.12	-0.34	0.45	0.33	-0.16	0.04	0.03	0.16	0.1	-0.26	0.34
X100	-0.28	-0.51	-0.73	0.25	0.49	0.62	-0.53	-0.21	-0.14	-0.79	-0.16	0.11	0.09	-0.31
X110	-0.44	-0.08	0.19	0	0.32	0.32	0.26	0.2	0.24	0.1	-0.02	0.01	-0.23	-0.27
X120	-0.09	0	-0.15	0.13	1.15	-0.6	-0.18	-0.47	0.91	-0.89	-0.17	-0.01	0.37	-0.29
X130	0.12	0.46	0.29	-0.2	0.2	0.27	0.35	0.35	0.42	0.05	0.1	-0.39	0.04	0.27
X140	0.06	-0.19	-0.08	-0.07	0.31	-0.27	0.18	0.11	0.58	-0.97	-0.14	-0.49	0.18	-0.35
X150	-0.04	0.04	0.16	-0.05	-0.28	0.21	0.34	0.52	-0.2	0.17	-0.09	1.92	0.82	0.3
X160	0.31	0.18	-0.05	0.43	-0.01	-0.4	0.28	-0.48	0.03	-0.31	0.02	0.02	-0.07	-1.19
X170	0.59	-0.3	0.12	0.07	-0.48	-0.11	0.16	-0.13	-0.38	0.18	-0.27	-0.13	0.25	0.19
X180	0.34	-0.38	-0.17	0.61	-0.4	-0.55	0.15	-0.26	-0.31	-0.14	0.16	-0.05	0.08	-0.44
X190	-0.28	0.07	0.11	-0.2	-0.59	0.24	0.11	-0.03	-0.56	0.46	-0.25	-0.35	-0.18	0.11
X200	-0.09	-0.04	-0.15	0.49	0.54	-0.19	0.57	-0.2	0.17	0.4	0.19	-0.01	-0.42	0.14
X210	-0.44	0.13	0.03	-0.43	-0.09	0.27	0.32	0.11	-0.3	0.87	0.08	-0.25	-0.08	0.17
X220	0.31	-0.06	-0.26	0.8	1.03	-0.5	0.23	-0.05	0.36	-0.24	-0.11	0.53	0.02	0.27

Rows represent experimental conditions and columns represent genes.

2.4.4 Imputing Missing Values by Mean Imputation

Then `impute.mean` function of the **Hot Deck Imputation** package of R is used to impute missing values. This method basically estimate the column mean of the complete cases for missing values.

2.4.5 Gibbs Bayesian Variable Selection and Linear regression

For the selection of predictors GibbsBvs method of Bayesian Variable Selection package of R is used and a model is developed based on those predictors.

2.4.5.1 Parameters of Gibbs Bayesian Variable Selection

Prior distribution for regression parameters as well as prior distribution over model space should be set initially. Possible choices for prior distribution for regression parameters are Robust, Liangetal, gZellner and ZellnerSiow while those for prior distribution over model space are Constant and ScottBerger. Possible choices for model at which simulation process starts are:

- **Null:** the model only with intercept
- **Full:** the model being defined by the formula
- A vector p (i.e., number of covariates present in the model) zeroes and ones defining the model.

A method has been developed based upon linear regression to calculate the missing values.

2.4.6 Normalized root Mean Square Error

To calculate NRMSE for the given complete data matrix, imputed data matrix and data matrix with missing values, **NRMSE** is used. It is the performance measure and implemented in missForest R package.

For full understanding of the proposed algorithm, it has been tested on the data having 5, 10, 20 and 40 percent missing values and is explained in Results and Discussion section.

Chapter 3

Results

RESULTS

The main aim of this study was to develop a method that can estimate the missing values more precisely based upon the identification of the most important predictors. The other methods for missing data identification like KNN, SVD, LSImpute, LLSImpute, missForest and BPCA do not identify the predictors. Hence a method was developed that involves the identification of predictors and then estimation of missing values using regression. The methods are then compared on the basis of their NRMSE values. If the NRMSE values lie close to 0 the estimation is considered to be good and reliable but if NRMSE values lies near 1 then the results of estimation are poor and not much reliable. Gibbs Bayesian Variable Selection along with linear regression shows better results as compared to the other methods.

3.1 Results of Proposed Algorithm

The NRMSE values decrease along with the increase in missing percentage of the values in dataset. This shows that using Gibbs Bayesian variable selection to select the important predictors which have significant effect and then predicting the missing values using those predictors in linear regression produces better results. Table 3.1 shows the NRMSE values for 5%, 10%, 20% and 40% missing percentages. For 5 percent missing data NRMSE values lies between 0.02 to 0.1 only once showing drastic increase in error rate when NRMSE is 0.109122. For 10 percent missing data NRMSE lies between 0.008 to 0.05, NRMSE values lie between 0.03 to 0.2 for 20 % missing values having a drastic increase in error when NRMSE is 0.242819, while for 40% missing percentage NRMSE values lie between 0.03 to 0.07 showing no drastic increase in NRMSE value. Figure 3.1 to Figure 3.4 shows the behaviour of NRMSE for 5%, 10%, 20% and 40% missing percentages respectively. Figure 3.5 shows that the performance of proposed algorithm gets better along with the increase in missing percentage. For 5 %, 10 % and 20 % missing data there is a significant increase and decrease of NRMSE but for 40% missing data there is no drastic change in NRMSE.

Table 3.1 NRMSE values of proposed algorithm for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.058336	0.041237	0.242819	0.039842
2	0.076179	0.008854	0.048969	0.078389
3	0.044825	0.051376	0.04246	0.078389
4	0.109122	0.046337	0.024596	0.07264
5	0.027699	0.036999	0.039855	0.049404

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

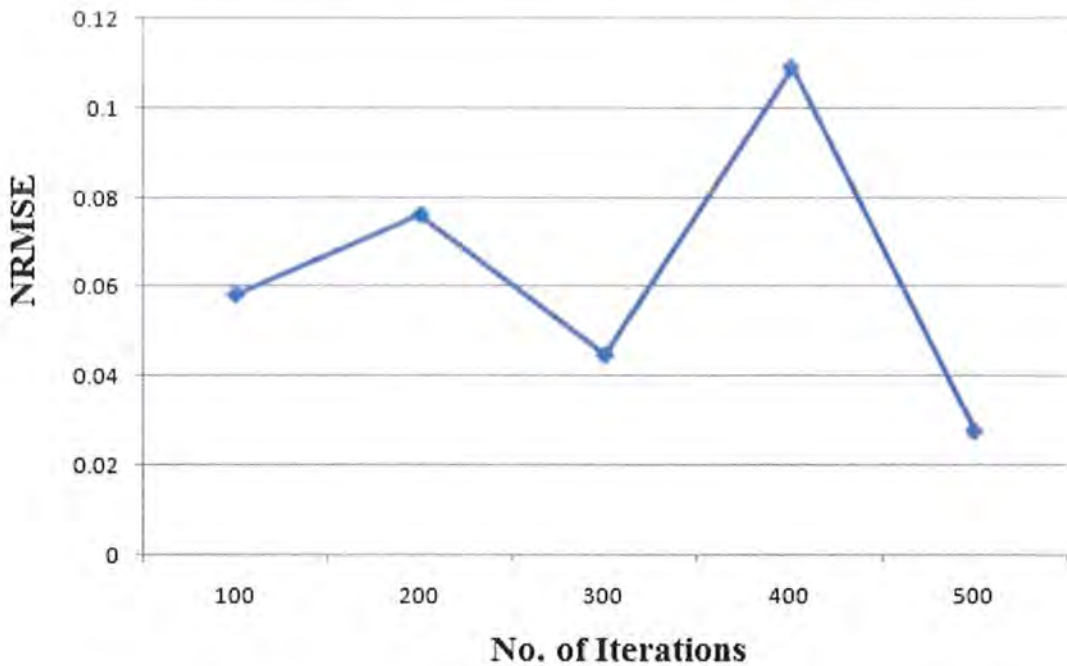


Figure 3.1 Results of proposed algorithm at 5% missing percentage.

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% missing percentage.

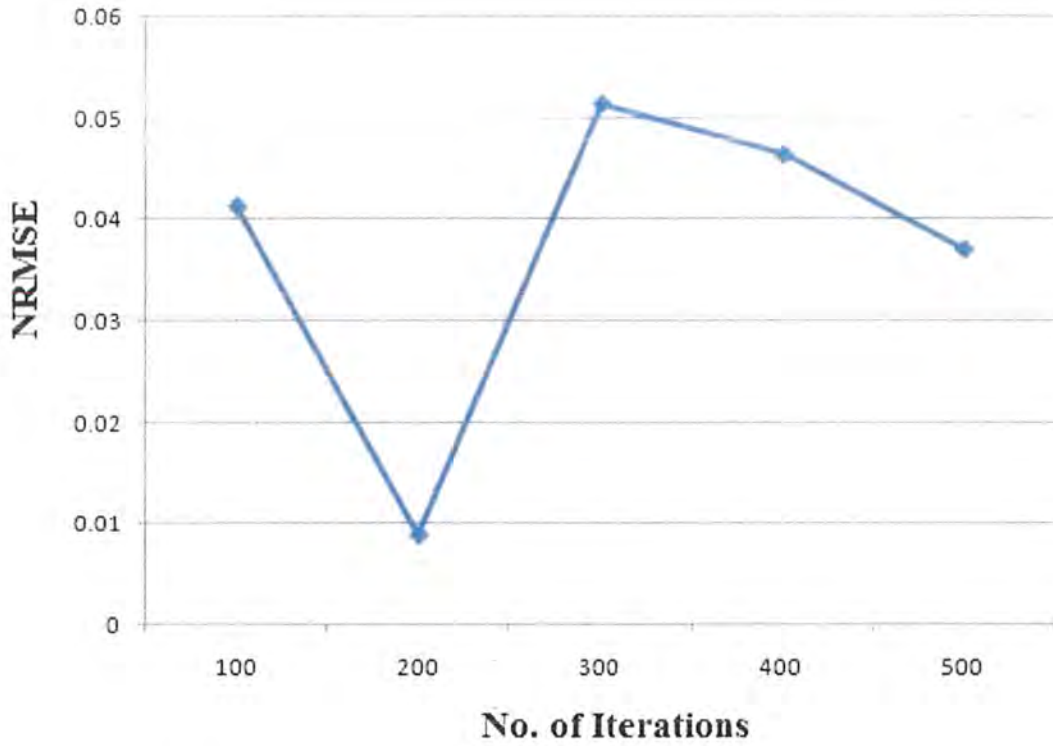


Figure 3.2 Results of proposed algorithm at 10% missing percentage.

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 10% missing percentage.

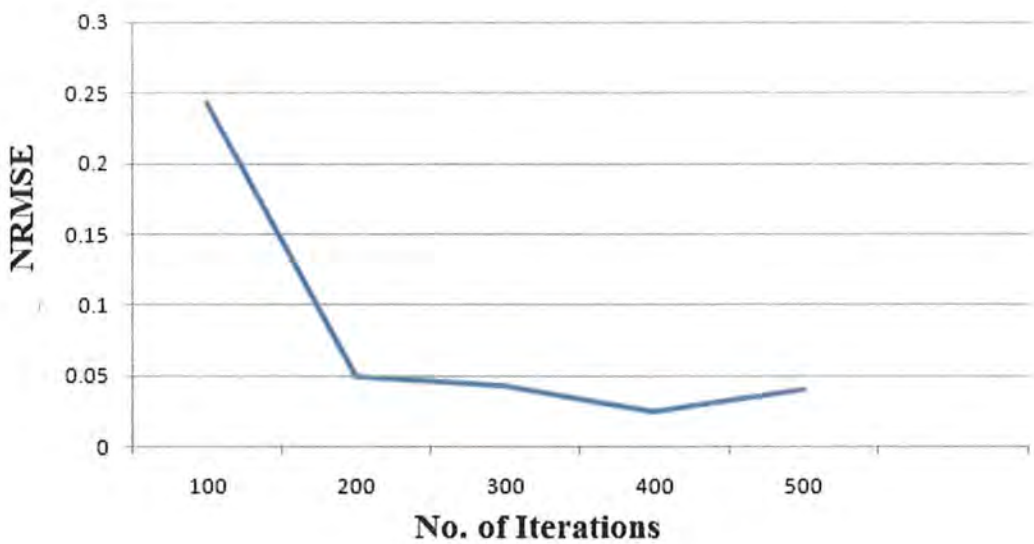


Figure 3.3 Results of proposed algorithm at 20% missing percentage.

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 20% missing percentage.

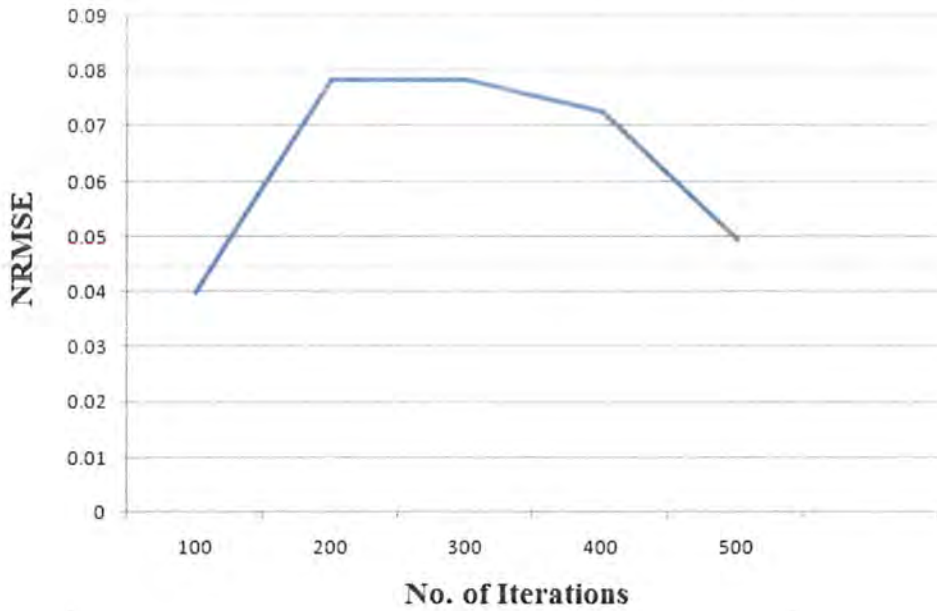


Figure 3.4 Results of proposed algorithm at 40% missing percentage.

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 40% missing percentage.

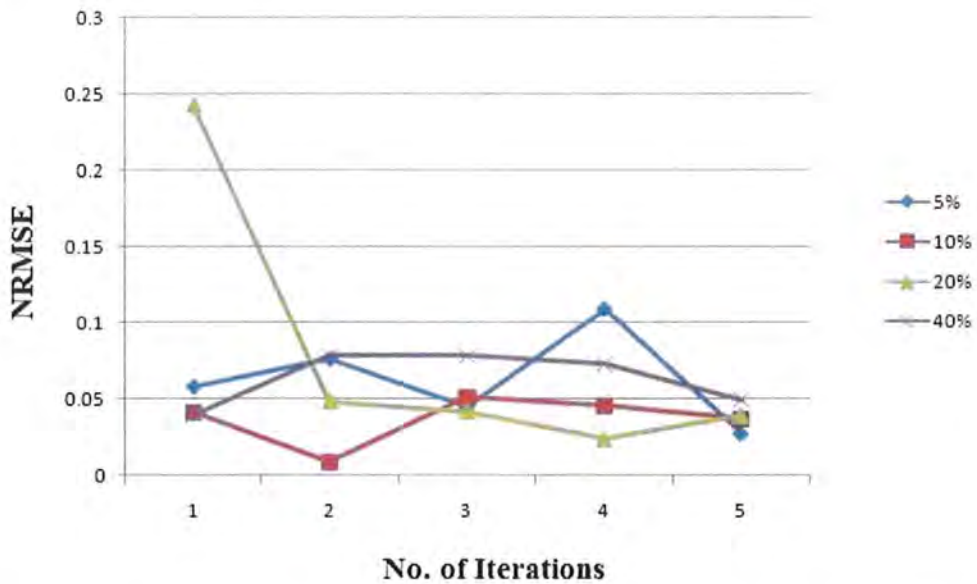


Figure 3.5 Results of proposed algorithm for given missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

3.2 Results of KNN

Drastic changes have been observed for all the percentages of missing values. Figure 3.6 shows the poor estimation of missing values along with the increase in missing percentage of values in data. The NRMSE values of the data lies close to 0 at 5% missing percentage (shown in Table 3.2) but it increases along with the increase in missing percentage. This shows that KNN calculates values more precisely with low missing percentage.

Table 3.2 NRMSE values of KNN for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.104321	0.104321	0.104321	0.104321
2	0.104321	0.133526	0.203857	0.290718
3	0.100041	0.100041	0.100041	0.100041
4	0.142654	0.142654	0.142654	0.142654
5	0.114296	0.114296	0.114296	0.114296

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

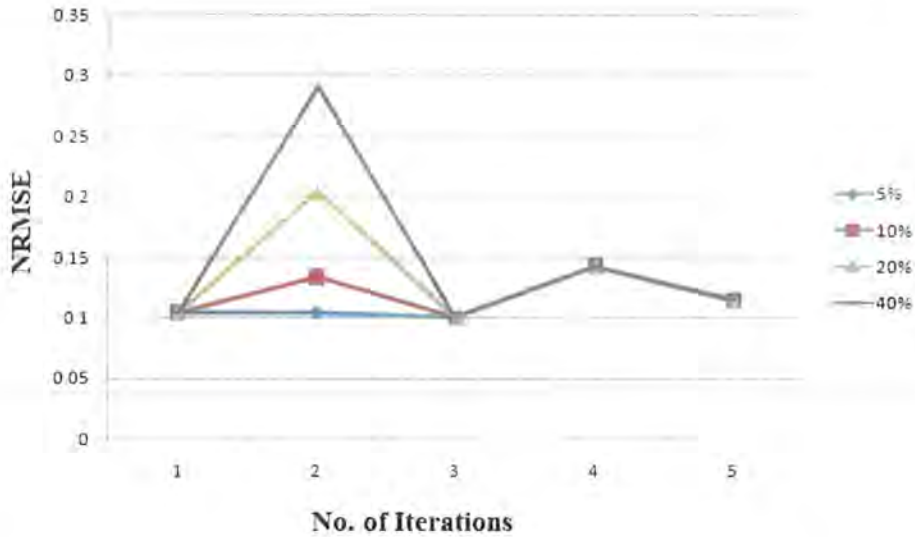


Figure 3.6 Results of KNN for given missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

3.3 Results of SVD

The NRMSE for the SVD estimation increases gradually along with the increase in missing percentage as shown in Figure 3.7. Table 3.3 shows values of NRMSE.

Table 3.3 NRMSE values of SVD for first 500 iterations for given missing percentages (5%, 10%, 20% and 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.087504	0.121827	0.180946	0.260561
2	0.069007	0.094092	0.170804	0.261514
3	0.078219	0.113484	0.171377	0.278108
4	0.099708	0.138852	0.174941	0.274315
5	0.088644	0.117303	0.162411	0.25085

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

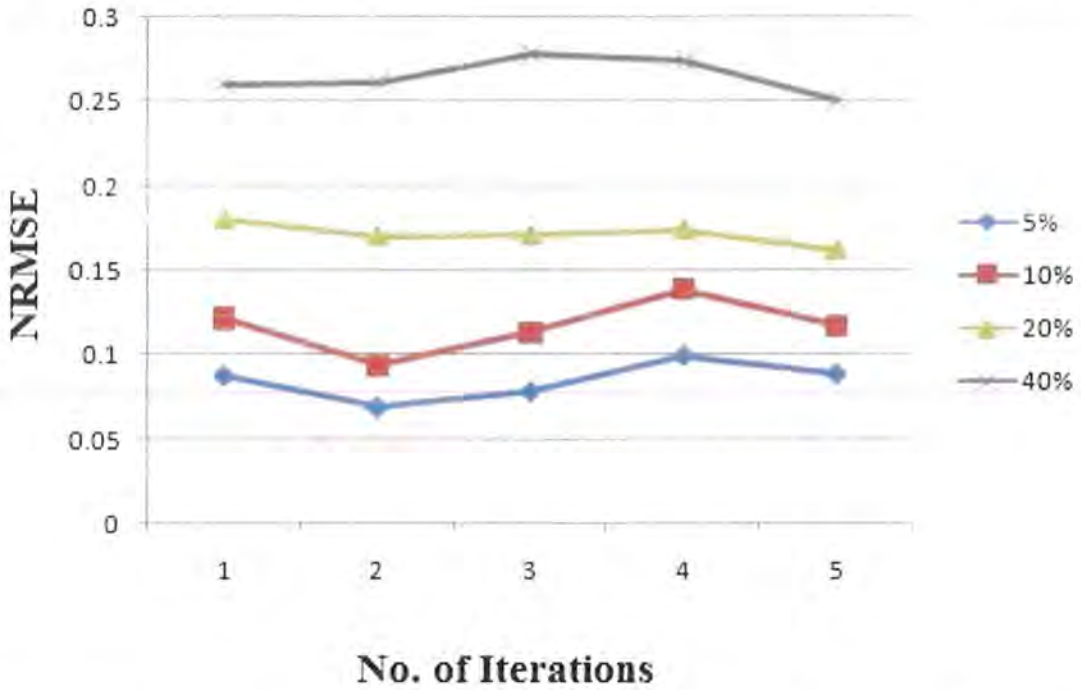


Figure 3.7 Results of SVD for all missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

3.4 Results of LLS Impute

The NRMSE increases drastically for 40% missing data (as shown in Figure 3.8) depicting that the poor estimation of missing values at greater percentage. So the LLS impute method performs better for the data with less missing percentage but its performance gets poor as the missing percentage increases.

Table 3.4 shows values of NRMSE for 5%, 10%, 20% and 40% missing percentages as imputed by LLS Impute method.

Table 3.4 NRMSE values of LLS Impute for first 500 iterations for given missing percentages (5%, 10%, 20%, and 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.125879	0.162152	0.217123	3.86E+08
2	0.102662	0.123407	0.169914	3.22E+19
3	0.10878	0.152962	0.176747	5.694894
4	0.161068	0.201203	0.224032	0.817376
5	0.107877	0.160089	0.185147	0.791383

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

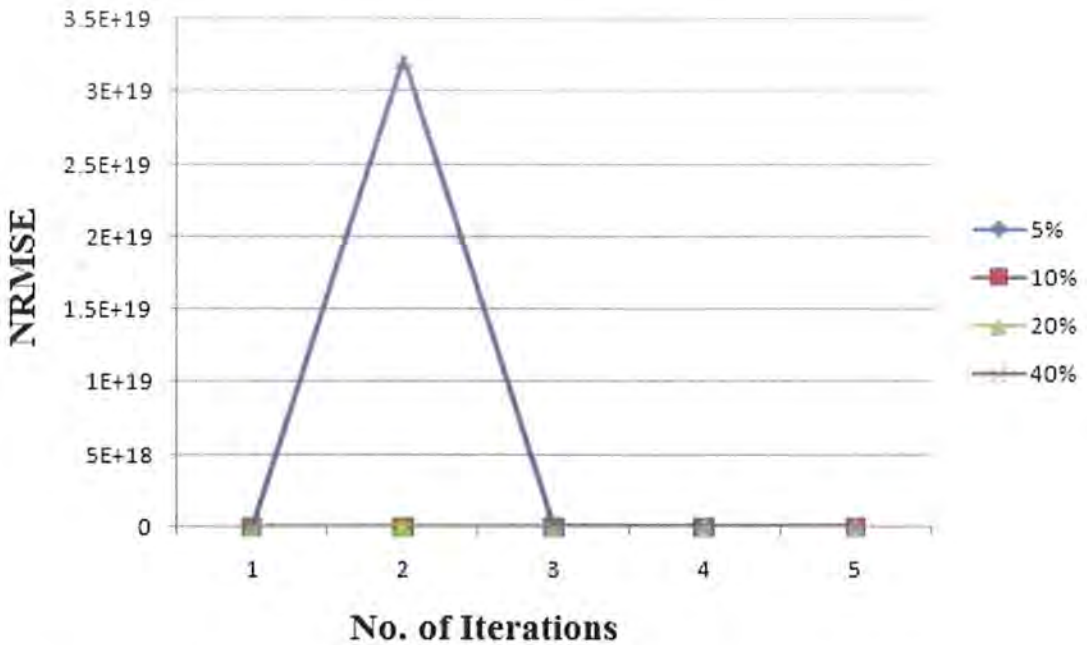


Figure 3.8 Results of LLS impute for given missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

3.5 Results of BPCA

For all the missing percentages, Table 3.5 shows that NRMSE values of BPCA estimation lies close to 1 rather than 0 which means that it does not provide much reliable results of estimation. Figure 3.9 shows that at lower missing percentage, the error rate of estimation is much higher as compared to that at higher percentage i.e., at 40 percent missing data.

Table 3.5 NRMSE values of BPCA for first 500 iterations for given missing percentages (5%, 10%, 20%, 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.77913	0.776885	0.763218	0.723067
2	0.780988	0.779482	0.774488	0.732742
3	0.779226	0.771872	0.764569	0.743539
4	0.770387	0.763297	0.749412	0.701004
5	0.778057	0.775844	0.762161	0.722136

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

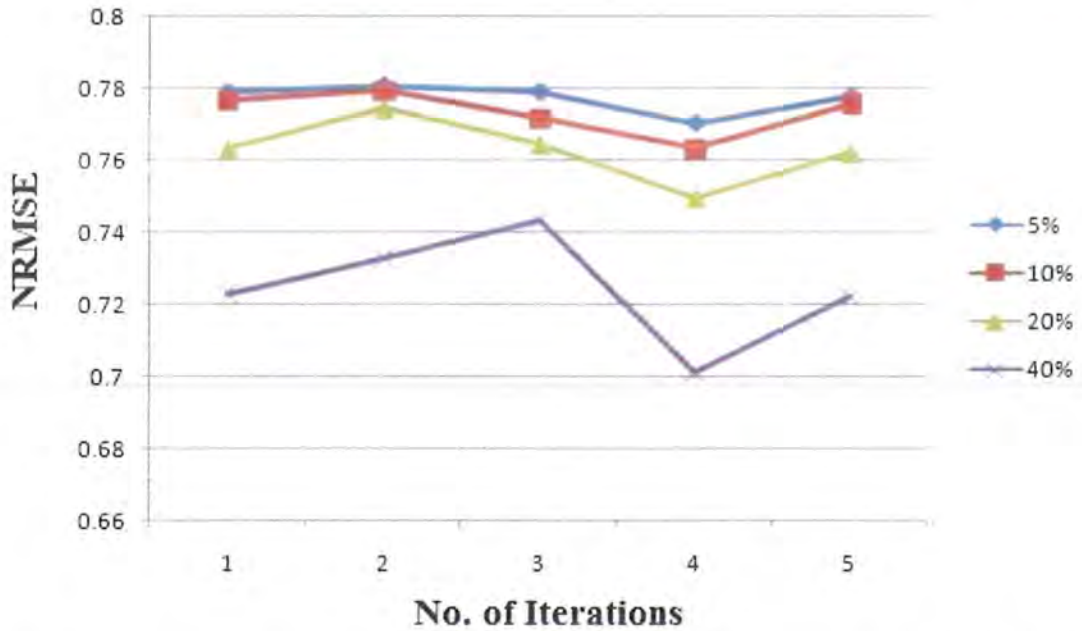


Figure 3.9 Results of BPCA for given missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

3.6 Results of missForest

The NRMSE values of missForest lies close to 1 (as shown in Table 3.6) for all missing percentages which means that it has a poor performance of estimation. Figure 3.10 shows the behavior of NRMSE values for missForest.

Table 3.6 NRMSE values of missForest for first 500 iterations for given missing percentages (5%, 10%, 20%, 40%).

Sr. #	NRMSE at 5%	NRMSE at 10%	NRMSE at 20%	NRMSE at 40%
1	0.665168	0.688834	0.784794	0.870111
2	0.756582	0.72854	0.830988	0.83544
3	0.774101	0.72557	0.789154	0.850927
4	0.771974	0.781072	0.75844	0.845835
5	0.648772	0.700445	0.713732	0.785122

In above table second column represents NRMSE value at 5% missing data, third column represents NRMSE value at 10% missing data, fourth column represents NRMSE value at 20% missing data and fifth column represents NRMSE value at 40% missing data.

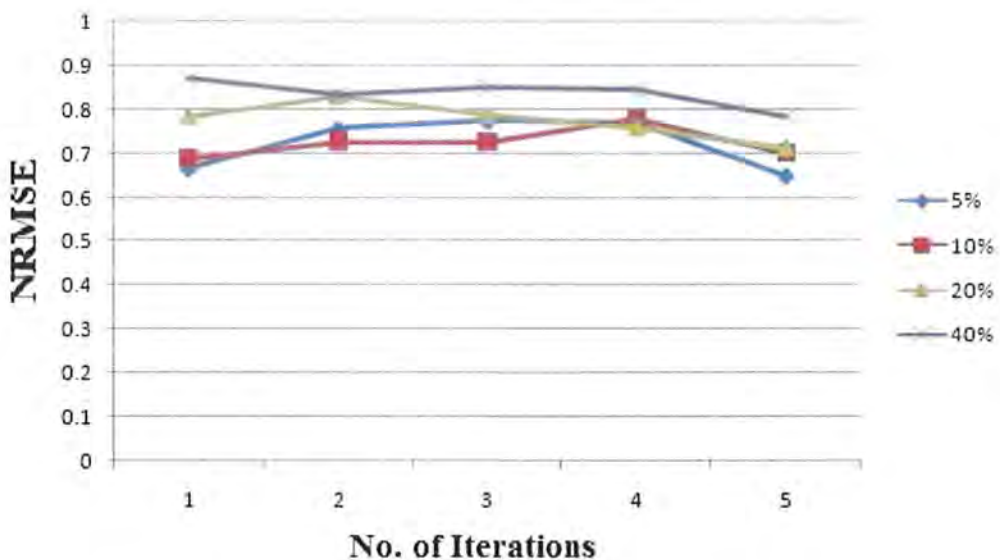


Figure 3.10 Results of missForest for given missing percentages (5%, 10%, 20%, 40%)

In above figure, x-axis represents number of iterations and y-axis represents NRMSE values at 5% (blue line), 10% (red line), 20% (green line) and 40% (purple line) missing percentage.

Chapter 4

Discussion

DISCUSSION

DNA microarray has been widely used by scientists to measure the expression level of large number of genes simultaneously. The DNA microarray produces an image after scanning and the results are then converted to numerical values and saved into a text file. This text file contains numerical values of expression level of every gene. The text file also contains missing values being a very common problem of microarray data analysis. Usually 1 to 10 percent of the data entries are missing in an ordinary microarray which affects upto 95 percent of genes. There are several reasons behind the generation of missing expression values. Several methods have been developed to estimate the MVs. The most trivial methods are row and columns based estimation, replacing the MVs by zero or eliminate the genes which contain MVs. Other computational methods have also been developed to deal with MVs. These methods have some drawbacks (Troyanskaya et al., 2001; Sahu et al., 2011).

The main purpose of this study is to develop a method that can impute MVs more precisely and with less error. The developed method uses Gibbs Bayesian Variable Selection and Linear Regression to estimate MVs. This method has been tested on microarray time series data set of *Saccharomyces cerevisiae*. It has been tested for four different percentages of MVs i.e., 5%, 10%, 20% and 40%. The results of the developed algorithm have been analyzed on the basis of NRMSE between the actual data set and the data set who's MVs have been calculated. The results of the developed algorithm has been compared to five other methods i.e., KNN, SVD, LLSImpute, BPCA and missForest. NRMSE has been used as a metric for comparison.

Table 4.1 shows the average NRMSE values of different methods for all the four percentages of MVs. NRMSE values of developed algorithm lies between 0.03 to 0.06 which is much better than other methods. The usual behaviour of NRMSE values shows that it increases with the increase in the percentage of MVs. Still the developed algorithm has the minimum NRMSE values generated as compared to other methods which show that it outperforms other methods.

Table 4.1 Average NRMSE values of proposed algorithm (GibbsBvs + Regression), KNN, SVD, BPCA, missForest and LLSImpute, for given missing percentages (5%, 10%, 20%, and 40%)

	Proposed algorithm	KNN	SVD	BPCA	missForest	LLS Impute
NRMSE at 5%	0.063232	0.113127	0.084616	0.777557	0.72332	0.121253
NRMSE at 10%	0.036961	0.118968	0.117111	0.773476	0.724892	0.159962
NRMSE at 20%	0.07974	0.133034	0.172096	0.76277	0.775421	0.194593
NRMSE at 40%	0.063733	0.150406	0.26507	0.724498	0.837487	6.45E+18

Rows represent values of NRMSE for 5%, 10%, 20% and 40% missing percentage of values while each column represents different techniques to impute missing values. Each cell represents values of NRMSE for different techniques.

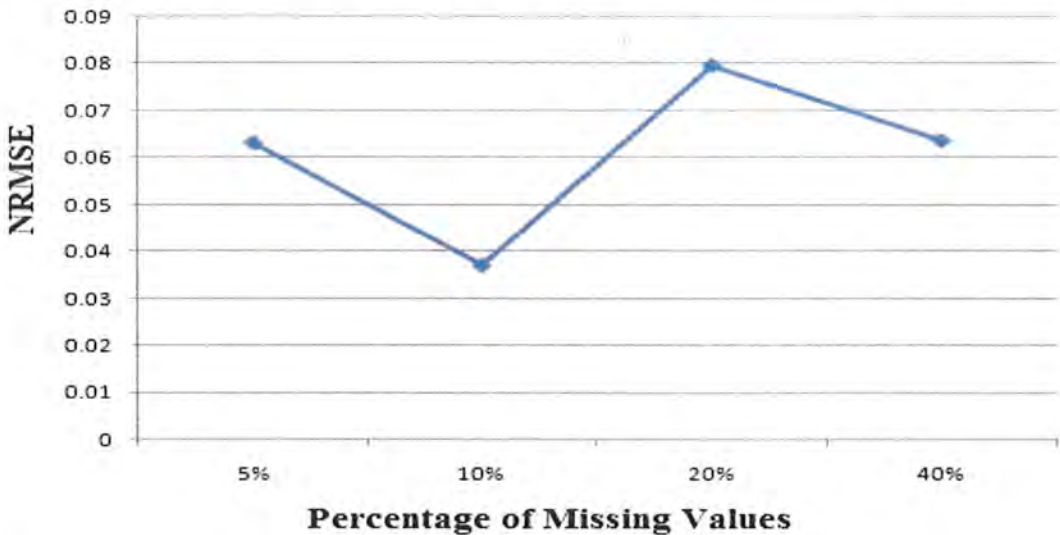


Figure 4.1 Results of proposed algorithm along with increasing percentage of missing values.

In above figure x-axis represents the percentage of missing values (5%, 10%, 20% and 40%) while y-axis represents the NRMSE values obtained at missing percentages by proposed algorithm.

Figure 4.2 shows the comparison of all already existing methods except LLSImpute with the newly developed method. The graph of the LLSImpute has been shown separately in Figure 4.3 as it shows a strange behaviour with the increase in percentage of MVs. It is clear from the figure that the NRMSE values of estimation of MVs by the BPCA and missForest method lies close to 1 hence showing the poor performance of these two methods. While the NRMSE values of estimation of MVs by the SVD and KNN method and the developed algorithm lies close to 0 which shows their better performance. The figure shows that NRMSE values of KNN lies between 0.1 to 0.2 and those of SVD lies between 0 to 0.3 while NRMSE values of the developed algorithm (GibbsBvs+Reg) lies between 0 to 0.1 for all the percentages of MVs. This figure hence shows that the developed algorithm (GibbsBvs+Reg) outperforms all other methods in the estimation of MVs and imputes MVs more precisely as compared to all other methods.

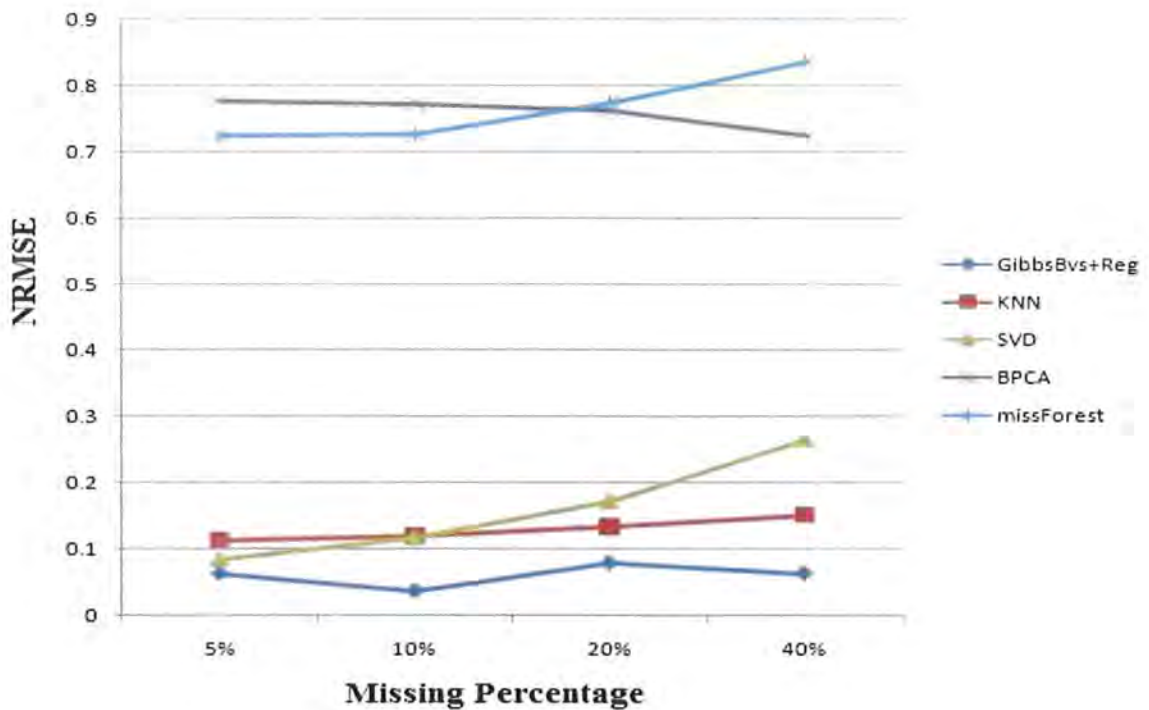


Figure 4.2 Comparison of results of already existing methods (KNN, SVD, BPCA, missForest) with the proposed algorithm (GibbsBvs+Reg)

In above figure x-axis represents the percentage of missing values (5%, 10%, 20% and 40%) while y-axis represents the NRMSE values obtained at missing percentages by different methods i.e., proposed method (GibbsBvs+Reg represented by blue line), KNN (red line), SVD (green line), BPCA (purple line), missForest (light blue line).

Figure 4.3 shows the strange behavior of NRMSE of llsimpute method. The NRMSE values for 5%, 10% and 20% are 0.121253, 0.159962 and 0.194593 respectively while for 40% it is $6.45E+18$. This shows the poor performance of llsimpute at higher percentage of missing values.

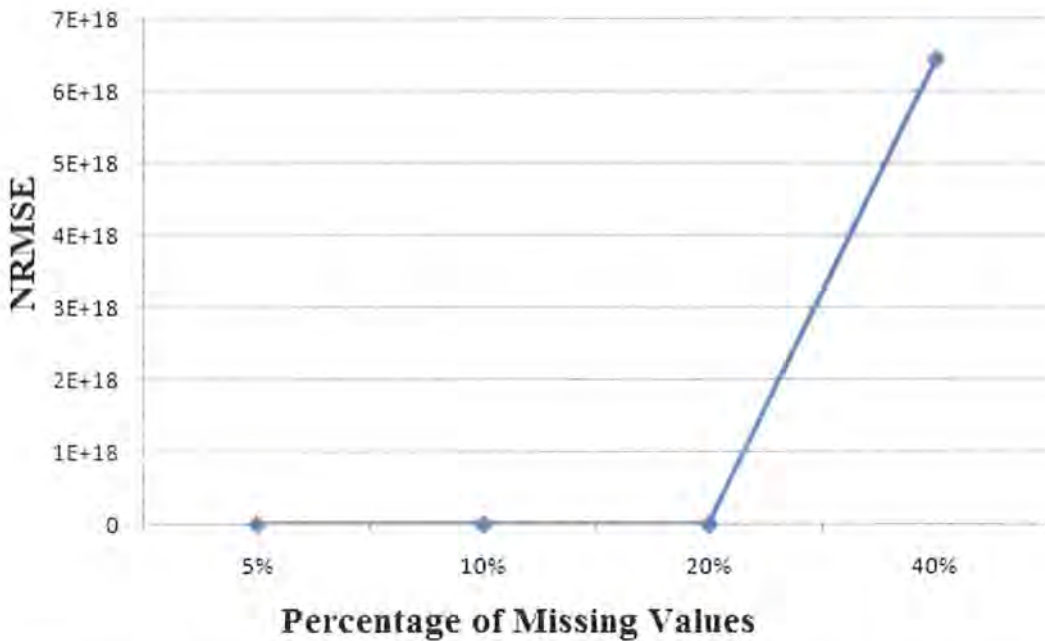


Figure 4.3 Results of LLSImpute method for all percentages of missing values.

In above figure x-axis represents the percentage of missing values (5%, 10%, 20% and 40%) while y-axis represents the NRMSE values obtained at missing percentages.

4.1 Conclusion

In this study the main focus was on statistical challenges faced during the imputation of missing data obtained from microarray. These statistical challenges involve differential gene expression issues. For the imputation of microarray missing data, I developed a method using Gibbs Bayesian Variable Selection to find out the important predictors and then impute missing values based on those predictors. Furthermore the results of this method were compared with previously developed other six methods. The preliminary tests indicate that the developed algorithm

(GibbsBvs+Reg) produces more accurate results as compared to other methods using NRMSE as a metric. The NRMSE for four missing percentages does not drastically increase along with the increase of missing percentage in the case of GibbsBvs+Reg while NRMSE increases along with the increase of missing percentage in case of BPCA, missForest and LLS Impute. Hence the developed algorithm outperformed rest of the methods.

References

- Alter, O., Brown, P.O., Botstein, D., 2000. Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences* 97, 10101-10106.
- Babu, M.M., 2004. Introduction to microarray data analysis. *Computational Genomics: Theory and Application*, 225-249.
- Bø, T.H., Dysvik, B., Jonassen, I., 2004. LSImpute: accurate estimation of missing values in microarray data with least squares methods. *Nucleic acids research* 32, e34-e34.†
- Dalma-Weiszhausz, D.D., Warrington, J., Tanimoto, E.Y., Miyada, C.G., 2006. [1] The Affymetrix GeneChip® Platform: An Overview. *Methods in enzymology* 410, 3-28.
- Dubitzky, W., Granzow, M., Downes, C.S., Berrar, D., 2003. Introduction to microarray data analysis. *A Practical Approach to Microarray Data Analysis*. Springer, 1-46.
- Durrant, G.B., 2009. Imputation methods for handling item-nonresponse in practice: methodological issues and recent debates. *International Journal of Social Research Methodology* 12, 293-304.
- Ehrenreich, A., 2006. DNA microarray technology for the microbiologist: an overview. *Applied microbiology and biotechnology* 73, 255-273.
- Friedland, S., Niknejad, A., Chihara, L., 2006a. A simultaneous reconstruction of missing data in DNA microarrays. *Linear Algebra and its applications* 416, 8-28.

- Friedland, S., Niknejad, A., Kaveh, M., Zare, H., 2006b. An algorithm for missing value estimation for DNA microarray data. *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, IEEE, II-II.
- Gabig, M., Wegrzyn, G., 2001. An introduction to DNA chips: principles, technology, applications and analysis. *ACTA BIOCHIMICA POLONICA-ENGLISH EDITION- 48*, 615-622.
- Gan, X., Liew, A.W.-C., Yan, H., 2006. Microarray missing data imputation based on a set theoretic framework and biological knowledge. *Nucleic acids research* 34, 1608-1619.
- George, E.I., McCulloch, R.E., 1993. Variable selection via Gibbs sampling. *Journal of the American Statistical Association* 88, 881-889.
- George, E.I., McCulloch, R.E., 1997. Approaches for Bayesian variable selection. *Statistica sinica* 7, 339-373.
- Ghoncim, V.F., Solouma, N.H., Kadah, Y.M., 2011. Evaluation of missing values imputation methods in cDNA microarrays based on classification accuracy. *Biomedical Engineering (MECBME), 2011 1st Middle East Conference on*. IEEE., 367-370.
- Heller, M.J., 2002. DNA microarray technology: devices, systems, and applications. *Annual review of biomedical engineering* 4, 129-153.
- Hourani, M.a., El, E.I.M., 2009. Microarray missing values imputation methods: Critical analysis review. *Computer Science and Information Systems* 6, 165-190.

- Ji, R., Liu, D., Zhou, Z., 2011. A Bicluster-Based Missing Value Imputation Method for Gene Expression Data. *Journal of Computational Information Systems* 7, 4810-4818.
- Kim, H., Golub, G.H., Park, H., 2005. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics* 21, 187-198.
- Kostrzynska, M., Bachand, A., 2006. Application of DNA microarray technology for detection, identification, and characterization of food-borne pathogens. *Canadian journal of microbiology* 52, 1-8.
- Li, F., Zhang, N.R., 2010. Bayesian variable selection in structured high-dimensional covariate spaces with applications in genomics. *Journal of the American Statistical Association* 105.
- Liew, A.W.-C., Law, N.-F., Yan, H., 2011. Missing value imputation for gene expression data: computational techniques to recover missing data from available information. *Briefings in bioinformatics* 12, 498-513.
- Little, R.J., Rubin, D.B., 2014. *Statistical analysis with missing data*. John Wiley & Sons.
- Macgregor, P. F., & Squire, J. A. 2002. Application of microarrays to the analysis of gene expression in cancer. *Clinical Chemistry*, 48(8), 1170-1177.
- Majtan, T., Bukovska, G., Timko, J., 2004. DNA microarrays—techniques and applications in microbial systems. *Folia microbiologica* 49, 635-664.
- Mayer, B., 2013. Hot Deck Propensity Score Imputation For Missing Values. *Science Journal of Medicine and Clinical Trials* 2013.

- Mehta, J.P., 2011. Microarray Analysis of mRNAs: Experimental Design and Data Analysis Fundamentals. Gene Expression Profiling, Springer, 27-40.
- Miller, M.B., Tang, Y.-W., 2009. Basic concepts of microarrays and potential applications in clinical microbiology. *Clinical microbiology reviews* 22, 611-633.
- Molaezadeh, S., Moradi, M., 2006. A new non parametric method for estimating missing value in microarray data by GO clustering. Proc. Cairo International Biomedical Engineering Conference.
- Ness, S.A., 2006. Basic microarray analysis. *Bioinformatics and Drug Discovery*. Springer, pp. 13-33.
- Ness, S.A., 2007. Microarray analysis: basic strategies for successful experiments. *Molecular biotechnology* 36, 205-219
- Nguyen, D.V., Bulak Arpat, A., Wang, N., Carroll, R.J., 2002. DNA microarray experiments: biological and technological aspects. *Biometrics* 58, 701-717.
- Piyushimita, 2010. Evaluation of alternative data imputation strategies: A case study of motor carrier safety. *Transportation Letters* 2, 199-216.
- Qin, L., Rueda, L., Ali, A., Ngom, A., 2005. Spot detection and image segmentation in DNA microarray data. *Applied Bioinformatics* 4, 1-11.
- Rasooly, A., Herold, K.E., 2008. Food microbial pathogen detection and analysis using DNA microarray technologies. *Foodborne pathogens and disease* 5, 531-550.

- Sahu, M.A., Swarnkar, M.T., Das, M.K., 2011. Estimation methods for microarray data with missing values: a review. *International Journal of Computer Science and Information Technologies* 2, 614-620.
- Sebastiani, P., Gussoni, E., Kohane, I.S., Ramoni, M.F., 2003. Statistical challenges in functional genomics. *Statistical Science*, 33-60.
- Snijders, A., Meijer, G., Brakenhoff, R., Van Den Brule, A., Van Diest, P., 2000. Microarray techniques in pathology: tool or toy? *Molecular Pathology* 53, 289.
- Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., 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.
- Statistical Package, R., 2009. R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing.
- Stekhoven, D.J., Bühlmann, P., 2012. MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics* 28, 112-118.
- Tchagang, A.B., Pan, Y., Famili, F., Tewfik, A.H., Benos, P.V., 2010. Biclustering of dna microarray data: Theory, evaluation, and applications.
- Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B., 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17, 520-525.

Van Buuren, S., 2012. Flexible imputation of missing data. CRC press.

Yoon, D., Lee, E.-K., Park, T., 2007. Robust imputation method for missing values in microarray data. BMC bioinformatics 8, S6.

Zhang, X., Song, X., Wang, H., Zhang, H., 2008. Sequential local least squares imputation estimating missing value of microarray data. Computers in biology and medicine 38, 1112-1120.

Webliography

Retrieved February 5, 2015, from INTECH:

<http://www.intechopen.com/books/howtoreference/gene-therapy-developments-and-future-perspectives/impacts-of-dna-microarray-technology-in-gene-therapy>

Retrieved June 15, 2014, from <http://www.exploredata.net/Downloads/Gene-Expression-Data-S>

Appendix

R Package	Usage
impute	Used for k nearest neighbour imputation of missing values.
HotDeckImputation	Uses column mean of complete cases to impute missing values.
peaMethods	This method provides a set of different PCA implementations to calculate the missing values. It includes SVD, BPCA and LLSImpute methods.
missForest	Used to impute missing values in case of mixed type data.
BVS	Used to analyze variable selection problem in linear regression models from Bayesian perspective.