”A principle is the expression of perfection, and as imperfect beings like us cannot practise perfection, we devise at every moment limits of its compromise in practice.” - Mohandas Karamchand Gandhi

University of Alberta

ROBUST LEARNING ALGORITHMS FOR BIOENGINEERING SYSTEMS

by

Venkat R. Nadadoor Srinivasan

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Process Control

Department of Chemical and Materials Engineering

Spring 2012

Edmonton, Alberta

To **my parents and my sister**

Three most important people in my life

Abstract

Biological engineering is a domain of study that involves applying known engineering principles to biological systems. Qualitative studies in the field of biology have undergone tremendous advancements in the last two decades but quantitation is still in its early stages due to various complexities involved in its design, control, and operation. The current state of research in the field of bioengineering involves mostly elementary quantitation of biological systems without a strong grasp into the fundamentals of engineering. Advanced learning algorithms can help overcome some of the problems generally associated with biological systems including model complexity, noisy measurements, and data scarcity. In the current study, bioengineering problems are viewed from process systems engineering perspective with a focus on three aspects: modeling, monitoring, and fault detection. The three representative bioengineering problems chosen to cover the three aforementioned aspects are:

- Modeling a gene network: Accurate inference of gene network can provide

information that can lead to new ideas for treating complex diseases. A novel algorithm for building gene networks from microarray datasets using a first principles differential equations model is proposed. The proposed algorithm was able to obtain a good estimate of the gene connectivity matrix for an experimental dataset on a nine gene network in *Eschericia coli*.

- Monitoring a microalgal bioreactor system: Monitoring of process conditions in algal cultures helps in maximizing oil productivity. A support vector regression based algorithm is proposed for monitoring the culture conditions of an algal bioreactor system. The multivariate sensor built using an experimental dataset gave good predictions for the concentrations of biomass, glucose and percentage oil content.

- Detection of transplant rejection: Early detection of graft rejection is mandatory to effectively treat and prevent cardiac dysfunction. An algorithm based on hypothesis testing is proposed for detecting biomarkers useful for detection of rejection. The chosen biomarkers are validated on publicly available microarray datasets. For these datasets, the biomarkers obtained based on the proposed method were able to achieve a good separation between the successful and failed transplant classes.

The methodologies and strategies proposed in this thesis have helped in the modeling, monitoring, and fault detection of bioengineering systems.

Acknowledgements

I am grateful to my supervisors Dr. Sirish Shah and Dr. Amos Ben-Zvi for the support and guidance they rendered during my program and showing full faith in me. Without their support and keen interest in my work, this dissertation would not have been possible. As a student fresh out of my undergraduate studies, I was full of energy and my supervisors gave direction to that energy and helped me become a thorough professional.

I would like to express my gratitude to Dr. Thomas Mueller, in the Department of Medicine at the University of Alberta, for providing me the transplantation dataset and for many illuminating suggestions regarding the microarray experiment and gene networks.

I am thankful to Dr. Hector De la Hoz for being a friend, guide, and critic and making my stay at the University of Alberta a very memorable one. He has been one of the source of inspiration and courage throughout my research program and

also provided me his experimental Raman spectroscopy data for analysis. I am also thankful to Dr. William McCaffrey for sparing his precious time in helping me understand some of the concepts of the microalgal bioreactor system.

My sincere gratitude to the Department of Chemical and Materials Engineering at University of Alberta, NSERC, Matrikon Inc. and Suncor Energy Inc. for supporting me financially. A huge thanks to the Computer Process Control (CPC) group at the University of Alberta for all the good times and friendships which I would carry forward for the rest of my life.

I would like to thank all my friends Karteek, Kasra, Sandeep, Sankar, Varma, Ananth, Sai, Sriram, Hari, Seyi, Sheehan, John, David, Hua, and Shima for making my stay in Edmonton a very special one. I will cherish their friendships and support throughout my life and carry forward the great memories. I would like to thank my childhood friends Omer and Damodar for being my friends for 20 odd years through thick and thin.

Last but certainly not the least I am would like to convey my love to my parents and sister. They have been the pillars of strength in my life and are the reason I have come this far in life. I could not have achieved anything without their love and care and I am thankful to have been part of a great family.

Contents

List of Figures

List of Tables

1

Introduction

Studies in the biological field have undergone significant changes in the last forty years. Biology has expanded from the usual area of qualitative scientific fact accumulation towards a more advanced field involving quantitation of the new knowledge obtained. The methods developed for quantitative prediction of the biological processes in turn lead to developing new tools for controlling these processes. This

lead to the design of various new biological-based products and thus ushered in a new domain in engineering involving the field of biology (Johnson and Phillips, 1995).

Biological engineering or Bioengineering covers broad range of fields including bioprocess engineering, biomedical engineering, systems biology, metabolic engineering, tissue engineering, etc. Bioengineering involves manipulating biological information, constructing bio-materials, processing bio-chemicals, producing biofuels, and help maintain or enhance human health. However, the tools developed for fast and reliable engineering of biological systems are quite limited. There are major challenges that greatly limit the engineering of biology including an inability to avoid or manage biological complexity, the tedious and unreliable construction and characterization of synthetic biological systems, the apparent spontaneous physical variation of biological system behaviour, and evolution (Endy, 2005). Some of the complexity of working with these biological applications include:

- **High amount of noise:** Understanding and modeling biological systems also involves taking into account the occurrence of noise and fluctuations in the system. In other words, biological systems and processes are inherently noisy and have to be addressed carefully so as to avoid undesirable results (Herranz and Cohen, 2010). Noise existing in biological systems is classified as external noise due to environmental fluctuations or internal noise due to certain regulatory molecules (Tian, 2010; J. Hasty et al., 2000). One of the examples of noise in the biological application involves building biochemical networks from gene expression microarray dataset which include

the measurement and hybridization noise (Tu et al., 2002; Thattai and van Oudenaarden, 2001). For reducing the effect of noise and in turn improving the signal to noise ratio, standard statistical and signal processing techniques including principal components analysis (PCA) (Hotelling, 1933; Nomikos and MacGregor, 1994), partial least squares (PLS) (Wold, 1966; Mejdell and Skogestad, 1991), filtering methods (Cleveland, 1979; Savitzky and Golay, 1964; Kalman, 1960; Seborg et al., 2004) applied in various process systems engineering applications can be used.

- **Nonlinearity:** The majority of applications in the field of biological and medical sciences are predominantly nonlinear complex systems (Hunter and Korenberg, 1986). Advanced statistical and machine learning algorithms used in process systems engineering applications including kernel PCA (Lee et al., 2004), support vector machines (Chitralekha and Shah, 2010) can be used for modeling such systems.

- **Data Scarcity and High Dimensionality:** Biological systems are usually characterized by high dimensional scarce dataset including applications involving use of gene expression microarrays datasets for building regulatory network (Yeung et al., 2002; Wang et al., 2006), identifying transplant rejection and designing durable biomaterials (Darrabie et al., 2005; Pickup et al., 2007). Statistical techniques including principal components analysis (PCA), partial least squares (PLS), and independent component analysis (ICA) can be used when dealing with such datasets.

Considerable research effort has been spent on focusing on important and potential

applications in bioengineering including production of biofuels and bioproducts from microalgae, identification of transplant rejection in patients and manufacture of biomaterials for biomedical application. Many analogies can be drawn between the existing process engineering applications and applications in the field of biological engineering. The following section talks about the different branches in the field of process systems engineering and its analogous applications in the field of biological engineering

1.1 Branches in Process Engineering

Process engineering or process systems engineering (PSE) is a branch of chemical engineering which deals with the understanding and development of systematic procedures for the design and operation of chemical process systems, ranging from microsystems to industrial scale continuous and batch processes (Grossmann and Westerberg, 2000). The various different areas in process engineering as characterized by Grossmann and Westerberg (2000) are as follows:

1.1.1 Process Design

The first and foremost branch in process systems engineering is process and product design. Process or product design involves deciding on the unique characteristics and features of the desired product. One of the major features in process design is not only to be innovative but also to be cost effective. Another major challenge that will remain is the design of sustainable and environmentally benign processes. An

analogy can be drawn between PSE and biological engineering in the application of design and analysis of metabolic networks. However, the design of metabolic networks can be more elaborate and convoluted when compared to design of PSE systems.

1.1.2 Modeling

One of the aspects of paramount importance in PSE is modeling. Process modeling attempts to relate a desired quantity based on the available variables which are deemed important for the purpose of modeling. The purpose of a model is to reduce the complexity of understanding a phenomenon by narrowing down the aspects that influence its relevant behavior. Curtis et al. (1992) states that a process model is an abstract description of an actual or proposed process that represents the chosen process elements that are important to the purpose of the model and can be enacted by a human or machine. For modeling various aspects involved in a bioengineering applications, more flexible modeling environments will be required that can accommodate a greater variety of models.

1.1.3 Process control

Process control involves the use of statistical and engineering principles to monitor the process and maintain it at the desired performance/ operating condition safely and efficiently. The significant accomplishments in the field of process control include model predictive control, robust control, nonlinear control, statistical process control, and process monitoring. Achievements in advanced process control and

process monitoring can be applied towards new applications in bioprocess systems and biomedical engineering.

1.1.4 Process Operations

The area of process operations, has a shorter history than process design and control. The broad area of process operations includes data reconciliation, real-time optimization, fault detection and diagnosis, and process planning and scheduling. Efficient fault detection and diagnosis is of increasing importance when dealing with applications in biomedical engineering. For example, identification of disease (fault) in a patient helps in early diagnosis which in turn can help in speedy recovery.

1.2 Thesis Contribution

The aim of this work is to apply well known statistical and machine learning techniques including principal components analysis, partial least squares, support vector learning, clustering algorithm, and hypothesis testing to different applications in the biological engineering. The specific objectives of this work fall in the following categories:

1) **Process design and modeling:** Obtain a gene regulatory network from gene expression data using a first principles differential equation (DE) model.

2) **Process Monitoring:** Develop an online multivariate sensor for monitor-

ing the chemical components in an algal bioreactor system.

3) **Process Operations:** Develop a novel strategy for identifying candidate biomarkers which aid in the detection and diagnosis of transplant rejection.

References

A. T. Johnson, W. M. Phillips, Philosophical foundations of biological engineering, Journal of Engineering Education 84 (1995) 311–318.

D. Endy, Foundations for engineering biology, Nature 438 (2005) 449–453.

H. Herranz, S. M. Cohen, MicroRNAs and gene regulatory networks: managing the impact of noise in biological systems, Genes & Development 24 (2010) 1339–1344.

T. Tian, Robustness of mathematical models for biological systems, Gene 45 (2010) 565–577.

J. J. Hasty, J. Pradines, M. Dolnik, J. J. Collins, Noise-based switches and amplifiers for gene expression, Proceedings of the National Academy of Sciences 97 (2000) 2075–2080.

Y. Tu, G. Stolovitzky, U. Klien, Qualitative noise analysis for gene expression microarray experiments, Proceedings in National Academy of Science 99 (2002) 14031–14036.

M. Thattai, A. van Oudenaarden, Intrinsic noise in gene regulatory networks, Proceedings in National Academy of Science 98 (2001) 8614–8619.

H. Hotelling, Analysis of a complex of statistical variables into principal components, Journal of Educational Psychology 24 (1933) 417–441.

P. Nomikos, J. F. MacGregor, Monitoring batch processes using multiway principal component analysis, AICHE Journal 40 (1994) 1361–1375.

H. Wold, Estimation of principal components and related models by iterative least squares, In Multivariate Analysis (1966) 391–420.

T. Mejdell, S. Skogestad, Estimation of distillation compositions from multiple temperature measurements using partial-least-squares regression, Industrial & Engineering Chemistry Research 30 (1991) 2543–2555.

W. S. Cleveland, Robust locally weighted regression and smoothing scatterplots, J. Am. Stat. Assoc. 74 (1979) 829–836.

A. Savitzky, M. J. E. Golay, Smoothing and differentiation of data by simplified least squares procedures., Analytical Chemistry 36 (1964) 1627–1639.

R. E. Kalman, A new approach to linear filtering and prediction problems, Transactions of the ASME - Journal of Basic Engineering (1960) 35–45.

D. E. Seborg, T. F. Edgar, D. A. Mellichamp, Process dynamics and control, John Wiley and Sons, New York, 2004.

I. W. Hunter, M. J. Korenberg, The identification of nonlinear biological systems: Wiener and hammerstein cascade models, Biological Cybernetics 55 (1986) 135–144.

J. M. Lee, C. K. Yoo, S. W. Choi, P. A. Vanrolleghem, I. B. Lee, Nonlinear process monitoring using kernel principal component analysis, Chemical Engineering Science 59 (2004) 223 – 234.

S. B. Chitralekha, S. L. Shah, Application of support vector regression for developing soft sensors for nonlinear processes, Can. J. Chem. Eng. 88 (2010) 696–709.

M. K. Yeung, T. J, C. J. J, Reverse engineering gene networks using singular value decomposition and robust regression, Proceedings of the National Academy of Sciences 99 (2002) 6163–6168.

Y. Wang, T. Joshi, D. Xu, X. S. Zhang, L. Chen, Supervised inference of gene regulatory networks by linear programming, in: Computational Intelligence and Bioinformatics, volume 4115, Springer Berlin / Heidelberg, 2006, pp. 551–561.

M. D. Darrabie, W. F. K. Jr., E. C. Opara, Characteristics of Poly-l-Ornithine-coated alginate microcapsules, Biomaterials 26 (2005) 6846 – 6852.

D. M. Pickup, I. Ahmed, P. Guerry, J. C. Knowles, M. E. Smith, R. J. Newport, The structure of phosphate glass biomaterials from neutron diffraction and ^{31}p nuclear magnetic resonance data, Journal of Physics: Condensed Matter 19 (2007) 696–709.

I. E. Grossmann, A. W. Westerberg, Research challenges in process systems engineering, AIChE Journal 46 (2000) 1700–1703.

B. Curtis, M. I. Kellner, J. Over, Process modeling, Communications of the ACM 35 (1992) 75–90.

2

Inferring Gene Networks using Robust Statistical Techniques [1]

2.1 Introduction

Gene expression profiling has produced insights into complex biological systems. In the field of genomics, gene expression profiling has been used to understand the mechanisms underlying biological processes including allograft rejection (Erickson et al., 2003), and breast cancer progression (Ma et al., 2009).

In this chapter, a novel algorithm is proposed for reverse engineering of gene reg-

[1] A version of this chapter has been published as: V. R. Nadadoor, A. Ben-Zvi, and S. L. Shah, "Inferring Gene Networks Using Robust Statistical Techniques", Statistical Applications in Genetics and Molecular Biology: Vol. 10: Iss. 1, 2011.

ulatory network from gene expression data obtained from microarray experiments. Microarray experiments have allowed the gene expression profiles to be measured for the whole genome (thousands of genes) simultaneously under a variety of conditions (Tu et al., 2002). Microarray technology has been applied to biological processes including acute allograft rejection (Stegall et al., 2002), during mouse and human pregnancy (Bethin et al., 2003), and yeast sporulation (Chu et al., 1998). Data from microarray experiments may be arranged in the form of a rectangular matrix containing expression level of genes (rows) at different experimental conditions (columns) (Troyanskaya et al., 2001). The number of time-samples (i.e, microarray slides) used to profile gene expression is typically less than the number of genes profiled. As a result, the data matrix from microarray experiments will often have more rows (i.e., genes) than columns (i.e., time points).

Much research effort has focused on estimating (or reverse-engineering) gene networks from gene expression data obtained from micro-array experiments (Yeung et al., 2002; Gardner et al., 2003; Liu et al., 2006) with applications including human B cells (Schadt and Lum, 2006), gap gene network of *Drosophila melanogaster* (Basso et al., 2005). Reverse engineering is the process of elucidating the structure of the system by reasoning backwards from observations of its behavior (Hartemink, 2005). Reverse engineering of gene network involves estimating the connectivity matrix, given observations of the system over time (D'haseseleer et al., 2000; Tegner et al., 2003; Yeung et al., 2002). These gene networks are capable of showing the interaction of a large number of genes in a concise manner (Brazhnik et al., 2002). Several graphical methodologies including graphical Gaussian (GG) ((Magwene and Kim, 2004)) and dynamic Bayesian network (DBN) (Zou and Conzen, 2005) modeling have been applied for reverse engineering of gene networks. Due to the high computational complexity and need for high number of data points, both these methods can be used only for small networks (gene networks of size smaller than 10) (He et al., 2009), (Hecker et al., 2009), (Bansal et al., 2007). He et al. (2009) also state that in the GG and the DBN method, the resultant gene network obtained is undirected. Other known methodologies including Boolean networks (Liang et al., 1998) and system of linear ordinary differential/algebraic equations (Yeung et al., 2002; Tegner et al., 2003; Bansal

et al., 2006; Gardner et al., 2003; Liao et al., 2003; Foteinou et al., 2009) have been proposed to reverse engineer the gene network from gene expression data obtained from these microarray experiments. Boolean network methodology is limited, as they give undirected networks using a binary set of variable $x_i \in \{0,1\}$, to represent the presence of a connection between genes ((Hecker et al., 2009)). Also, the method of inferring gene networks from linear algebraic equations as proposed in Liao et al. (2003) and Foteinou et al. (2009), need *apriori* information for estimation of gene connectivity matrix. In this work, the research effort is concentrated on the approach of inferring gene networks from ordinary differential equations (ODEs) without any given *apriori* information regarding the network.

Ordinary differential equations (ODEs) have been used to model biological networks (Gardner et al., 2003; Yeung et al., 2002; Bansal et al., 2006; Kim et al., 2007). For a network of n genes, the corresponding ODE system is given by (McAdams and Arkin, 2000; Jong, 2002):

$$\begin{aligned} \dot{x}_i(t) = \quad & f_i(x_1(t),\ldots,x_i(t),\ldots,x_n(t),u_1(t),\ldots,u_n(t)) \\ & i = 1,2,...,n \end{aligned} \tag{2.1}$$

where each x_i is a function of time representing expression levels of the i^{th} gene; and f_i is a nonlinear function representing the time-derivative in the expression level of the i^{th} gene. The measured gene expression levels, $\tilde{x}_i$'s, are corrupted with measurement noise (Tu et al., 2002; Thattai and van Oudenaarden, 2001), and therefore can be written as an added sum of the signal and noise components as follows:

$$\tilde{x}_i(t) = x_i(t) + \xi_i(t) \tag{2.2}$$

where $\tilde{x}_i$ is a function of time representing noisy expression levels of the i^{th} gene and $\xi_i(t)$ is a function of time representing the measurement noise in the expression levels of the i^{th} gene.

The system of nonlinear ODEs as described in Equation 2.1, operating around a

hyperbolic rest point [2] can be approximated by a system of linear ODEs ((Kreyszig, 1999)).

$$\dot{x}(t) = A_{n\times n}x(t) + B_{n\times p}u(t) \;\; \forall\, t \in T = [0, t_f]. \tag{2.3}$$

$$\tilde{x}(t) = x(t) + \xi(t) \tag{2.4}$$

where A is the connectivity matrix of the n genes; B is the perturbation matrix; $x(t)$ is a function of time representing noise-free expression levels of the n genes (i.e, $x(t) = [x_1(t)\; x_2(t)\; \ldots\; x_n(t)]^T$); $\tilde{x}(t)$ is a function of time representing noisy expression levels of the n genes; $\xi(t)$ is a function of time representing the measurement noise in the expression levels of the n genes; and $u(t)$ is the input function, which is the perturbation vector, at time t. The input function, $u(t)$,is a $p \times 1$ vector, containing the information regarding all perturbations at time t and is typically a constant vector perturbing a select set of p genes, as perturbation of all the genes in the network is not feasible (Bansal et al., 2007, 2006). Estimation of the connectivity matrix A, from Equation 2.3 has been proposed by Gardner et al. (2003), Yeung et al. (2002) and Bansal et al. (2006).

In Gardner et al. (2003), Equation 2.3 is solved at steady state, i.e. $\dot{x}(t) = 0$, using multiple linear regression. The method requires perturbation of all genes in the network, which is not always feasible in a gene expression experiment. Furthermore, obtaining a steady state data is expensive as it requires performing multiple perturbations to the cell (Bansal et al., 2007).

In Yeung et al. (2002), an algorithm is proposed for estimating the entries of the connectivity matrix A. In this approach the gene expression levels, $x(t)$, is sampled at time $t_j = \{t_1 < t_2 < .. < t_m\}$ with $t_j \in T$, and is written in the form of a gene expression matrix, $X_{n\times m}$, with rows indicating the various genes and columns indicating different time samples. That is, each cell in the gene expression matrix represent expression level of that particular gene at a given time. Typically due to high experimental costs, the number of samples are far fewer than the number of

[2] Let x_0 be the rest point for the differential equation $\dot{x} = r(x,u)$ (i.e. $r(x_0,u) = 0$). The point x_0 is called the hyperbolic rest point if every eigenvalue of $M = \frac{\partial r}{\partial x}(x_0)$ is non-zero.(Chicone, 1999)

genes ($n >> m$). In this respect this is an underspecified estimation problem or there are more unknowns than the number of equations and therefore it is not possible to obtain a unique identification solution for A.

$$X_{nxm} = \begin{array}{c} \longrightarrow t \\ \left(\begin{array}{ccc} x_1(t_1) & .. & x_1(t_m) \\ x_2(t_1) & .. & x_2(t_m) \\ : & .. & : \\ x_i(t_1) & .. & x_i(t_m) \\ : & .. & : \\ x_n(t_1) & .. & x_n(t_m) \end{array} \right) \end{array} \begin{array}{l} \\ \\ \\ \\ \\ \\ \downarrow \text{Genes} \end{array}$$

Equation 2.3, is rewritten in a matrix form as shown:

$$\dot{X}_{n\times m} = A_{n\times n}X_{n\times m} + B_{n\times p}U_{p\times m} \tag{2.5}$$

where A is the gene connectivity matrix, and $B = [b_1, \ldots, b_p]$ is the input (or external stimuli) matrix. The goal of reverse engineering is to estimate each of the entries in matrix A. However, for a typical experimental data set, the number of time samples, m, is fewer than the number of genes, n. Therefore, the maximum number of independent equations implied by System 2.5 (i.e., $n \times m$) is less than the number of connections in A (i.e., $n \times n$). As a result, there exists several solutions for A in Equation 2.5. Yeung et al. (2002) discuss a methodology to reverse-engineer gene networks in Equation 2.5, using singular value decomposition (SVD) and robust regression. The method suggested in Yeung et al. (2002) is computationally efficient for larger gene expression datasets. One of the big drawback of the method is that the time derivative matrix, $\dot{X}$, is estimated using linear interpolation. For a gene expression data, which are inherently noisy, the linear interpolation strategy could lead to erroneous results (Bansal et al., 2006).

In Bansal et al. (2006), an algorithm TSNI (Time Series Network Identification) is proposed to infer the gene network from a linear ODE by perturbing any one gene in the network. The method provides an effective way for estimating the gene network

and the perturbation matrix. However, the method does not provide a statistically significant approach for obtaining a sparse network and needs apriori information regarding the connections.

In this work, the method of partial least squares is applied to obtain the gene connectivity matrix (gene network). The proposed algorithm combines statistical tools including leave-one-out jackknifing and the Akaike information criterion (AIC) to ensure that the entries in the obtained gene connectivity matrix are statistically significant. To the best of our knowledge these three methods have not been collectively applied in studies concerned with gene network. The proposed algorithm provides a robust estimation of the connectivity matrix in the presence of measurement noise. A significant part of the study is dedicated to comparing and highlighting the superior performance of the proposed method in comparison with the methods available in the literature.

2.1.1 Partial Least Squares Regression (PLSR)

Typically time series microarray data are characterized by a large number of genes, n, and a few measurements, m ($m << n$). Therefore, well established dimension reduction tools including PCR (principal component regression) and partial least squares regression (PLSR) are used for performing multivariate regression in the reduced dimension space (Pihur et al., 2008). PLSR was first proposed by Herman Wold during mid-sixties (Wold, 1966) and subsequently found success in various applications in the field of chemometrics (Wold et al., 2001), neuro imaging (McIntosh and Lobaugh, 2004), and process control (Dayal and MacGregor, 1997). The PLSR algorithms have also found applications in the field of systems biology as an exploratory tool for potential gene-gene interactions (Datta, 2001; Pihur et al., 2008).

As in the case of multiple linear regression (MLR), the main purpose of partial least squares regression (PLSR) is to build a linear model, $Y = Z\beta + \zeta$. In this work, Y is an $(m-1) \times n$ variables response matrix, Z is an $(m-1) \times (n+p)$ variables predictor matrix, β is a $(n+p) \times n$ regression coefficient matrix, and ζ is

a noise term for the model which has the same dimensions as Y. The partial least squares model can be considered as consisting of outer relations for both the Z and Y matrices and an inner relation linking them (Geladi and Kowalski, 1986). The outer relations for the Z, and Y matrices are built using the principal components analysis, as follows:

$$Z = TP^T + E = \sum t_h p_h^T + E \tag{2.6}$$

$$Y = UQ^T + F^\star = \sum u_h q_h^T + F^\star \tag{2.7}$$

where T, P, and E are the score, loading, and the error matrices of Z, respectively; U, Q, and $F^\star$ are the score, loading, and the error matrices of Y, respectively. An inner relationship is obtained between the two score matrices U and T. For example, a simple inner relation is a linear one.

$$\hat{u}_h = b_h t_h \tag{2.8}$$

A Graphical representation of PLSR algorithm is presented in Figure 2.1.

In this work, a SIMPLS algorithm is used to obtain the gene connectivity matrix from a linear ODE. SIMPLS algorithm was first proposed by Sijmen de Jong as an alternative approach to NIPALS partial least squares regression. A detailed version of the SIMPLS algorithm is given in (Jong, 1993).

2.1.2 Leave-one-out Jackknifing

While the PLS algorithm can be used to obtain a gene connectivity matrix, it cannot be used to guarantee that all parameters in a model are statistically significant (Pihur et al., 2008). Leave-one-out jackknifing is a commonly used technique in statistical analysis that can be used for judging whether a particular entry in the connectivity matrix is spurious (de la Fuente and Makhecha, 2006; Fisher, 1973; Gardner et al., 2003). In this work, the assertion that μ (the mean estimate of a coefficient) is equal to zero, is the null hypothesis. The alternative hypothesis is that μ is not equal to zero. In this work, a normal distribution will be assumed for the mean

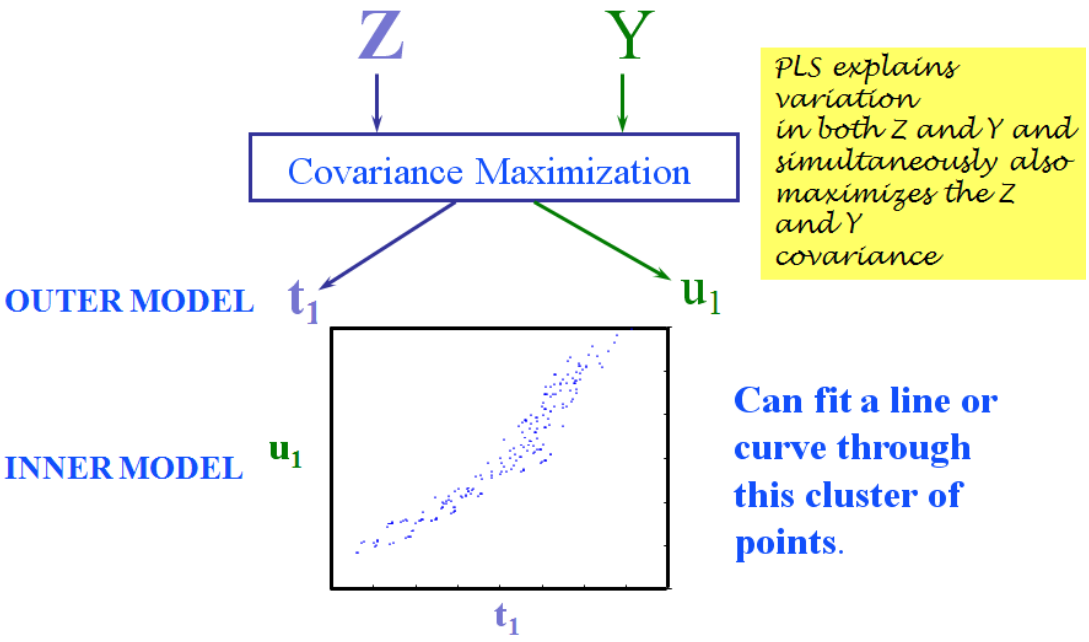

Figure 2.1: Graphical demonstration of the partial least squares regression (PLSR) algorithm

of the coefficient estimates and a significance level of $\alpha = 0.05$ will be used. In order to obtain samples of the entries of the connectivity matrix, the leave-one-out method described by Fukunaga and Hummels (1989) and Fukunaga and Hummels (1987) was used.

2.1.3 Akaike Information Criterion

Gene networks are highly sparse with most entries in the connectivity matrix being zero (Jeong et al., 2001; Tegner et al., 2003; Nacher and Ochiai, 2008; Hoguland et al., 2006). The Akaike information criterion (AIC) can be used to obtain further sparsity in the gene connectivity matrix. The AIC is an approach used for model selection and is widely accepted in various statistical model identification problems (Bozdogan, 1987; Yamaoka et al., 1978). This criterion has also been successfully applied in the literature to achieve sparsity in a gene connectivity matrix (Hoon et al., 2003; Ferrazzi et al., 2007; Cedersund and Roll, 2009; Chen et al., 2005). The

AIC is used to find an optimal tradeoff between accuracy and model complexity by penalizing both the modeling error and the number of parameters in the model.

Akaike (Akaike, 1974, 1981) gives the definition of AIC as follows:
AIC = (-2)log(maximum likelihood)+2(number of independently adjusted parameters within the model).

In this work, the model errors are assumed to be Gaussian and independent and identically-distributed random variables (i.i.d). Let m be the number of observations and $RSS = \sum_{i=1}^{m} \hat{\varepsilon}_i^2$ be residual sum of squares. Then Akaike information criterion (AIC) becomes:

$$AIC = 2n_p + m[log(\frac{2\pi RSS}{m}) + 1] \tag{2.9}$$

where n_p is the number of parameters. The Akaike information criterion (AIC) not only rewards the accuracy of fit based on residual sum of squares, but also penalizes number of parameters n_p. This penalty term avoids over-fitting by having a tradeoff between the goodness of fit with a parsimonious model. The preferred model is the one with the lowest AIC value.

For small sample size applications, the Akaike information criterion (AIC) does lead to biased estimate, which in turn leads to overfitting (Hurvich and Tsai, 1989). Therefore, a corrected AIC has been used in the current study based on the model suggested by McQuarrie and Tsai (1998). The corrected Akaike information criterion (AIC) is given by the equation:

$$AIC = 2n_p + mlog(\frac{RSS}{m}) + \frac{m + n_p}{m - n_p - 2} \tag{2.10}$$

2.2 Challenges

As mentioned in Section 2.1, microarray technology have enabled the gene expression profiles to be measured for thousands of genes, n, simultaneously. Also, the experimental cost for obtaining the time samples, m, for these thousands of genes are high. Therefore, the number of equations ($m \times n$) are fewer than the number of unknowns ($n \times n$). The system of ODEs needed to be solved are under-determined.

Without *apriori* information regarding the gene network, *no methodology* can give a unique and a true estimate for all the entries in the connectivity matrix, A. Typically, to identify the complete network model or connectivity matrix, the number of time samples in the data, m, must be at least equal to number of genes, n. Even with $m = n$ time samples, due to the presence of noise in the gene expression data matrix, it is not practically feasible to achieve a true estimate for all the entries of the connectivity matrix. In this work, a methodology is proposed to obtain a consistent estimate for some of the entries of the connectivity matrix.

As mentioned in Nacher and Ochiai (2008) and Hoguland et al. (2006), most of the elements in the connectivity matrix, A, are zero. An entry in the connectivity matrix, $\hat{A}$, can be estimated as zero by two means, namely, one by applying a particular methodology, and secondly the entry in connectivity matrix is assigned zero by default due of the lack of sufficient data. Not all the entries in the connectivity matrix are affected by the gene expression data matrix, X. Therefore, for a large and a highly sparse matrix, the percentage of entries estimated correctly and the percentage of entries obtained vary significantly. For example, consider a simulated case study with 500×500 connectivity matrix following the power law as mentioned in Nacher and Ochiai (2008) and Hoguland et al. (2006). For this simulated example, the number of non-zero entries in the connectivity matrix is 904. Choosing an estimate for the connectivity matrix, $\hat{A}$, with all the entries in the connectivity matrix as zero, the percentage error in the estimate $\hat{A}$ is 0.36%. Based on the percentage error, the estimate, $\hat{A}$, can be considered to be a very accurate one. This is a unique feature when dealing with sparse matrices, where a metric defining the number of errors is not a true indication of the usefulness of the methodology. In this work, the performance of the method is assessed based on both the correctly identified zero and non-zero coefficients. For validating the non-zero coefficients, only the sign of the coefficients are considered, whilst ignoring the magnitude.

2.3 Methods

2.3.1 Gene Connectivity Network Model

As stated in Section 2.1, for a system operating around steady state the gene connectivity matrix can be modeled with a set of linear ordinary differential equations (ODEs). Equation 2.3, can be re-written in the form:

$$\dot{x}_i(t) = \sum_{j=1}^{n} a_{ij} x_j(t) + \sum_{l=1}^{p} b_{il} u_l(t) \quad \forall t \in T = [t_1, t_m]. \tag{2.11}$$

where $i = 1, \ldots, n$ is the number of genes; $x_i(t)$ is the expression level of the i^{th}*gene* at time t; $\dot{x}_i(t)$ is the rate of change in the expression level of the i^{th}*gene* at time t; p is the number of genes perturbed in the system; a_{ij} is the influence of the j^{th} gene on the i^{th} gene; b_{il} is the l^{th} perturbation on the i^{th} gene and $u_l(t)$ is the l^{th} perturbation at time t (Bansal et al., 2006).

Equation 2.11, is rewritten in a matrix form as suggested in Equation 2.3

$$\dot{x}(t) = A_{n\times n} x(t) + B_{n\times p} u(t) \tag{2.12}$$

where $x(t)$ and $\dot{x}(t)$ are the expression level and the rate of change of expression level vectors for all n genes at time t, respectively; $u(t)$ is a $p \times 1$ vector containing the information regarding all perturbations at time t.

2.3.2 Algorithm

The continuous form of the Equation 2.12 needs to be discretized for analysis. However, exact discretization may sometimes be intractable due to the heavy matrix exponential and integral operations involved. It is much easier to calculate an approximate discrete model. Euler's approximation can be used to discretize a continuous system of equations to a discrete form. For a noisy data, however, taking a derivative by applying Euler's approximation will further increase the noise level.

Bilinear Transformation is one of the highly recommended methods for continuous to discrete transformation. Bilinear transformation applies the trapezoidal rule approximation which incorporates the higher-order integration procedure unlike the Euler's approximation. One of the advantages of bilinear transformation is that for any value of sampling time, the discrete time approximation to a stable continuous-time system is also stable. Since biological systems have a high time constant, the time step size chosen does not have an effect on the approximation (Ober and Montgomery-Smith, 1990; Mayhan, 1984).

$$x(t_{k+1}) = A_d x(t_k) + B_d u(t_k) \quad \forall\, k = 1,2,..m-1 \text{ time points} \tag{2.13}$$

where $x(t_k)$ is the noise-free or the signal component of the gene expression level measured for the n genes at a given time t_k. The noisy gene expression level measurement for the discrete case is defined as follows:

$$\tilde{x}(t_k) = x(t_k) + \xi(t_k) \quad \forall\, k = 1,2,..m \text{ time points} \tag{2.14}$$

where $\xi(t_k)$ is the noise component of the measured expression level, for the n genes, at time t_k. Equation 2.13 can be rewritten in a matrix form for all time points as:

$$Y = GA_d^T + U^T B_d^T \tag{2.15}$$

where Y is a transpose of the matrix having $x(t_2)$, $x(t_3)$, and so on till $x(t_m)$ as columns (i.e. $Y = [x(t_2)\ ..\ x(t_m)]^T$)and G is a transpose of the matrix with vectors $x(t_1)$, $x(t_2)$, and so on till $x(t_{m-1})$ as columns (i.e. $G = [x(t_1)\ ..\ x(t_{m-1})]^T$). Equation 2.15 is rewritten as follows:

$$Y = Z\beta \tag{2.16}$$

where $Z = \begin{bmatrix} G & U^T \end{bmatrix}$ and $\beta = \begin{bmatrix} A_d^T \\ \\ B_d^T \end{bmatrix}$.

Applying SIMPLS on the Z and Y matrices in Equation 2.16 and choosing the first k PLS components, the following solution is obtained

$$\beta_{pls} = RC^T = \begin{bmatrix} A_{d0}^T \\ \\ B_{d0}^T \end{bmatrix} \tag{2.17}$$

where R and C matrices are the weights of the Z matrix and loadings of the Y matrix calculated based on the algorithm suggested in (Jong, 1993), respectively; A_{d0} and B_{d0} are the solution obtained by applying partial least squares (PLS) on the Y and Z matrices in Equation 2.16.

The solution $\hat{A} = A_{d0}$ does not give a sparse estimate for the connectivity matrix. To obtain sparsity of $\hat{A}$ and to ensure that each connection is significant, the leave-one-out jackknifing and the AIC methods are applied sequentially. Firstly, the leave one out jackknifing method is applied to eliminate spurious connections. Secondly, the AIC method is applied to achieve further sparsity by finding a optimal tradeoff between accuracy and model complexity by penalizing both the modeling error and the number of parameters in the connectivity matrix..

The leave-one-out jackknifing (p-value hypothesis testing) is carried out on the entries of the $\hat{A}$ matrix, to eliminate spurious connections. The procedure for the leave-one-out jackknifing is as follows: For each time $t = t_1, t_2, \ldots, t_m$, the sample $x(t)$ is removed and the connectivity matrix is estimated using partial least squares (PLS) on the new Y and Z matrices in Equation 2.16. In this way a series of m samples are obtained for each of the entries in the connectivity matrix. A hypothesis test based on a t-distribution with $m-1$ degrees of freedom is then used to determine if each of the entries in the connectivity matrix are significant. As suggested in Section 2.1.2, a confidence level of $\alpha = 0.05$ is chosen for performing leave-one-out jackknifing.

The Akaike information criterion (AIC) method, as defined in Equation 2.10, is applied to the $\hat{A}$ matrix obtained after applying leave-one-out jackknifing to achieve further sparsity by finding a optimal tradeoff between accuracy and model complexity by penalizing both the modelling error and the number of parameters in the connectivity matrix. A series of steps in applying the AIC is listed as follows:

1. A nominal AIC score, $I_{\hat{A}}$, is computed for the model $\hat{A}$.
2. For each entry ij in $\hat{A}$, a new model $\hat{A}_{ij}$ is defined which is identical to $\hat{A}$ but the ij^{th} entry is zero.
3. For each of the new models the AIC score, $I_{\hat{A}_{ij}}$, is calculated .
4. The model with the lowest AIC score among the $\hat{A}_{ij}$ models is selected (i.e. $\hat{A}^{\star}_{ij} = \arg\min\{I_{\hat{A}_{ij}}\}$)
5. If $I_{\hat{A}^{\star}_{ij}} < I_{\hat{A}}$, then make $\hat{A} = \hat{A}^{\star}_{ij}$ and repeat steps 2 to 5
6. The procedure is terminated when no connection can be found whose elimination reduces the AIC score.

Let $A_d = \hat{A}$, be the final model obtained. The discretized form of final solution, A_d and B_{d0}, are transformed into continuous form, A and B, using inverse bilinear transformation suggested in Ober and Montgomery-Smith (1990).

$$A = \frac{2(A_d - I)}{\delta t(A_d + I)} \tag{2.18}$$

$$B = \frac{2}{\sqrt{\delta t}}(A_d + I)^{-1} B_{d0} \tag{2.19}$$

2.3.3 An Illustrative Example

Before highlighting the effectiveness of the proposed method on a real data set, the algorithm was applied on a simulation example. In this example, a set of 1000 random sparse gene networks of 10 genes are chosen. Each of these 1000 random networks, A, are chosen based on the following characteristics:

- Each network is represented by a full rank matrix with eigenvalues of the real part less than zero to ensure stability of dynamical systems (Bansal et al., 2006; Ljung, 1999).
- Each network follows a power-law distribution meeting the requirements of $P(k) \sim k^{-1.8}$ (Nacher and Ochiai, 2008; Hoguland et al., 2006).

For the network of 10 genes, the perturbation matrix, *B*, with a single perturbation, is chosen (p=1). The gene perturbed is chosen randomly and is stored in the $B_{n\times 1}$ matrix. Since only one gene is perturbed, the *B* matrix has all its entries except the one chosen randomly, equal to zero. The U, $1 \times m$, matrix is chosen with all the entries being constant and equal to 1.

2.3.4 Building a Simulated gene expression matrix

For each of the 1000 networks, a simulated expression matrix $X = \begin{bmatrix} x(t_1) & .. & x(t_m) \end{bmatrix}$ was obtained using the *lsim* command in MATLAB (Bansal et al., 2006) by solving Equation 2.12. The initial time t_1 is chosen to be zero and the end time t_m is chosen to be equal to 4 times the absolute value of the real part of the smallest eigen value of *A* (Ljung, 1999; Bansal et al., 2006; Gardner et al., 2003). For every gene expression matrix, X, five equally sampled time points (m = 5) are chosen. White Gaussian noise component is added to the X matrix with zero mean and varying standard deviations, from σ = 0.01*$||X||$ (1 % noise level) to 0.25*$||X||$ (25 % noise level) in increments of 0.01*$||X||$, where $||X||$ is the absolute values of entries of the gene expression matrix, X (Bansal et al., 2006; Gardner et al., 2003). In total, there are 1000 simulated noisy gene expression matrices for each of the 25 different noise components.

2.3.5 Comparitive study of the three methods

A comparative study is performed to assess the performance of the proposed method, by comparing it with the methods suggested in Yeung et al. (2002) and Bansal

et al. (2006). For the sake of simplicity and uniformity in comparing all three methods, the sparsity constraints in the proposed method based on leave-one-out jackknifing and the AIC were not applied to the recovered A_{d0} matrix, given in Equation 2.17. Instead, the recovered A_{d0} matrix, in Equation 2.17, is directly transformed using bilinear transformation, mentioned in Equation 2.18, to obtain the A matrix. The resultant, A, matrix is compared to the corresponding connectivity matrices obtained using methods suggested in Yeung et al. (2002) and Bansal et al. (2006).

The network sparsity for each of these methods was achieved based on the method proposed in Bansal et al. (2006). In the method suggested by Bansal et al. (2006), for the purpose of obtaining sparsity, the smallest h entries in recovered network, $\hat{A}$, are set to zero. The variable h is defined such that it varies from zero to total number of entries in connectivity matrix (in this case $10\times$ 10 =100) (Bansal et al., 2006). For every smallest $h \in \{0,1,...,100\}$ entries set to zero, a corresponding connectivity matrix, A, is obtained.

The performance of the algorithm proposed in this work along with the algorithms proposed in Yeung et al. (2002) and Bansal et al. (2006) are assessed based on the correctly identified zero and non-zero coefficients (based on only the sign of the coefficients) in the A matrix. For this purpose, two ratios r_z and r_{nz} are introduced as suggested in Bansal et al. (2006):

$$r_z = \frac{\text{Identified correct zero coefficients}}{\text{Total number of zero coefficients}} \tag{2.20}$$

$$r_{nz} = \frac{\text{Identified correct non-zero coefficients with agreeing sign}}{\text{Total number of non-zero coefficients}} \tag{2.21}$$

An average r_{nz} versus the r_z curve, across 1000 networks, is plotted for all three methods and a comparison is made. The curve which ensures a maximum area under the r_{nz} versus the r_z curve is considered the best method (Bansal et al., 2006).

2.3.6 Robustness of the proposed method to noise

To highlight the robustness of the proposed method to measurement noise, a single 10 gene system is chosen based on the characteristics stated in Section 2.3.3. White Gaussian noise components with zero mean and standard deviation equal to 0.25*$||X||$ (25 % noise level) is added, at 100 different times, to the simulated X matrix, in Monte-Carlo fashion. A set of 100 different X matrices are obtained, one for each noise component. For each noise component, a connectivity matrix A is estimated using the proposed method (including the sparsity constraint proposed by the method). The variance of the entries in recovered A, across the 100 different noise components, are calculated. The method which gives lower values for the variances of the entries is considered a better method, because it ensures the consistency in the estimates.

2.3.7 Experimental Data

The algorithm was applied to a nine-transcript subnetwork of the SOS pathway in *E.coli*. The total RNA was extracted at 6 time points: 0, 12, 24, 36, 48, and 60 min. Each experiment was done in triplicate and an average expression is chosen at all time points. The noise level in the experiment was found to be approximately around 13 % (refer Bansal et al. (2006) for experimental description and the noise in the experimental data).

Table 2.1 gives a list of the 9 genes in the SOS network along with average expression levels at different times.

2.4 Results and Discussion

2.4.1 10 gene simulated networks

Each of the 1000 recovered networks, A, are made sparse by setting the smallest h absolute values of $\hat{A}$ matrix equal to zero. The two ratios r_z and r_{nz}, suggested

Genes	0 min	12 mins	24 mins	36 mins	48 mins	60 mins
recA	0	3.4555	3.7139	3.5245	3.3526	3.4996
lexA	0	0.7193	1.0782	1.0783	0.8543	0.8787
Ssb	0	0.599	1.1959	0.8905	0.4406	0.425
recF	0	1.4377	0.7241	0.2964	-0.0114	0.1034
dinI	0	2.1853	3.3187	3.3862	3.2019	3.2664
umuDC	0	0.4214	1.0584	0.9315	0.8259	1.0371
rpoD	0	1.8529	1.3839	0.4021	-0.0522	-0.1174
rpoH	0	0.1713	-0.2225	-0.65	-0.9738	-0.7261
rpoS	0	-0.5088	-0.3991 -	1.0944	-1.7731	-1.4595

Table 2.1: The gene expression data of the 9 gene SOS subnetwork.

in Equations 2.20 and 2.21 respectively, are calculated by varying the value of h from zero to the total number of entries in A. The best value of h is the value when all the connections (positive, negative and zero) are identified correctly with no false negatives or positives (i.e. $r_z = r_{nz} = 1$). For a noisy under-determined system, estimating all the connections accurately, without any *apriori* information, is not feasible. Therefore, the method which ensures a maximum area under the r_{nz} versus the r_z curve is considered as the better method (Bansal et al., 2006).

The area under the average r_{nz} versus the r_z curve, across the 1000 random networks versus different noise levels is plotted. Figure 2.2 shows the average area under the r_{nz} versus the r_z curve versus noise level for the proposed method. The plot indicates that for low noise (noise level less than 5 %), choosing three PLS components gives the best estimate for the connectivity matrix and at higher noise level, choosing two PLS components is a better option. Based on the area under the curve value in Figure 2.2, two PLS components are chosen for the estimation of the connectivity matrices using the proposed methodology.

An average r_{nz} versus the r_z plot comparing the three methods is presented in this chapter. For the comparison, three principal components are chosen for estimating the network using the method proposed in Bansal et al. (2006). As suggested in Section 2.3.7, the noise level of the real data is approximately around 13 %. Hence, a plot comparing the average r_{nz} versus the r_z across 1000 random networks, for a noise level of 13 % is presented in Figure 2.3,

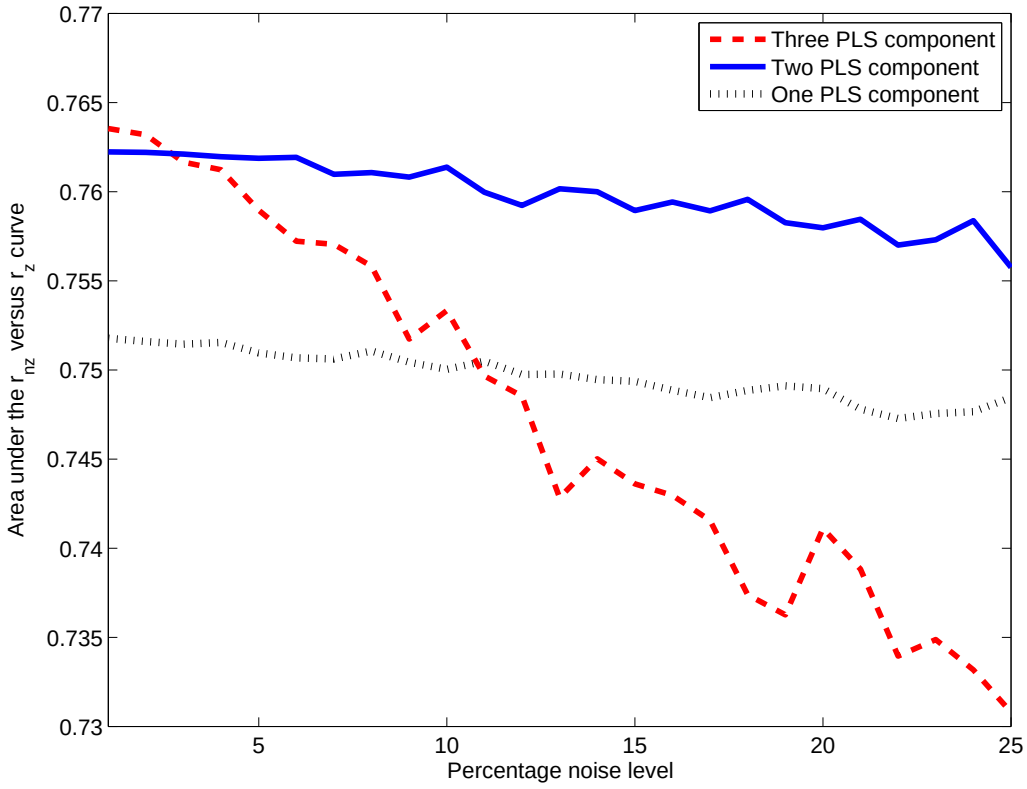

Figure 2.2: Plot of the area under the average r_{nz} (true non-zeros) versus the r_z (true zeros) curve across 1000 random networks, versus the percentage noise levels for the proposed method.

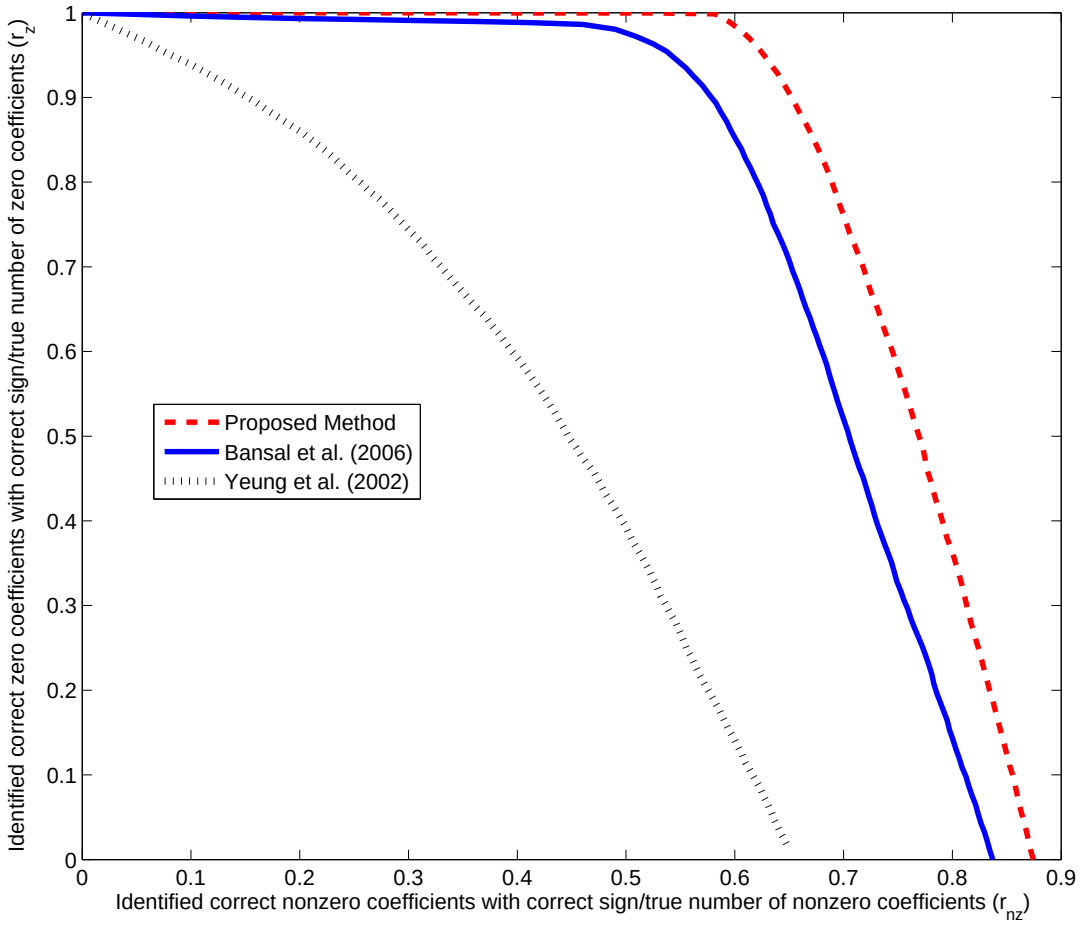

Figure 2.3: Plot of the average r_{nz} (true non-zeros) versus the r_z (true zeros) curve across 1000 random networks, at a noise level of 13 %, for all the three methods. Subscript 1, 2 and 3 indicate the proposed method, method in Bansal et al. (2006), and method in Yeung et al. (2002) respectively.

The area under the average r_{nz} versus the r_z curve for the proposed method is higher than the area under the curve for the methods proposed in (Bansal et al., 2006) and (Yeung et al., 2002). Based on this result, the performance of the method proposed is better compared to the performance of the other two methods.

2.4.2 Advantages of PLSR over PCR

The advantage of the PLSR method, used in the study, compared to PCR method, used in Bansal et al. (2006), is illustrated with the help of an simulated example. A set of 5000 different sparse gene networks of 10 genes are chosen, based on the characteristics mentioned in Section 2.3.3:

- Each network is represented by a full rank matrix with eigenvalues of the real part less than zero to ensure stability of dynamical systems (Bansal et al., 2006; Ljung, 1999).
- Each network follows a power-law distribution meeting the requirements of $P(k) \sim k^{-1.8}$ (Nacher and Ochiai, 2008; Hoguland et al., 2006).

For each of the 5000 sparse networks, an expression matrix $X = \begin{bmatrix} x(t_1) & .. & x(t_m) \end{bmatrix}$ was obtained using the *lsim* command in MATLAB (Bansal et al., 2006) by solving Equation 2.12. The initial time t_1 is chosen to be zero and the end time t_m is chosen to be equal to 4 times the absolute value of the real part of the smallest eigen value of A (Ljung, 1999; Bansal et al., 2006; Gardner et al., 2003). For every gene expression matrix, X, five equally sampled time points (m = 5) are chosen. White Gaussian noise component is added to the X matrix with zero mean with varying standard deviations, from σ = 0.01*$||X||$ (1 % noise level) to 0.25*$||X||$ (25 % noise level) in increments of 0.01*$||X||$, where $||X||$ is the absolute values of entries of the gene expression matrix, X (Bansal et al., 2006; Gardner et al., 2003). In total, there are 5000 simulated noisy gene expression matrices for each of the 25 different noise components.

For each of the 5000 recovered networks, A, the two ratios, r_z and r_{nz}, suggested in Equations 2.20 and 2.21 respectively, are calculated by varying the value of h

from zero to the total number of entries in A. The best value of h is the value when all the connections (positive, negative and zero) are identified correctly with no false negatives or positives (i.e. $r_z = r_{nz} = 1$). For a noisy under-determined system, estimating all the connections accurately, without any *apriori* information, is not feasible. Therefore, the method which provides a larger area, under the r_{nz} versus the r_z curve, compared to the other methods is considered superior (Bansal et al., 2006).

The area under the r_{nz} versus the r_z curve, for the methods proposed in this work and method suggested in Bansal et al. (2006), are compared. Since the PLSR method, used in this study, is compared with the PCR method, used in Bansal et al. (2006), the sparsity constraints in the proposed method based on leave-one-out jackknifing and the AIC were not applied to the recovered A_{d0} matrix, given in Equation 2.17. Instead, the recovered A_{d0} matrix is directly transformed using bilinear transformation, mentioned in Equation 2.18, to obtain the A matrix.

In Figure 2.4, a histogram plot of the difference in the area under the r_{nz} versus the r_z curve (at 13% noise level), between methods proposed in this work and (Bansal et al., 2006), for all the 5000 recovered networks, is plotted. The histogram shows that with approximately 78% confidence, the method proposed in this work gives higher area under the r_{nz} versus the r_z curve compared to the method proposed in Bansal et al. (2006). This can be used as a conclusion to suggest that the PLSR method, used in this study, gives a better estimate for the connectivity matrix over PCR method, used in Bansal et al. (2006).

The confidence level in obtaining a higher area under the r_{nz} versus r_z curve using the PLSR method compared to PCR method, across various noise levels, is plotted in Figure 2.5. It can seen from the figure that PLSR method consistently outperforms the PCR method for all chosen noise levels. This can be used as a conclusion to suggest that the PLSR method, used in this study, gives a better estimate for the connectivity matrix over PCR method, used in Bansal et al. (2006).

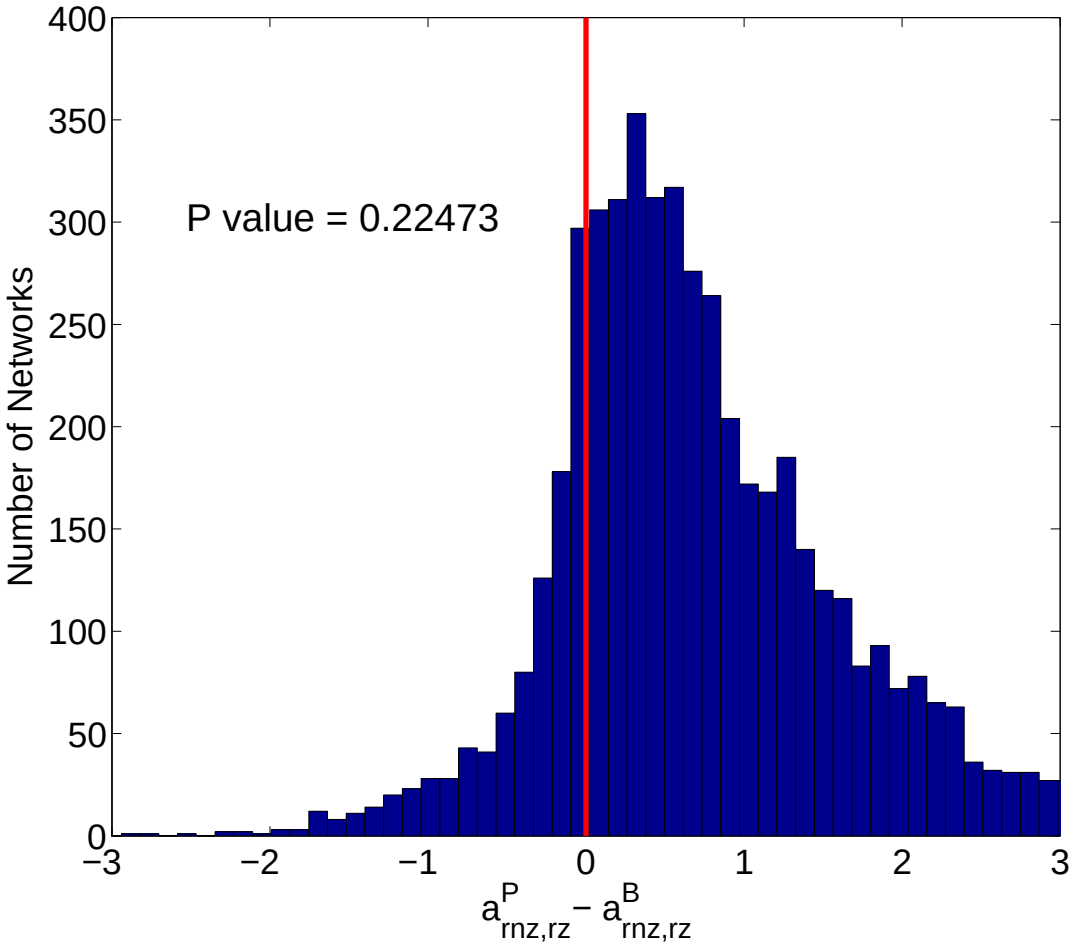

Figure 2.4: Comparative performance of the proposed method with respect to the method in Bansal et al. (2006), for the 5000 simulated gene networks. The histogram shows the distribution of the difference in area $a^{P}_{rnz,rz} - a^{B}_{rnz,rz}$.$a^{P}_{rnz,rz}$, $a^{B}_{rnz,rz}$ are the areas under r_{nz} (true non-zeros) curve versus the r_z (true zeros) for the proposed method and the method in Bansal et al. (2006), respectively, at a noise level of 13 %, for the 5000 simulated networks, , as shown in Figure 2.3. Each bin corresponds to the number of networks obtained with the similar differences in the area between the dashed (proposed method) and solid (Bansal et al. (2006)'s method) curves in Figure 2.3

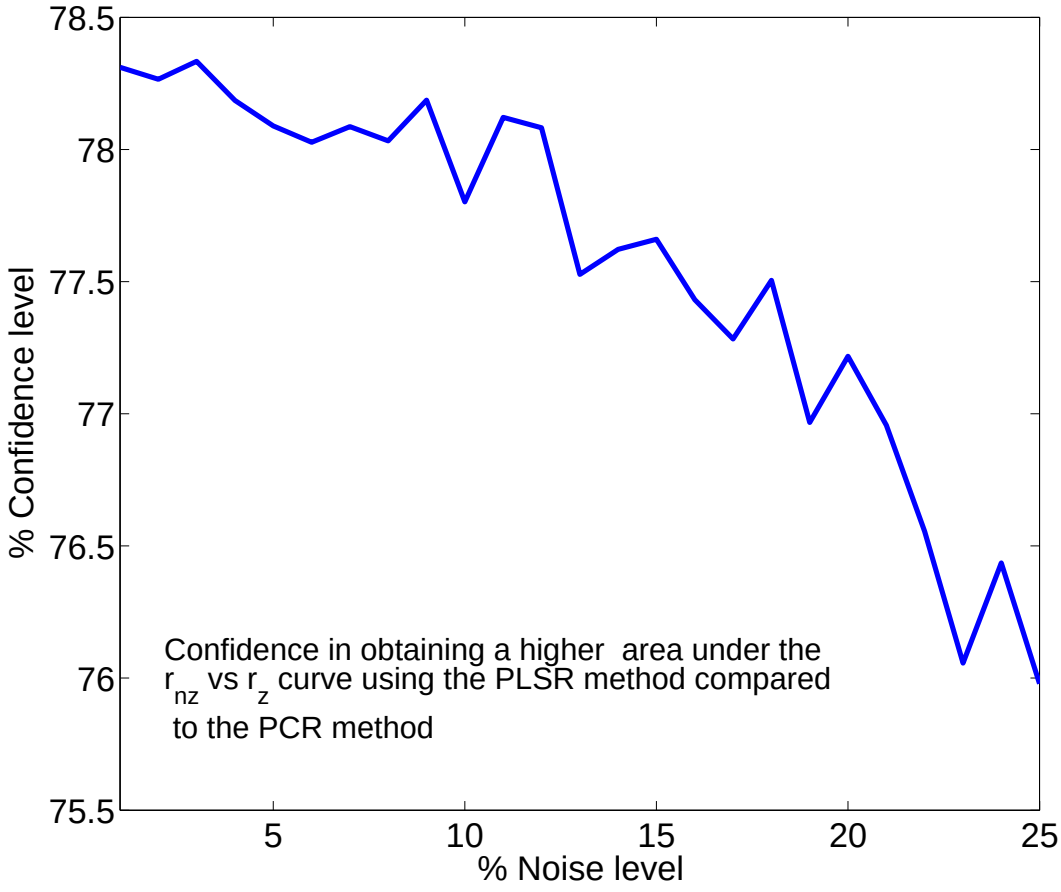

Figure 2.5: Comparing the performance of PLSR method, in this study, to the PCR method, in (Bansal et al., 2006), across different noise levels. The percentage confidence of obtaining a higher area under the curve distribution, for the 5000 simulated gene networks, for the PLSR method compared to PCR method across various noise levels, is shown in the plot.

2.4.3 Advantages of applying both leave-one-out jackknifing and the AIC methods

In the current study, both leave-one-out jackknifing and the AIC methods are used for obtaining a sparse estimate of the gene connectivity matrix. The advantage of using both the methods for obtaining an estimate is emphasized with the help of 5000 simulated noisy gene expression matrices at each of the 25 different noise levels, as mentioned in Section 2.4.2.

As proposed in Section 2.2, to obtain a sparse estimate of the gene connectivity matrix, leave-one-out jackknifing is applied first and then the AIC method is applied. Applying the sparse estimate using the proposed approach is compared with the sparse estimate obtained by applying the leave-one-out jackknifing method alone, the AIC method alone, and first the AIC methodology and second leave-one-out jackknifing. The comparative study involves enforcing the sparsity constraints to the recovered A_{d0} matrix, in 2.17 by applying the four different procedures. For each of the cases, the resultant matrix, A_d, is transformed using bilinear transformation, suggested in 2.18, to obtain the A matrix.

As in Section 2.4.2, for each of the 5000 recovered networks, A, the two ratios, r_z and r_{nz}, for the sparse network obtained are calculated. The best value of the two ratios is when all the connections are correctly identified (i.e. r_z =r_{nz}=1). An average of the two ratios, r_z and r_{nz}, across the 5000 recovered networks is calculated. For a noisy under-determined system, estimating all the connections accurately, without any *apriori* information, is not feasible. Therefore, the technique which has the $\{r_{nz}, r_z\}$ point in the average r_{nz} versus the r_z curve closer to the point (smaller distance) $\{1,1\}$ is considered as the better method.

To this end, a distance metric, $d_{rnz,rz}$, is defined which calculates the distance of the $\{r_{nz}, r_z\}$ point in the average r_{nz} versus the r_z curve to the point $\{1,1\}$. The distance metric is defined as follows:

$$d_{rnz,rz} = \sqrt{(1-rnz)^2 + (1-rz)^2}$$

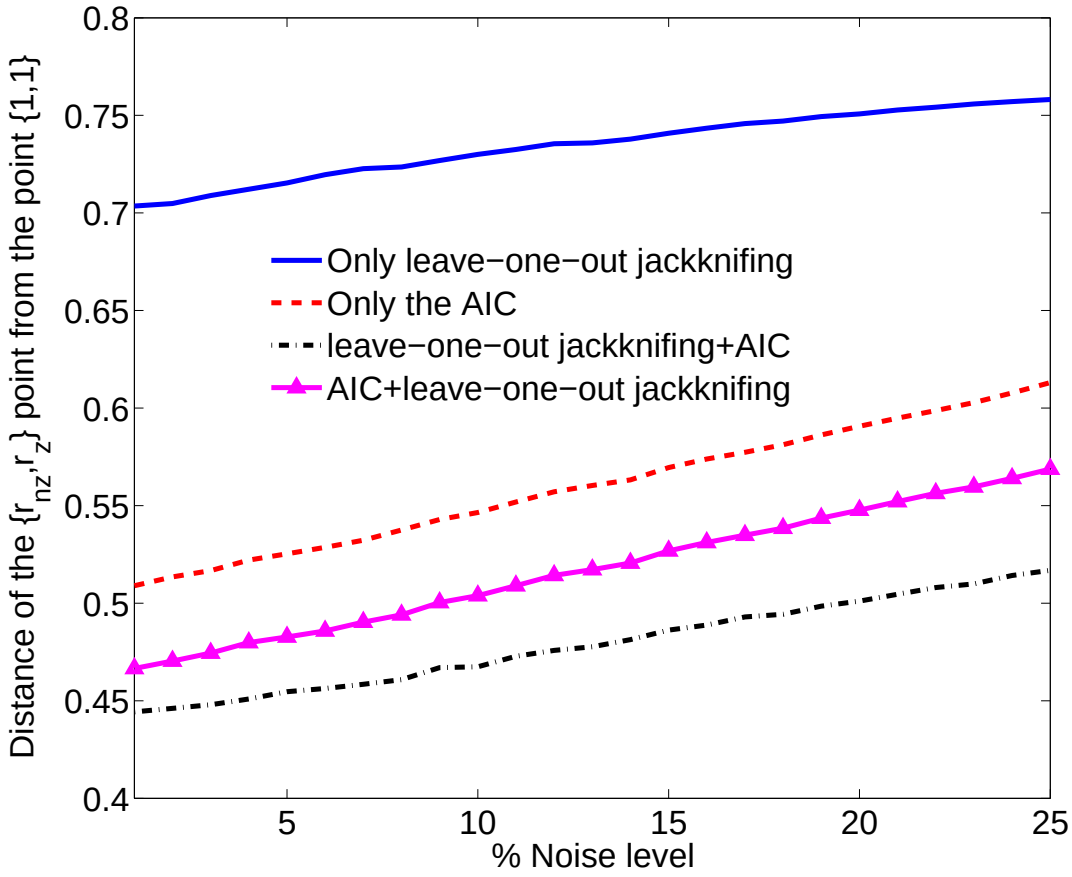

Figure 2.6: Comparing the distance of the point on the average r_{nz} (true non-zeros) versus the r_z (true zeros) curve applying only leave-one-out jackknifing method, only the AIC method, leave-one-out jackknifing+AIC methodologies, and the AIC+leave-one-out jackknifing at different noise levels.

Based on the smaller value of the distance metric, $(d_{rnz,rz})$ in Figure 2.6, the sparsity constraint enforced by applying first the leave-one-out jackknifing method and then the AIC method is the ideal combination. The method of applying leave-one-out jackknifing (LOOJ) before AIC is also justified based on the fact that applying LOOJ first removes the spurious connections obtained using PLSR method and gives a robust connectivity matrix and further applying the AIC method reduces the complexity of the obtained matrix. Obtaining a robust model prior to reducing the model complexity is a more judicious approach.

2.4.4 Analysis of Noise-Robustness Via Monte Carlo Simulations

Since, microarray data are highly noisy, the consistency of the entries in the connectivity (recovered) matrix in presence of measurement noise is a necessary requirement. The method which shows a higher confidence for the entries is indicative of a better performance. Since the noise level is 25 %, two PLS components are chosen for the proposed method and two principal components (PCs) are chosen for the method proposed in Bansal et al. (2006).

A histogram of the variances of the entries, across the 100 Monte-Carlo samples, is also plotted. Figure 2.7 shows a histogram plots of the variance of the entries in A for the proposed method and the methods in Yeung et al. (2002) and Bansal et al. (2006). As can be seen from Figure 2.7, the variances of the entries obtained by the proposed method is smaller than variances of the entries using the methods proposed in Bansal et al. (2006) and Yeung et al. (2002). From the plots one of the important observation is that the variances of some of the entries using method proposed in Yeung et al. (2002) are significantly higher and hence the confidence on the estimates are very poor.

2.4.5 Nine Gene SOS Network

For the nine gene SOS dataset in Table 2.1, the algorithm proposed, in this study, is applied and the network obtained is shown in Table 2.2. Since the noise level in

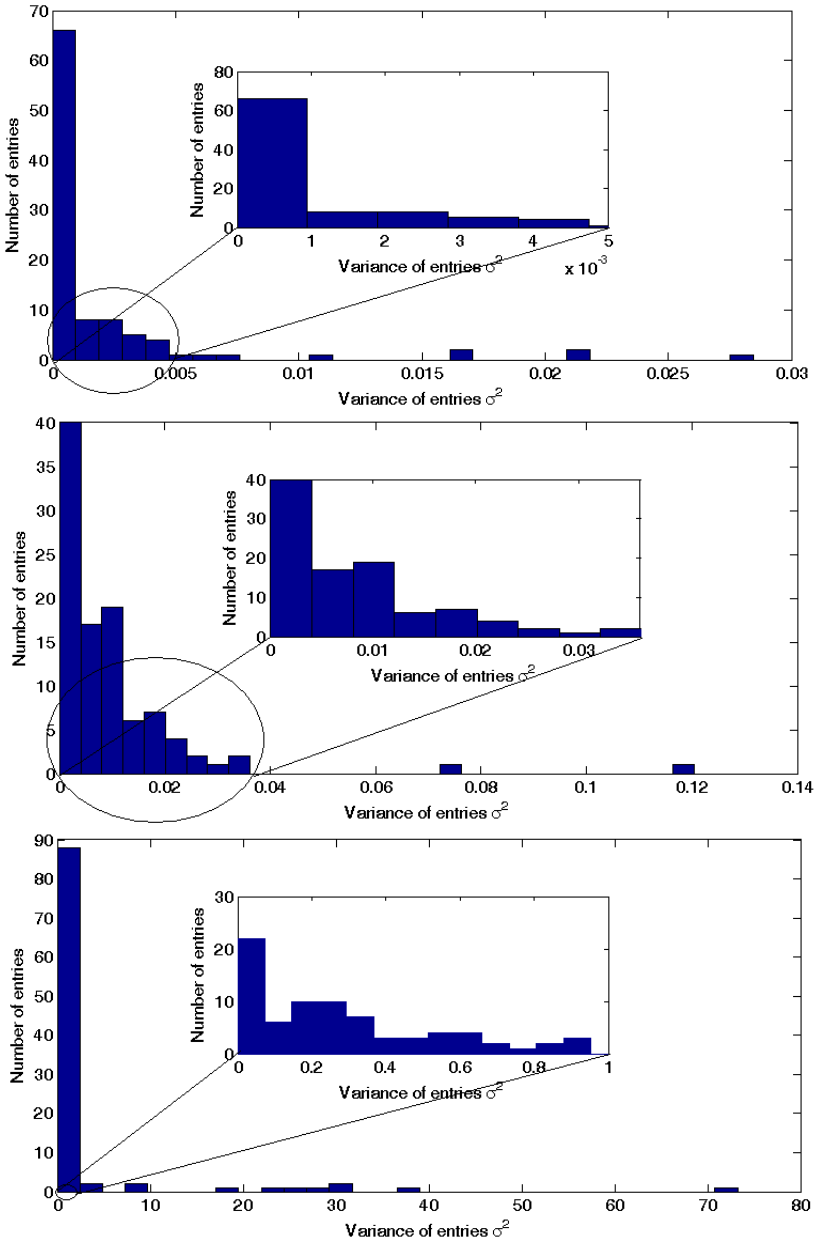

Figure 2.7: Histograms of the variance of the entries of the recovered connectivity matrices estimated by the proposed method (first), by method proposed in Bansal et al. (2006) (second) and the method proposed in Yeung et al. (2002) (third). The initial values of the histograms are zoomed and presented in the inset of the plot. Note the different scale in these plots.

the real data is around 13%, three PLS components are chosen for analysis. A 95% confidence level is chosen for obtaining sparsity using leave-one-out jackknifing (α= 0.05).

The inferred network is compared with known interactions given in the literature. There were 43 proposed connections, apart from the self feedback, between these 9 genes (Bansal et al., 2006) . For estimating the final gene network, no *apriori* information regarding the number of connections per gene is used. Table 2.2 gives the final gene network estimated using the method proposed in this work.

Table 2.2: The nine gene SOS network recovered using the proposed methodology. The connectivity matrix values are rounded off to two decimal places.

	recA	lexA	Ssb	recF	dinI	umuDC	rpoD	rpoH	rpoS
recA	-0.83	0	0.02	1.06	0.36	0.09	0.76	0	0
lexA	0.32	-2.00	0.01	0.30	0.10	0.03	0.21	0	0
Ssb	-0.07	-0.04	-1.99	0.66	0.22	0.06	0.46	0	0
recF	0.11	0	-0.09	-1.93	-0.39	0.10	0.78	0	0
dinI	1.12	0	0.01	0.61	-1.79	0.06	0.44	0	0
umuDC	0.33	0	0	0.14	0.05	-1.99	0.11	0	0
rpoD	0.23	0	0	0.12	-0.69	-0.33	-0.69	0	0
rpoH	-0.08	0	0	0.17	-0.48	0.04	0.36	-2.00	0
rpoS	0.26	0	0	0.14	-0.79	-0.27	0.10	0	-1.63

For comparing the original and the recovered networks, only the signs of the entries are taken into account whilst ignoring the magnitude. Therefore, the network given in Table 2.2 is converted to a sign network given in Table 2.3 for the purpose of comparison. Tables 2.3 and 2.4, show the recovered gene network (only signs) using the algorithm suggested in this study and the network proposed in the literature, respectively.

As many as 25 of the 43 proposed connections were correctly identified as compared to the 20 connections obtained using the method proposed in Bansal et al. (2006). For obtaining a sparse matrix, the method proposed in (Bansal et al., 2006) used the information that each gene is connected to five other genes (based on the work proposed in Gardner et al. (2003)). Using the proposed method, we were able to achieve as many as 25 connections correctly without using the *apriori*

information regarding the connectivity of the genes, suggested in Gardner et al. (2003). The method proposed was also able to identify 19 true zero coefficients in the network compared to the 17 true zero coefficients obtained using the method proposed in Bansal et al. (2006).

Table 2.3: The recovered SOS network (only signs) using the proposed methodology

	recA	lexA	Ssb	recF	dinI	umuDC	rpoD	rpoH	rpoS
recA	-1	0	1	1	1	1	1	0	0
lexA	1	-1	1	1	1	1	1	0	0
Ssb	-1	-1	-1	1	1	1	1	0	0
redF	1	0	-1	-1	-1	1	1	0	0
dinI	1	0	1	1	-1	1	1	0	0
umuDc	1	0	0	1	1	-1	1	0	0
rpoD	1	0	0	1	-1	-1	-1	0	0
rpoH	-1	0	0	1	-1	1	1	-1	0
rpoS	1	0	0	1	-1	-1	1	0	-1

Table 2.4: The original nine gene SOS network as proposed in the (Bansal et al., 2006). The values 1, -1, and 0 indicate a positive connection, negative connection, and a lack of connection respectively

	recA	lexA	Ssb	recF	dinI	umuDC	rpoD	rpoH	rpoS
recA		-1	-1	1	1	-1	1	0	0
lexA	1		-1	1	1	-1	1	0	0
Ssb	1	-1		1	1	-1	1	0	0
recF	0	0	-1		0	-1	1	0	1
dinI	1	-1	-1	1		-1	1	0	0
umuDc	1	-1	-1	1	1		1	0	0
rpoD	1	-1	-1	1	1	-1		1	0
rpoH	0	0	0	0	0	0	1		0
rpoS	0	0	0	0	0	0	1	0	

To emphasize the advantage of using first, the leave-one-out jackknifing method and then the AIC method, as suggested in the proposed method, the number of connections obtained using the proposed method (25 of the 43 proposed connections) is compared with the number of connections obtained by applying only leave-one-out jackknifing method (31 of the 43 proposed connections), applying only the AIC method (16 of the 43 proposed connections), and applying first the AIC methodology and second leave-one-out jackknifing (16 of the 43 proposed connections), as sparsity constraints.

Although, the number of connections correctly identified, by using only leave-one-out jackknifing, increased from 25 to 31, the number of true zero coefficients identified in the network decreased from 19 to zero. Therefore, due to the presence of large number of false positives, the method of obtaining sparsity using leave-one-out jackknifing alone is not preferred.

2.5 Advantages of the proposed method to the method based on (Varah, 1982)

For the purpose of applying the procedure suggested in Varah (1982), the gene expression levels, $x(t)$, is sampled at time $t_j = \{t_1 < t_2 < .. < t_m\}$ with $t_j \in T$, and is written in the form of a gene expression matrix, $X_{n\times m}$, with rows indicating the various genes and columns indicating different time samples. That is, each cell in the gene expression matrix represent expression level of that particular gene at a given time (refer Section 1).

Equation 9, is written for all m samples in the matrix form as follows:

$$\dot{X}_{n\times m} = A_{n\times n}X_{n\times m} + B_{n\times p}U_{p\times n} \tag{2.22}$$

An estimate for both the original matrix X and its first derivative matrix $\dot{X}$, in Equation 2.22, are obtained by applying a uniform cubic b-spline least squares method (Varah (1982); Deng et al. (2009)). Equation 2.22, can be written analogous

to the Equation 2.15 as follows:

$$Y = GA^T + U^T B^T \tag{2.23}$$

where Y is a transpose of the $\dot{X}$ matrix and G is a transpose of the X matrix. Equation 2.23 is rewritten as follows:

$$Y = ZH \tag{2.24}$$

where $Z = \begin{bmatrix} G & U^T \end{bmatrix}$ and $H = \begin{bmatrix} A^T \\ \\ B^T \end{bmatrix}$.

Applying SIMPLS on the Z and Y matrices in Equation 2.16 and choosing the first k PLS components, the following solution is obtained

$$H_{pls} = RC^T = \begin{bmatrix} A^T \\ \\ B^T \end{bmatrix} \tag{2.25}$$

where R and C matrices are the weights of the Z matrix and loadings of the Y matrix calculated based on the algorithm suggested in (Jong, 1993), respectively; A and B are the solution obtained by applying partial least squares (PLS) on the Y and Z matrices in Equation 2.24.

The simulated case study of 5000 gene networks, suggested in Section 3.2, is used to recover the connectivity matrix using the method proposed in this section. For each of the 5000 recovered networks, A, the two ratios, r_z and r_{nz}, are calculated by varying the value of h from zero to the total number of entries in A.

The area under the r_{nz} versus the r_z curve, for the method proposed in this work and method using the procedure in Varah (1982), are compared. Figure 2.8, shows a histogram plot of the difference in the area under the r_{nz} versus the r_z curve, between

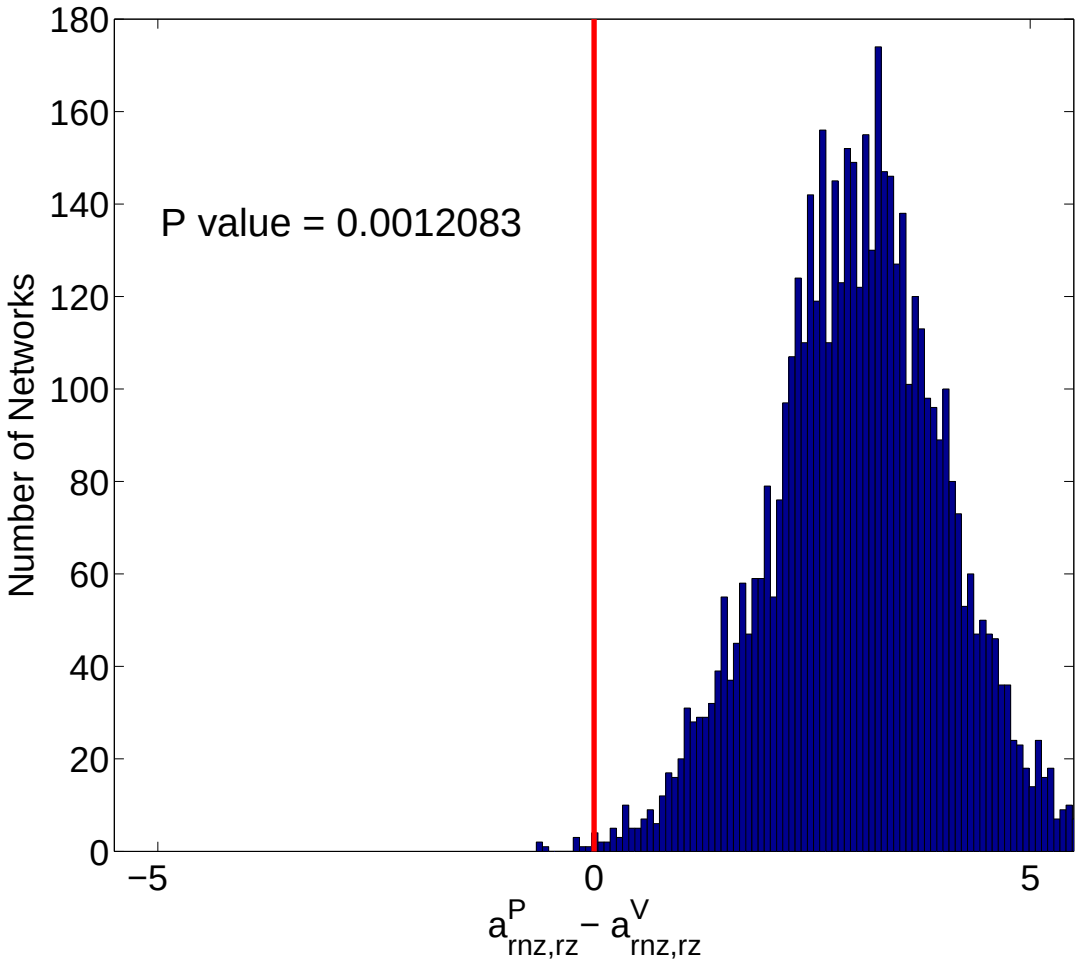

Figure 2.8: Comparative performance of the proposed method with respect to the method using the procedure in Varah (1982), for the 5000 simulated gene networks. The histogram shows the distribution of the difference in area $a^P_{rnz,rz} - a^V_{rnz,rz}$. $a^P_{rnz,rz}$, $a^V_{rnz,rz}$ are the areas under r_{nz} (true non-zeros) versus the r_z (true zeros) for the proposed method and the method based on the procedure in Varah (1982), respectively, at a noise level of 13 %, for the 5000 simulated networks.

method proposed in this work and the method using the procedure in Varah (1982), for all the 5000 recovered networks. The histogram shows that with more than 99% confidence, the method proposed in this work gives higher area under the r_{nz} versus the r_z curve compared to the method proposed using the procedure in Varah (1982). This can be used as a conclusion to suggest that the method proposed in this study, gives a better estimate for the connectivity matrix over the method using the procedure in Varah (1982), for smaller networks.

Also, for large networks, the estimate of the connectivity matrix obtained using the procedure in Varah (1982) did not yield a better result compared to the method proposed in this work. Therefore, it can be concluded that the method proposed in this work shows a superior performance compared to the method using the procedure in Varah (1982).

2.6 Limitations

As mentioned in Section 2.1, microarray technology have enabled the gene expression profiles to be measured for thousands of genes, n, simultaneously. Also, the experimental cost for obtaining the time samples, m, for these thousands of genes are high. Therefore, the number of equations ($m \times n$) are fewer than the number of unknowns ($n \times n$). The system of ODEs needed to be solved are under-determined. For a large gene network with very few time samples, the proposed method does not yield satisfactory results. This is due to the highly under-determined nature of the system which require more than one set of experiments for obtaining a satisfactory result.

2.7 Concluding remarks

In this study, three objectives are achieved, firstly, a novel algorithm is proposed for obtaining a statistically significant estimate of the gene network from linear ODEs using a combination of well known statistical tools such as partial least squares (PLS), leave-one-out jackknifing and the Akaike information criterion (AIC). The method uses the knowledge of bilinear transformations for discretizing a linear ODE problem into a linear algebraic problem.

Secondly, a comparative study performed with a simulated gene network, illustrated the superior performance of the method as compared to methods available in the literature. The simulated gene network is built so that it closely resembles a real gene network (i.e. a stable network with gene connectivity satisfying a power law distribution). The obtained estimates were consistent and robust to measurement noise in the data.

Finally, the method applied on experimental data for a nine-gene SOS network was able to successfully extract 25 out of 43 proposed connections in the literature without any *apriori* knowledge on the network.

References

L. M. Erickson, F. Pan, A. Ebbs, M. Kobayashi, H. Jiang, Microarray-based gene expression profiles of allograft rejection and immunosupression in the rat heart transplantation model, Transplantation 76 (2003) 582–588.

X. J. Ma, S. Dahiya, E. Richardson, M. Erlander, D. C. Sgroi, Gene expression profiling of the tumor microenvironment during breast cancer progression, Breast Cancer Research 11 (2009).

Y. Tu, G. Stolovitzky, U. Klien, Qualitative noise analysis for gene expression microarray experiments, PNAS 99 (2002) 14031–14036.

M. Stegall, W. park, W. Kremers, Gene expression during acute allograft rejection: Novel statistical analysis of micorarray data, American Journal of Transplantation 2 (2002) 913–925.

K. E. Bethin, Y. Nagai, R. Sladek, M. Asada, Y. Sadovsky, T. J. Hudson, L. J. Muglia, Microarray analysis of uterine gene expression in mouse and human pregnancy, Molecular Endocrinology 17 (2003) 1454–1469.

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

O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. H. R. Tibshirani, D. Botstein, R. B. Altman, Missing value estimation methods for DNA microarrays, Bioinformatics 17 (2001) 520–525.

M. K. Yeung, T. J, C. J. J, Reverse engineering gene networks using singular value decomposition and robust regression, PNAS 99 (2002) 6163–6168.

T. S. Gardner, D. di Bernardo, D. Lorenz, Inferring genetic networks and identifying compound mode of action via expression profiling, Science 301 (2003).

T. F. Liu, W. K. Sung, A. Mittal, Model gene network by semi-fixed bayesian network, Expert Systems with Applications 30 (2006) 42 – 49.

E. E. Schadt, P. Y. Lum, Reverse engineering of genetic networks to identify key drivers of complex disease, Journal of Lipid Research 47 (2006) 2601–2613.

K. Basso, A. A. Margolin, G. Stolovitzky, U. Klien, R. Dalla-Favera, A. Califano, Reverse engineering of regulatory networks in human b cells, Nature Genetics 37 (2005) 382–390.

A. J. Hartemink, Reverse engineering gene regulatory networks, Nature Biotechnology 23 (2005) 554 – 555.

P. D'haseseleer, S. Liang, R. Somogyi, Genetic network inference: From co-expression clustering to reverse engineering, Bioinformatics 16 (2000) 707–726.

J. Tegner, M. K. Yeung, J. Hasty, J. J. Collins, Reverse engineering gene networks: Integrating genetic perturbations with dynamic modeling, PNAS 100 (2003) 5944–5949.

P. Brazhnik, A. de la Fuente, P. Mendes, Gene networks: how to put the function in genomics, Trends in Biotechnology 20 (2002) 467–472.

P. M. Magwene, J. Kim, Estimating genomic coexpression networks using first-order conditional independence, Genome Biology 5 (2004) R100.

M. Zou, S. D. Conzen, A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data, Bioinformatics 21 (2005) 71–79.

F. He, R. Balling, A.-P. Zeng, Reverse engineering and verification of gene networks: Principles, assumptions, and limitations of present methods and futu, Journal of Biotechnology 144 (2009) 190–203.

M. Hecker, S. Lambeck, S. Toepfer, E. van Someren, R. Guthke, Gene regulatory network inference: Data integration in dynamic models–a review, Biosystems 96 (2009) 86–103.

M. Bansal, V. Belcastro, A. A. Impiombato, D. di Bernardo, How to infer gene networks from expression profiles, Molecular Systems Biology 3 (2007).

S. Liang, S. Fuhrman, R. Somogyi, Reveal, a general reverse engineering algorithm for inference of genetic network architectures, Proceedings of the Pacific Symposium on Biocomputing (Singapore) (R. Altman, A. Dunker, L. Hunter, and T. Klien,eds.), World Scientific Press 3 (1998) 18–29.

M. Bansal, G. D. Gatta, D. di Bernardo, Inference of gene regulatory networks and compound mode of action from time course gene expression profiles, Bioinformatics 22 (2006) 815–822.

J. C. Liao, R. Boscolo, Y.-L. Yang, L. M. Tran, C. Sabatti, V. P. Roychowdhury, Network component analysis: Reconstruction of regulatory signals in biological systems, Proceedings of the National Academy of Sciences of the United States of America 100 (2003) 15522–15527.

P. Foteinou, E. Yang, G. Saharidis, M. Ierapetritou, I. Androulakis, A mixed-integer optimization framework for the synthesis and analysis of regulatory networks, Journal of Global Optimization 43 (2009) 263–276.

S. Kim, J. Kim, K.-H. Cho, Inferring gene regulatory networks from temporal expression profiles under time-delay and noise, Computational Biology and Chemistry 31 (2007) 239–245.

H. H. McAdams, A. Arkin, Gene regulation: Towards a circuit engineering discipline, Current Biology 10 (2000) R318–R320.

H. D. Jong, Modeling and simulation of genetic regulatory systems: A literature review, Journal of Computational Biology 9 (2002) 67–103.

M. Thattai, A. van Oudenaarden, Intrinsic noise in gene regulatory networks, PNAS 98 (2001) 8614–8619.

C. C. Chicone, Springer, 1999.

E. Kreyszig, Advanced Engineering Mathematics, volume 8th Edition, John Wiley and Sons, Inc, 1999.

V. Pihur, S. Datta, S. Datta, Reconstruction of genetic association networks from microarray data: a partial least squares approach, Bioinformatics 24 (2008) 561–568.

H. Wold, Estimation of principal components and related models by iterative least squares, In Multivariate Analysis (1966) 391–420.

S. Wold, M. Sjostrom, L. Eriksson, Pls-regression: a basic tool of chemometrics, Chemometrics and Intelligent laboratory systems 58 (2001) 109–130.

A. R. McIntosh, N. J. Lobaugh, Partial least squares analysis of neuroimagind data: applications and advances, NeuroImage 23 (2004) S250–S263.

B. S. Dayal, J. F. MacGregor, Recursive exponentially weighted pls and its applications to adaptive control and prediction, Journal of Process Control 7 (1997) 169–179.

S. Datta, Exploring relationships in gene expressions: a partial least squares approach, Gene Expression 9 (2001) 249–255.

P. Geladi, B. R. Kowalski, Partial least squares regression: A tutorial, Anal. Chim. Acta 185 (1986) 1–17.

S. D. Jong, Simpls: an alternative approach to partial least squares regression, Chemometrics and Intelligent Laboratory Systems 18 (1993) 251–263.

A. de la Fuente, D. P. Makhecha, Unravelling gene networks from noisy under-determined experimental perturbation data, IEEE proceedings-System Biology 153 (2006) 257–262.

R. Fisher, Statistical methods and scientific inference, Macmillan, 3rd edition, 1973.

K. Fukunaga, D. M. Hummels, Leave-one-out proedures for nonparametric error estimates, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (1989) 421–423.

K. Fukunaga, D. M. Hummels, Bayes error estimation using parzen and k-nn proceduresq, IEEE Transactions on Pattern Analysis And Machine Intelligence PAMI-9 (1987) 634–643.

H. Jeong, S. P. Mason, A. L. Barabasi, Z. N. Oltvai, Lethality and centrality in protein networks, Nature 411 (2001) 41–42.

J. C. Nacher, T. Ochiai, Power-law distribution of gene expression fluctuations, Physics Letter A 372 (2008) 6202–6206.

M. Hoguland, A. Frigyesi, F. Mitelman, A gene fusion network in human neoplasia, Oncogene 25 (2006) 2674–2678.

H. Bozdogan, Model selection and akaike's information criterion: The general theory and its analytical extensions, Psychometrika 52 (1987) 345–370.

K. Yamaoka, T. Nakagawa, T. Uno, Application of akaike information criterion (aic) in the evaluation of linear pharmacokinetic equations, Journal of Pharmacokinetics and Pharmacodynamics 6 (1978) 165–175.

M. J. L. D. Hoon, S. Imoto, K. Kobayashi, N. Ogasawara, S. Miyano, Inferring gene regulatory networks from time-ordered gene expression data of bacillus subtilis using differential equations, Pacific symposium on Biocomputing, World scientific, singapore 8 (2003) 17–28.

F. Ferrazzi, P. Magni, L. Sacchi, A. Nuzzo, U. Petrovic, R. Bellazzi, Inferring gene regulatory networks by integrating static and dynamic data, International Journal of Medical Informatics 76 (2007) S462–S475.

G. Cedersund, J. Roll, Systems biology: model based evaluation and comparison of potential explanations for given biological data, FEBS Journal 276 (2009) 903–922.

K.-C. Chen, T.-Y. Wang, H.-H. Tseng, C.-Y. F. Huang, C.-Y. Kao, A stochastic differential equation model for quantifying transcriptional regulatory network in saccharomyces cerevisiae, Bioinformatics (2005) bti415–.

H. Akaike, A new look at the statistical model identification, Automatic Control, IEEE Transactions on 19 (1974) 716 – 723.

H. Akaike, Likelihood of a model and information criteria, Journal of econometrics 16 (1981) 3–14.

C. M. Hurvich, C. L. Tsai, Regression and time series model selection in small samples, Biometrika 76 (1989) 297–307.

A. D. R. McQuarrie, C. L. Tsai, Regression and time series model selection, World Scientific, 1998.

R. Ober, S. Montgomery-Smith, Bilinear transformation of infinite-dimensional state-space systems and balanced realizations of nonrational transfer functions, Siam J. Control and Optimization 28 (1990) 438–465.

R. J. Mayhan, Addison-Wesley, 1984.

L. Ljung, System Identification: Theory for the user, Printice Hall, Upper Saddle River. NJ, 1999.

J. M. Varah, A spline least squares method for numerical parameter estimation in differential equations, Siam J. Sci. Stat. Comput. 3 (1982) 28–46.

H. Deng, J. B. Wiskel, A. Ben-Zvi, M. D. Reider, H. Henein, Strain measurement of forming process using digital imaging, Materials Science and Technology 25 (2009) 527–532.

3

Limitations in Inferring Gene Networks from Microarray Datasets [1]

In this chapter, an algorithm for reverse engineering gene networks using data obtained from microarray experiments is proposed. Under the proposed scheme, the parameter space describing gene interaction is partitioned into estimable and inestimable linear subspaces. The estimable subspace is obtained by using principal components analysis (PCA). It is shown that these estimable subspaces are robust with respect to experimental noise. Also, a method for designing experiments which will allow the estimation of the complete network is presented. As a result, the proposed procedure will, necessarily, only allow the estimation of a subset or some

[1] A portion of this chapter has been published in the IFAC proceedings. V. R. Nadadoor, A. Ben-Zvi, and S. L. Shah, "Challenges in Reverse Engineering of Gene Networks from Algebraic Perspective", Proceedings on the 11th symposium Computer Applications in Biotechnology, IFAC symposia, on July 2010.

combination of the entries in A. However, the benefit of the proposed approach is that one is explicitly aware of which portion of the network is identified and which is not.

3.0.1 Principal Components Analysis

As there are more entries in the connectivity matrix A than can typically be estimated from experimental data, one can only estimates a portion of the gene network. Under the proposed framework, key *linear combinations* of genes are identified and their interconnectivity is estimated. Principal Components analysis (PCA) is a statistical technique that can be used to separate or extract the key linear combinations from a set of noise data (Wold, 1978, 1966). PCA has been widely used and has been extremely successful in a number of applications including clustering of gene expression data, assessment of biological age and diagnosis of coronary heart disease (Yeung and Ruzzo, 2001; Nakamura et al., 1988; Brindle et al., 2002)

The singular value decomposition (SVD) algorithm is used to perform PCA on the gene expression data. SVD involves factorization of a given matrix, in this case X^T, into three matrices U, S, and V as shown:

$$X^T = USV^T \tag{3.1}$$

where U consists of orthonormalized eigenvectors associated with eigenvalues of X^TX, and the matrix V consists of orthonormalized eigenvectors of XX^T. S is a diagonal matrix with elements being non-negative square roots of eigenvalues of XX^T, called the singular values.

In the PCA notation, $T = US$ is the score vector, and $\tilde{P} = V$ forms the loading vectors or the principal component vectors (PC). The 1^{st} principal component (PC) captures direction of the greatest variability followed by the 2^{nd} orthogonal PC this relation continues until the n^{th} PC which captures the least variability. Typically last few principal components are assumed to capture the variability due to noise.

The X^T matrix can be written in the PCA notation as follows:

$$X^T = \sum_{i=1}^{d} T_i P_i^T + \sum_{i=d+1}^{n} T_i P_i^T = \tilde{X}^T + \zeta \tag{3.2}$$

where n is the total number of PCs, d is the significant PCs that capture the signal component. T_i's and P_i's are the score vectors and loading vectors of the i^{th} principal component. The scores in Equation 3.2, are in the decreasing order of magnitude as shown:

$$||T_1||_2 > ||T_2||_2 > .... > ||T_n||_2 \tag{3.3}$$

Therefore, T_{d+1} to T_n are scores of PCs which are attributed to noise in the matrix X^T.

3.1 Proposed Method

Let X_{nxm}, represent an experimentally observed gene expression data matrix. First, principal components analysis (PCA) is performed on the data matrix $X^T_{m\times n}$. This allows the matrix X^T to be written as a linear combination of $d < m << n$ principal components representing the signal component in the data, and a set of $n-d$ principal components which represent the noise component in the data. The integer d is chosen using the prediction error sum of squares (PRESS) method (Wold, 1978). The matrix X^T can therefore be written as

$$X^T_{m\times n} = T_{m\times d} P^T_{d\times n} + T^e_{m\times(n-d)} (P^{\perp}_{(n-d)\times n})^T \tag{3.4}$$

where P and T are the loading and score matrices for the first d principal components respectively. Likewise, $P^{\perp}$ and T^e are the loading and score matrices for the remaining $(n-d)$ components. The d loading vectors in the P matrix, and $(n-d)$ loading vectors in the $P^{\perp}$ matrix together form an orthonormal basis. That is, the vectors in the matrix $\tilde{P} = \left[P, P^{\perp}\right] = [P_1, P_2, \ldots, P_d, P_{d+1}, \ldots, P_n]$ form a orthonormal

basis for $\mathbb{R}^n$. In order to simplify the notation, Equation 3.4 can be written as:

$$X^T_{m\times n} = \tilde{X}^T_{m\times n} + \zeta \tag{3.5}$$

$$\text{where, } \tilde{X}^T_{m\times n} = T_{m\text{x}d} P^T_{d\times n}, \text{ and} \tag{3.6}$$

$$\zeta = T^e_{m\times(n-d)} (P^{\perp}_{(n-d)\times n})^T \tag{3.7}$$

are the signal and noise terms respectively.

For a system operating around steady state the gene connectivity matrix can be modeled with a set of linear ordinary differential equations (ODEs)as follows:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{3.8}$$

For the sake of simplicity, in this work, a discrete linear model is assumed to model the gene network instead of ODEs. The system considered is as follows:

$$x(t_{k+1}) = x(t_k) + \Delta t(Ax(t_k) + Bu(t_k)) \tag{3.9}$$

Equation is rewritten in a matrix form as shown:

$$\Delta X = AX + BU \tag{3.10}$$

where $\Delta X = \frac{1}{\Delta t}\begin{bmatrix} (x(1)-x(0)) & \dots & (x(m)-x(m-1)) \end{bmatrix}$. Substituting Equation 3.5 into Equation 3.10, leads to:

$$\Delta\tilde{X} + \Delta\zeta = A\tilde{X} + A\zeta + BU \tag{3.11}$$

Taking expectation of Equation 3.11, gives the following expression :

$$E[\Delta\tilde{X}] + E[\Delta\zeta] = E[A\tilde{X}] + E[A\zeta] + E[BU]$$

Note that using the algorithm proposed in this work, Equation 3.11 contains a noise term ζ whose mean is assumed to be zero (i.e., $E[\zeta] = E[\Delta\zeta] = 0$). Simplifying

and dropping the $E(\cdot)$ notation for compactness one obtains:

$$\Delta\tilde{X} = A\tilde{X} + BU \tag{3.12}$$

To obtain a general solution, Equation 3.6 is substituted into the transpose of Equation 3.12 giving:

$$(\Delta\tilde{X})^T = \tilde{X}^T A^T + B^T = TP^T A^T + (BU)^T \tag{3.13}$$

The least squares solution for $P^T A^T$, in Equation 3.13 is

$$P^T \hat{A}^T = (T^T T)^{-1} T^T ((\Delta\tilde{X})^T - (BU)^T) \tag{3.14}$$

where $\hat{A}$ denotes the least-squares estimate of A. The general solution for $\hat{A}$ in Equation 3.14 is then:

$$\hat{A} = A_0 + C(P^{\perp})^T \tag{3.15}$$

$$\text{where, } A_0^T = P(T^T T)^{-1} T^T ((\Delta\tilde{X})^T - (BU)^T) \tag{3.16}$$

and C is an arbitrary matrix. Recalling that the columns of the $P^{\perp}$ matrix are the principal components associated with the noise in the data, an optimal noise-free estimable portion of A is obtained by setting $C = 0$. For the sake of simplicity, an external stimuli matrix $B_p = BU$ is defined, and henceforth all the equations are rewritten based on B_p. Therefore, the Equation 3.12 can be rewritten in the following form:

$$\Delta\tilde{X} = A\tilde{X} + B_p \tag{3.17}$$

3.1.1 Creating a simulated $\tilde{X}$ matrix and testing the proposed method

A simulated gene expression data, $\tilde{X}$ is built from a given gene network, A and an external stimuli matrix B_p. The simulated example involves obtaining m time samples for analysis given an initial vector $\tilde{x}(0)$, A and B_p.

A procedure is shown for creating these m time samples. A sparse connectivity matrix A, an initial gene expression sample vector, $\tilde{x}(0) = T(0)P^T$, at time t=0, and an external stimuli matrix B_p are chosen. Equation 3.1, can be rewritten for the noise free case as follows:

$$\underline{\tilde{x}}(t+1) = \underline{\tilde{x}}(t) + \Delta t \underline{\tilde{x}}(t)A^T + \underline{b}(t) \tag{3.18}$$

where $B_p = [\underline{b}(0) \ldots \underline{b}(m-1)]$ and $\tilde{X} = [\underline{\tilde{x}}(0) \ldots \underline{\tilde{x}}(m-1)]$.

Equations 3.18 can be written in the matrix form for $t_j = \{0 < 1 < .. < m\}$ as follows:

$$\begin{pmatrix} \underline{\tilde{x}}(1) \\ \underline{\tilde{x}}(2) \\ : \\ \underline{\tilde{x}}(m) \end{pmatrix} = \begin{pmatrix} \underline{\tilde{x}}(0)[I + \Delta t A^T] \\ \underline{\tilde{x}}(1)[I + \Delta t A^T] \\ : \\ \underline{\tilde{x}}(m-1)[I + \Delta t A^T] \end{pmatrix} + \begin{pmatrix} \underline{b}(0) \\ \underline{b}(1) \\ : \\ \underline{b}(m-1) \end{pmatrix} \tag{3.19}$$

While creating a simulated matrix with m time samples , $\tilde{X}$, there is a need to choose an initial vector $\tilde{x}_0$, connectivity matrix A, B, and t.

Let A, be the original connectivity matrix. Equation 3.19 is used to generate $\tilde{X}$ matrix, from the connectivity matrix A, initial sample $\tilde{x}_0$ (at time $t = 0$), and B_p for various time samples t.

The procedure described above is applied to generate $\tilde{X}$ matrix, which in turn is used to re-estimate the connectivity matrix, $\hat{A}$ by applying the method proposed in the current work. Shown below are the A, $\tilde{X}$, B_p, and t data used for the above simulated example:

$$A = \begin{pmatrix} 0 & 6.51 & 3.15 & 0 & 0 & 4.71 \\ -0.33 & 0 & 0 & 6.49 & 1.52 & 0 \\ 2.38 & 0 & 0 & 4.35 & 0.58 & 0 \\ 0 & 20.87 & 15.87 & 0 & 0 & 21.56 \\ 0 & 5.42 & 4.94 & 0 & 0 & 4.45 \\ 2.41 & 0 & 0 & 5.71 & 1.91 & 0 \end{pmatrix}$$

$$\tilde{X} = \begin{pmatrix} -0.24 & -1.25 & -0.94 & -0.94 & -0.24 & -1.25 \\ -0.40 & -1.32 & -0.99 & -1.62 & -0.41 & -1.32 \\ -0.59 & -1.44 & -1.08 & -2.34 & -0.59 & -1.44 \\ -0.78 & -1.60 & -1.20 & -3.13 & -0.78 & -1.60 \\ -1.00 & -1.82 & -1.36 & -4.01 & -1.00 & -1.82 \end{pmatrix}$$

$$B_p = \begin{pmatrix} 0.01 & -0.73 & -0.55 & 0.03 & 0.01 & -0.73 \\ -0.05 & -0.78 & -0.59 & -0.20 & -0.05 & -0.78 \\ -0.18 & -0.02 & -0.01 & -0.71 & -0.18 & -0.02 \\ -0.24 & -0.57 & -0.43 & -0.95 & -0.24 & -0.57 \end{pmatrix}$$

$$t = \begin{pmatrix} 0 & 0.01 & 0.02 & 0.03 & 0.04 \end{pmatrix}$$

3.1.2 Estimates using the Proposed methodology

A set of two random noise components are added to the $\tilde{X}$ matrix and the first step of the proposed methodology as indicated in section 3.1, is applied. The consistency of the estimates below will indicate the significance of considering the noise in the methodology.

Two random noise components with standard deviation of 0.01 and 0.05 are added, to the simulated gene matrix $\tilde{X}$ given in the section 4.1. The connectivity matrices A_1, and A_2, are estimated for the two noise components.

Noise with 0.01 Standard Deviation For noise with standard deviation 0.01, the

loading matrix, P_1, and the score matrix, T_1, as suggested in Equation 3.16 are:

$$P_1 = \begin{pmatrix} -0.1795 & 0.1527 \\ -0.4048 & -0.4757 \\ -0.3036 & -0.3569 \\ -0.7180 & 0.6110 \\ -0.1795 & 0.1527 \\ -0.4049 & -0.4757 \end{pmatrix}$$

$$T_1 = \begin{pmatrix} 2.0576 & 0.8750 \\ 2.6797 & 0.4956 \\ 3.3835 & 0.1450 \\ 4.1856 & -0.2033 \end{pmatrix}$$

The connectivity matrix A_1 estimated in the first step of the methodology is:

$$A_1 = \begin{pmatrix} 0.00 & 5.30 & 3.98 & 0.00 & 0.00 & 5.30 \\ 1.51 & 0.00 & 0.00 & 6.03 & 1.51 & 0.00 \\ 1.13 & 0.00 & 0.00 & 4.53 & 1.13 & 0.00 \\ 0.00 & 21.20 & 15.90 & 0.00 & 0.00 & 21.20 \\ 0.00 & 5.30 & 3.98 & 0.00 & 0.00 & 5.30 \\ 1.51 & 0.00 & 0.00 & 6.03 & 1.51 & 0.00 \end{pmatrix}$$

Noise with 0.05 Standard Deviation For noise with standard deviation 0.05, the loading matrix, P_2, and the score matrix, T_2, as suggested in Equation 3.16 are:

$$P_2 = \begin{pmatrix} -0.1795 & 0.1533 \\ -0.4050 & -0.4743 \\ -0.3036 & -0.3569 \\ -0.7178 & 0.6106 \\ -0.1799 & 0.1545 \\ -0.4050 & -0.4769 \end{pmatrix}$$

$$T_2 = \begin{pmatrix} 2.0578 & 0.8774 \\ 2.6815 & 0.4948 \\ 3.3858 & 0.1398 \\ 4.1870 & -0.2033 \end{pmatrix}$$

Similar to the first case, the connectivity matrix A_2 estimated in the first step of the methodology is found to be:

$$A_2 = \begin{pmatrix} -0.03 & 5.3563 & 4.03 & -0.10 & -0.03 & 5.37 \\ 1.53 & -0.04 & -0.03 & 6.13 & 1.54 & -0.05 \\ 1.15 & -0.05 & -0.04 & 4.60 & 1.16 & -0.06 \\ -0.11 & 21.38 & 16.07 & -0.39 & -0.13 & 21.45 \\ -0.03 & 5.38 & 4.04 & -0.10 & -0.04 & 5.40 \\ 1.54 & -0.08 & -0.06 & 6.14 & 1.55 & -0.09 \end{pmatrix}$$

The sum of squared error of between the two estimates, A_1 and A_2, is given as follows: $\sum\sum(A_1 - A_2)^2$ =0.64. Therefore, the estimated matrices, A_1 and A_2, are quite similar in the presence of noise. Based on the aforementioned two estimates, A_1 and A_2, the following conclusions can be drawn.

1. The connectivity matrix estimated using the proposed methodology helps in obtaining true values for some of the coefficients in the connectivity matrix.

2. It does not give the true estimate for all coefficients in the connectivity matrix. The limitation in the proposed method, in estimating all the coefficients in the connectivity matrix, is explained in a later section.

To further highlight the significance of the proposed methodology, a simulated case study is performed by selecting a set of 100 different random noise components with standard deviations of 0.01 and 0.05 each. Two sets of 100 different connectivity matrices are estimated, using the proposed methodology, for each of the two noise components.

Figure 3.1, is the histogram plot of the variance of the coefficients in the connectivity matrices for the two different noise components with standard deviation of 0.01

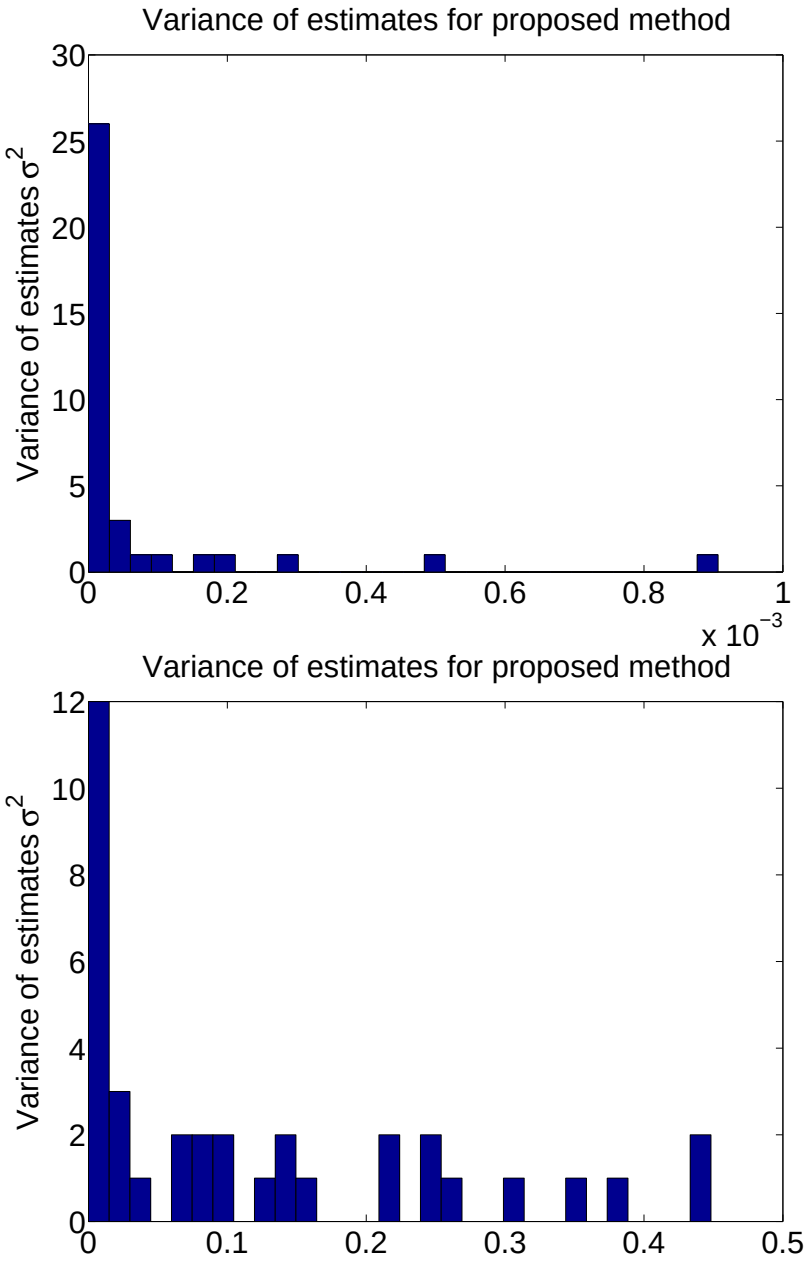

Figure 3.1: Histograms of the variance of the entries of the recovered connectivity matrices estimated by the proposed method. The variance of the coefficients estimated by the proposed methodology for noise components with std. of 0.01 (top panel) and std. of 0.05 (bottom panel) are presented. The scale in the plots suggest a consistent estimate of the entries in the connectivity matrix.

and 0.05. The figure highlights the robustness of the estimates in the presence of noise in the data. Smaller variances for the coefficients in the connectivity matrix as shown in Figure 3.1, indicates the robustness of the methodology proposed in the presence of noise.

3.1.3 Limitations

The method proposed in this work assumes that the gene expression matrix, $\tilde{X}$ is correlated. That is, a few linear combinations of the genes explain the gene expression matrix, $\tilde{X} = TP^T$. The connectivity matrix estimated using this assumption, may not be the true estimate of the connectivity matrix. Given a gene expression data from a given microarray experiment, the solution of the connectivity matrix estimated is not necessarily the true estimate for all the coefficients of the connectivity matrix. The limitation that only a few of the connections can be estimated based on the assumption that $\tilde{X}$ is correlated is highlighted using an illustrative example as shown below:

An illustrative example in the form of a small case study is performed. The simulated example involves obtaining various time samples for analysis given an initial vector, $\tilde{x}_1(0)$, connectivity matrix, A, and an external stimuli matrix, B_1. Let A, $\tilde{x}_1(0)$, B_1 be given as shown below.

$$A = \begin{bmatrix} -1 & 0 \\ -1 & 1 \end{bmatrix}$$

$$\tilde{x}_1(0) = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad B_1 = \begin{bmatrix} 1 & 1 \\ : & : \\ 1 & 1 \end{bmatrix} \tag{3.20}$$

Equation 3.19 is used to calculate $\tilde{x}_1(i),\ \forall i \in \mathbb{N}$, at various times and the result is:

$$\begin{aligned} \tilde{x}_1(1) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \\ \tilde{x}_1(2) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \\ &\vdots \\ \tilde{x}_1(i) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \end{aligned} \tag{3.21}$$

Additional number of time samples does not gives additional information regarding the data. The time samples get trapped into a subspace given by the vector $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. The loading matrix, P_1, and the score matrix, T_1, obtained from Equation 3.16 are given as follows:

$$P_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad T_1 = \begin{bmatrix} 1 \end{bmatrix}$$

The connectivity matrix, $\hat{A}_1$, estimated using the methodology proposed in this work is:

$$\hat{A}_1 = \begin{bmatrix} -1 & 0 \\ -1 & 0 \end{bmatrix}$$

The solution $\hat{A}_1$, does not give a true estimate for all the coefficients in the connectivity matrix. It gives the exact estimate for all the connections in the first column of the A matrix. To get a true estimate for all the coefficients in the connectivity matrix, another case study is performed.

The case study involves obtaining various time samples for analysis given an initial vector, $\tilde{x}_2(0)$, A, and B_2 as shown:

$$A = \begin{bmatrix} -1 & 0 \\ -1 & 1 \end{bmatrix}$$

$$\tilde{x}_2(0) = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & -1 \\ : & : \\ 0 & -1 \end{bmatrix}$$

Once again, Equation 3.19 is used to calculate $\tilde{x}_2(i),\ \forall i \in \mathbb{N}$, at various times:

$$\begin{aligned} \tilde{x}_2(1) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \\ \tilde{x}_2(2) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \\ &: \\ \tilde{x}_2(i) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \end{aligned} \tag{3.22}$$

The loading matrix, P_2, and the score matrix, T_2, obtained from Equation 3.16 are given as follows:

$$P_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad T_2 = \begin{bmatrix} 1 \end{bmatrix}$$

The connectivity matrix, $\hat{A}_2$, estimated using the methodology proposed in this work now:

$$\hat{A}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

The true estimate of the connectivity matrix, $\hat{A}$, as obtained by taking the sum of both the estimates, $\hat{A}_1$ and $\hat{A}_2$ is:

$$\hat{A} = \begin{bmatrix} -1 & 0 \\ -1 & 1 \end{bmatrix}$$

Note that the matrix formed by the loading vectors from the two case studies, [P_1 P_2], form a basis in $\mathbb{R}^2$. Therefore, there exists an underlying relationship between the two case studies shown in this section. This relationship helps in estimating the coefficients of the connectivity matrix in entirety. In the following section, a general procedure for estimating all the coefficients in connectivity matrix is illustrated.

3.2 Independent Microarray Experiments for Estimating Gene Network

Microarray experiments referred in this work constitutes a given gene expression matrix, $\tilde{X}$, and a prescribed external stimuli matrix, B (Gardner et al., 2003; Yeung et al., 2002). The external stimuli matrix B has a direct effect on the A matrix. Therefore a suitable external stimuli matrix, B, is needed to estimate the original connectivity matrix, A.

As indicated in the previous section, from a given microarray experiment, the solution obtained is only a portion of the connectivity matrix. For estimating the complete connectivity matrix, a series of different micro-array experiments have to be performed. This section deals with a methodology of estimating the complete connectivity matrix A.

Any given matrix, A, can be partitioned as follows:

$$A = \tilde{P} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \tilde{P}^T \tag{3.23}$$

$$A = Pa_{11}P^T + Pa_{12}(P^{\perp})^T + P^{\perp}a_{21}P^T + P^{\perp}a_{22}(P^{\perp})^T \tag{3.24}$$

The matrix, A, given in Equation 3.24, will represent the general form of connectivity matrix, A, given in Equation 3.15 if and only if the right-hand side of Equation 3.24 and Equation 3.15 are the same.

The general form of the connectivity matrix as given in Equation 3.15 is as follows:

$$A = A_0 + C(P^{\perp})^T \tag{3.25}$$

where $A_0^T = P(T^TT)^{-1}T^T((\Delta\tilde{X})^T - (B_p)^T)$. The estimate A_0 obtained can be rewritten as follows:

$$A_0 = ((T^TT)^{-1}T^T((\Delta\tilde{X})^T - (B_p)^T))^T P^T = a_0 P^T \tag{3.26}$$

where $a_0 = ((T^T T)^{-1} T^T ((\Delta\tilde{X})^T - (B_p)^T))^T$.

Equating right-hand side of Equations 3.25 and 3.24 leads to the following equality:

$$A_0 + C(P^{\perp})^T = Pa_{11}P^T + Pa_{12}(P^{\perp})^T + P^{\perp}a_{21}P^T + P^{\perp}a_{22}(P^{\perp})^T \tag{3.27}$$

Post multiplying both sides of Equation 3.27 with P and substituting Equation 3.26, leads to the following equation:

$$a_0 P^T P = Pa_{11}P^T P + P^{\perp}a_{21}P^T P \tag{3.28}$$

$$a_0 = Pa_{11} + P^{\perp}a_{21} \tag{3.29}$$

Again, post multiplying both sides of Equation 3.29 with P^T, leads to the following equation:

$$A_0 = a_0 P^T = Pa_{11}P^T + P^{\perp}a_{21}P^T \tag{3.30}$$

Equation 3.30 refers only to a portion of estimate, $\hat{A}_1 = A_0$, of gene connectivity matrix A which is estimated by using the first step of the proposed methodology. The estimate $\hat{A}_1$, only gives the true estimate for some of the connections in the original connectivity matrix. To obtain the true estimate for all connections in the connectivity matrix, a series of independent experiments are needed to be performed. The following section gives a methodology to estimate all the coefficients of the connectivity matrix.

3.2.1 Independent Experiments

Starting with the gene expression matrix, $\tilde{X}_1 = T_1 P^T$, and external stimuli matrix, B_1, satisfying the equation $\Delta\tilde{X}_1 = A\tilde{X}_1 + B_1$, the solution, A_0, obtained in the first step of the proposed method is given as shown in Equation 3.30. That is,

$$\hat{A}_1 = A_0 = (Pa_{11} + P^{\perp}a_{21})P^T \tag{3.31}$$

By an independent experiment, starting with $\tilde{X}_2 = T_2(P^{\perp})^T$ and B_2, satisfying the equation $\Delta\tilde{X}_2 = A\tilde{X}_2 + B_2$, the solution, $A_{0\perp}$, obtained in the first step of the proposed methodology is given by:

$$\hat{A}_2 = A_{0\perp} = (Pa_{12} + P^{\perp}a_{22})(P^{\perp})^T \tag{3.32}$$

Since the number of genes is much greater than the number of samples possible ($n >> m > d$), the estimate, $A_{0\perp}$, given in Equation 3.32 cannot be estimated by a single experiment. Therefore, a series of independent experiments are needed to be performed starting with the gene expression matrices and the external stimuli matrices as shown

$$\tilde{X}_k = T_k v_k^T \text{ and } B_k \; \forall k = 2,3,4,.. \tag{3.33}$$

where v_k is the matrix formed by a subset of the vectors in the matrix $P^{\perp}$. Vectors in each subset matrix v_k form a partition for the $P^{\perp}$ matrix. Each of these gene expression matrix, X_k, and external stimuli matrix, B_k, satisfy the equation $\Delta\tilde{X}_k = A\tilde{X}_k + B_k$.

For each of the independent experiments shown in Equation 3.33, a solution, $\hat{A}_k = A_{0k}$, is obtained using the proposed methodology. The final estimate for the gene connectivity matrix, $\hat{A}$, is the sum of all the estimates obtained from each experiment:

$$\hat{A} = \sum_{i=1}^{\theta} \hat{A}_i \tag{3.34}$$

where θ, is the number of independent experiments performed. $\hat{A}$ is the estimate for the all the coefficients of the connectivity matrix.

3.3 Concluding Remarks

Gene networks is useful in getting a better understanding of mechanisms of complex biological processes such as organ transplant rejection and breast tumors. It is a well known fact that reverse engineering of such gene networks from gene expression data tend to be underdetermined. Also, the gene expression data obtained

from microarray experiments are very noisy. In this work, the identification of the gene network is treated as a step-wise problem. The gene network was separated into estimated and unestimated components based on the experiments performed. The robustness of the data in the presence of noise is also discussed in this article. It is also shown, the limitations of the methods available in the literature have also been discussed. Simulated examples generated, illustrates the importance of this work.

The model obtained gives true estimate for some of the connections in the gene network. The method also suggests the need for further microarray experiments to be performed for constructing the gene network topology in entirety. Overall, the importance of this work is get an understanding of various different portions or partitions of the gene networks and suggests a procedure for estimating each one of portions individually.

References

S. Wold, Cross-validatory estimation of the number of components in factor and principal component models., Technometrics 20 (1978) 397–405.

H. Wold, Estimation of principal components and related models by iterative least squares, In Multivariate Analysis (1966) 391–420.

K. Y. Yeung, W. L. Ruzzo, Principal component analysis for clustering gene expression data, Bioinformatics 17 (2001) 763–774.

E. Nakamura, K. Miyao, T. Ozeki, Assessment of biological age by principal component analysis, Mechanisms of Ageing and Development 46 (1988) 1–18.

J. T. Brindle, H. Antti, E. Holmes, G. Tranter, J. K. Nicholson, H. W. L. Bethell, S. Clarke, P. M. Schofield, E. Mckillingin, D. E. Mosedale, D. J. Grainger, Rapid and noninvasive diagnosis of the presence and severity of coronary heart disease using 1h-nmr-based metabonomics, Nature Medicine 8 (2002) 1439–1445.

T. S. Gardner, D. di Bernardo, D. Lorenz, Inferring genetic networks and identifying compound mode of action via expression profiling, Science 301 (2003).

M. K. Yeung, T. J, C. J. J, Reverse engineering gene networks using singular value decomposition and robust regression, PNAS 99 (2002) 6163–6168.

4

Online Sensor for Monitoring a Microalgal Bioreactor System Using Support Vector Regression [1]

4.1 Introduction

The biotechnological use of microalgae for the production of fine chemicals and biofuels is of growing interest due to the higher growth rate and productivity of algae compared to higher plants (Chisti, 2007). Moreover, microalgae can be inten-

[1] A version of this chapter has been accepted for publication. V. R. Nadadoor, H. De la Hoz Siegler, S. L. Shah, W. C. McCaffrey, and A. Ben-Zvi, "Online Sensor for Monitoring a Microalgal Bioreactor System Using Support Vector Regression", Accepted for publication in the Chemometrics and Intelligent Laboratory Systems, 2011.

sively grown in traditional bioreactors, reducing the pressure over cropland (Singh et al., 2011). Several microalgae species are remarkable for their capacity to produce and store large amounts of oil. For example, lipid content in *A. protothecoides* can represent up to 57.8% of the cell dry weight when grown heterotrophically (Xiong et al., 2008). In the heterotrophic growth mode, an organic substrate is used as both the carbon and energy source. It has been shown that when growing either heterotrophically, or photoheterotrophically microalgae exhibits a higher productivity in terms of either biomass or oil when compared to phototrophically cultured algae (Liu et al., 2011; Liang et al., 2009).

Oil production in algae has been shown to be dependent on culture conditions (De la Hoz et al., 2011), and therefore appropriate monitoring and control of these conditions are required in order to maximize oil productivity. From a process control perspective, it is desirable to know, at any given moment, the cell concentration, oil content, and substrate concentration in the reactor. These quantities, however, are rarely directly measured, as their quantification involve a series of elaborate and time consuming steps. For example, cell concentration in the reactor is usually expressed in terms of cell dry weight per unit volume. Dry weight determination requires the removal of a sample from the reactor, centrifugation and washing, and further drying of the sample until constant weight is achieved. This procedure can take anywhere from two hours up to a day, to be completed. Similarly, intracellular oil content and extra-cellular nutrient concentration require several hours or even days to be determined. Oil is generally quantified by solvent extraction of a dry algal sample or by derivatization and chromatographic quantification, and the substrate concentration is determined by gas or liquid chromatography. Furthermore, highly qualified personnel are required for measuring the cell and substrate concentrations along with quantifying the oil content.

In this work, an online multivariate sensor based on support vector regression is developed to monitor the concentrations of biomass, glucose and oil content in microalgal cultures in a bioreactor system. A portion of the study is dedicated for comparing and highlighting the superior performance of the proposed method with respect to other techniques available to build online sensors. Also, the effect of

several preprocessing techniques on the goodness of model fit is assessed. A review of the current status of Raman spectroscopy, as a tool for bioprocess monitoring is presented in the next section.

4.2 Background

Developing sensors for online monitoring of bioprocess systems has been extensively studied and successfully applied using various different types of spectroscopic methods including fluorescence spectroscopy (Marose et al., 1998; Skibsted et al., 2001) and near-infrared spectroscopy (Landgrebe et al., 2010; Yeung et al., 1999). Furthermore, performance improvement methods for online monitoring of bioprocess systems by reducing the prediction error of the concentration estimates have also been studied (Dabros et al., 2009). However, the lack of detailed structural information obtained by these spectroscopic methods limits their use for the identification of the chemical constituents in complex samples, as in the case of algal bioreactors.

Raman spectroscopy has the potential to be used as a process analytical technology to estimate several key process variables in algal bioreactors (Huang et al., 2010). The Raman scattering is produced by the inelastic interaction between light and matter. These inelastic interactions are highly dependent on the vibrational characteristics of the molecular bonds of the components in the sample under analysis. As such, the Raman spectra will be a function of all of the cellular components (i.e., proteins, lipids, DNA, etc.) as well as constituents in the growth media. Of course, this implies that the generated spectra will be highly convoluted, due to the presence of thousands of components in the cell and the culture media.

Shope et al. (1987) were the first to propose the use of Raman spectroscopy for bioprocess monitoring, namely, the analysis of ethanol fermentation products. The Raman spectra were measured off-line and it was shown that several features of the spectra can be used for quantifying the concentration of the fermentation products. However, no model was built and fluorescence was reported as the main predica-

ment that hindered proper model building. To reduce the effect of fluorescence, Xu et al. (1997) compared two different laser sources (Argon ion at 514.5 nm and solid state diode laser at 785 nm) and removed the cells from the broth. It was reported that the 785 nm laser substantially eliminated the background fluorescence and improved the limit of detection by a factor of 5, thereby allowing the simultaneous measurement of concentration of glucose, glutamine, lactate and ammonia in the fermentation broth. Shaw et al. (1999) followed the fermentation of glucose to ethanol on-line by using a flow-thru cell (ex-situ), concluding that Raman spectroscopy is an ideal method for following biotransformations in a nondestructive and noninvasive way. As in the case of Xu et al. (1997), Shaw et al. (1999) used a 780 nm laser and removed the cells from the broth, through an in-line filter, previous to spectra acquisition.

The first on-line and in-situ application of Raman spectroscopy to monitor a bioprocess was reported by Cannizaro et al. (2003). A 785 nm laser and a 12.5 mm immersion Raman probe inserted in a side port of the bioreactor were used. The probe was connected to the control unit with a fiber optic. Carotenoids production by *Phaffia rhodozyma* was quantified. Cannizaro et al. (2003) took advantage of the unique enhanced Raman signal characteristic of carotenoids to build a calibration model without the use of complex chemometric tools for signal deconvolution and without removing the cells from the sample. Lee et al. (2004) monitored *Escherichia coli* bioreactions using Raman both in-situ and off-line. Limited accuracy of the on-line measures was reported, which was associated to a change in the Raman spectrum of the sapphire window probe after steam-sterilization. A chemometric model was built using data from pure components spectra measured off-line and before probe sterilization.

The increasing interest in the technological applications of microalgae has arisen the need for proper quantification of microalgal products. Currently, analytical methods for biomass and product quantification in microalgal cultures are time consuming and prone to error. An on-line, multivariate, spectroscopic monitoring tool has the potential to facilitate and speed algal bioprocess development and commercialization. Huang et al. (2010) stated that the Raman spectra are related

to the key variables in a microalgal culture. Their study, however, did not involve obtaining a relationship between the spectra and the components in the culture. The foremost application involving the quantification of lipids in a microalgal system was proposed by Wu et al. (2010). The study demonstrated that Raman spectroscopy can directly obtain quantitative information of the lipids, albeit in single cells. Recently, Abbas et al. (2011) studied the distribution of carotenoids in single algal cells using Raman spectroscopy. There is, however, a need for a comprehensive understanding of a quantitative relationship between the spectra and the components in the culture media.

The main aim of the present study is to construct chemometric models, in a statistical or mathematical framework, to estimate chemical compositions in a faster and noninvasive procedure. Chemometric models of spectroscopic data have previously been built using known statistical and machine learning tools including principal component regression (PCR) (Estienne and Massart, 2001), partial least squares (PLS) (Goetz et al., 1995), and support vector machines (SVR) (Thissen et al., 2004).

The use of principal components in regression was first suggested by Kendall (1957) and Hotelling (1957). Since then PCR have been successfully applied in various fields including chemometrics ((Marbach and Helse, 1990; Naes and Martens, 1988)), flow-injection analysis ((Blanco et al., 1993)), and biomedical studies for multi-class cancer classification (Tan et al., 2005). Partial least squares (PLS) was first proposed by Herman Wold during mid-sixties (Wold, 1966) and subsequently found success in various applications in the field of chemometrics (Sjostrom et al., 1983; Wold et al., 2001), neuro imaging (McIntosh and Lobaugh, 2004), and process control (Dayal and MacGregor, 1997). The robustness of PCR and PLS to overfitting, makes it an important tool in the field of chemometrics. One of the major disadvantages of the PCR and the PLS is their inadequacy when applied to nonlinear systems (Demiriz et al., 2001). To deal with the system nonlinearities, the regression method based on support vector learning can be used.

Along with handling of system nonlinearities, the support vector learning methodology has other advantages over the traditional PCR and PLS methods which include

better performance in the presence of outliers in the calibration dataset, superior modeling with a smaller dataset, and a simpler model (in terms of order) obtained based on the structural risk minimization (SRM) principle as opposed to empirical risk minimization (ERM), employed by the PLS and PCR methods. Based on SRM principle, SRM minimizes the loss function (empirical risk) as well as the model complexity (structure of the model), thus avoiding overfitting. On the other hand, ERM only minimizes the loss function (empirical Risk) defined for the task (Khatibisepehr et al., 2011). The combined application of Raman spectroscopy and support vector regression (SVR) was presented by Barman et al. (2010) for monitoring blood glucose levels. Barman et al. (2010) showed that the use of nonlinear SVR model represents a 30% enhancement in prediction accuracy over the PLS model, when measurements from multiple human volunteers were considered.

4.3 Theory

4.3.1 Support Vector Regression

Support vector regression (SVR) was developed as an extension of the theory of support vector machines(SVM) to regression problems (Scholkopf and Smola, 2002). The support vector algorithm was proposed by Vapnik in 1992 and was later developed over the years (Boser et al., 1992). The concept of support vector learning has been successfully applied to various classification and regression problems including the development of robust calibration models for monitoring blood glucose levels (Barman et al., 2010), applying SVR for multivariate nonlinear processes (Chitralekha and Shah, 2010; Khediri et al., 2010), material optimization of salon ceramics (Xu et al., 2006), and identification of time series models (Thissen et al., 2003).

Given a training dataset, $\{(x_1,y_1)...(x_m,y_m)\} \subset \mathbb{R}^n \times \mathbb{R}$, regression involves minimizing a loss function. In the case of a simple least squares regression, the quadratic loss function shown in Equation 4.1 is minimized.

$$\min_{w} L = \sum_{i=1}^{m} (y_i - f(x_i, w))^2 \tag{4.1}$$

where $y = f(x,w)$ is the linear function used for the regression problem; m is the number of sample points; x_i and y_i are the i^{th} independent predictor variable and observation respectively; w is the parameter vector that defines the function f.

In SVR, a new ε-insensitive loss function, $L(|y - f(x,w)|_\varepsilon)$, is defined as suggested in (Vapnik, 1998):

$$L(|y - f(x,w)|_\varepsilon) = |y - f(x,w)|_\varepsilon \tag{4.2}$$

where,

$$|y - f(x,w)|_\varepsilon = \begin{cases} 0 & |y - f(x,w)| \leq \varepsilon \\ |y - f(x,w)| - \varepsilon & \text{otherwise} \end{cases} \tag{4.3}$$

For the case of linear regression, a linear function, f, is defined as follows:

$$f(x,w,b) = \langle w, x \rangle + b \tag{4.4}$$

In SVR, the goal is to find the optimal variables (w^*, b^*) that generate the function, $f^*(x)$, that gives the minimum loss function. This problem is formulated as a constrained convex optimization problem:

$$\min_{w,b,\xi_i,\xi_i^\star} J = \frac{||w||^2}{2} + C \sum_{i=1}^{m} (\xi_i + \xi_i^\star) \tag{4.5a}$$

$$\text{subject to} \begin{cases} f(x_i,w) - y_i \leq \varepsilon + \xi_i \\ y_i - f(x_i,w) \leq \varepsilon + \xi_i^\star \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{4.5b}$$

where $C > 0$ is the regularization parameter, which is a tradeoff between the penalty imposed on w and the tolerance on deviations larger than ε; ξ_i and ξ_i^* are slack variables that allow the constraints to have a training error greater than ε and also penalize them in the objective function; and $i = 1, 2, .., m$ are the training data points.

Figure 4.1 is a graphical depiction of the ε-SVR model.

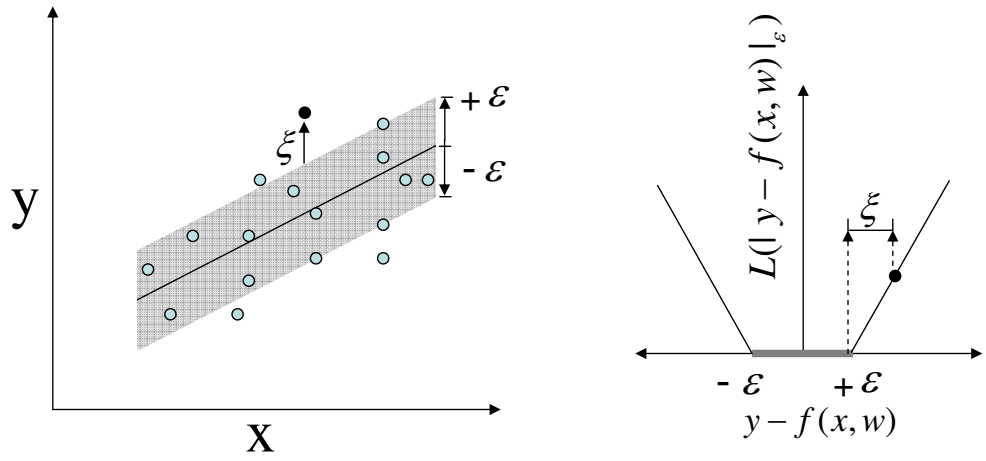

Figure 4.1: Graphical Representation of the ε-SVR model for a linear case (Chitralekha and Shah, 2010).

The use of an ε-insensitive loss function has been previously investigated in great detail (Vapnik, 1998). The ε-insensitive loss function builds a tube of insensitivity with only the points outside the tube being penalized so as to minimize the resulting errors in the objective function. The value of ε affects the smoothness of the SVRs response and also affects the number of support vectors, so both the complexity and the generalization capability depend on its value. Also, there is a considerable investigation regarding the noise model of the ε-SVR method (Pontil et al., 2000; Kwok and Tsang, 2003). The ε-insensitive loss function can be used when the noise affecting the data is assumed to be additive and Gaussian. The mean and variance of the noise model, however, are random variables whose probability distributions can be computed explicitly (Pontil et al., 2000).

The constrained minimization, given in Equations 4.5a and 4.5b, is a standard problem in optimization theory. This can be solved by constructing the Lagrangian

for the objective function and the constraints. By solving the Lagrangian, the weight vector, w, can be derived as follows:

$$w = \sum_{i=1}^{m} (\alpha_i - \alpha_i^\star) x_i, \tag{4.6}$$

where $\{\alpha_i, \alpha_i^\star\}$ are the Lagrange multipliers associated with the training point x_i. The linear function,in Equation 4.4, can be rewritten as follows:

$$f(x) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^\star) \langle x_i, x \rangle + b \tag{4.7}$$

Equation 4.6 indicates that the weight vector w can be described as a linear combination of training vectors, which in turn leads to the property that, for evaluating $f(x)$ it is not required to explicitly calculate the weight vector, w. These observations become important when the linear SVR is extended to the nonlinear case.

The basic idea behind the nonlinear SVR is to project $\{x_i\}$ onto a feature space F. The aforementioned linear SVR algorithm is then applied to the projected dataset. Let $\phi(x)$ be a mapping that maps the x according to the relation $\phi : \mathbb{R}^n \to F$. A linear function, f, in the projected space is then defined as follows:

$$f(\phi(x)) = \langle w, \phi(x) \rangle + b \tag{4.8}$$

In short, the nonlinear SVR algorithm behaves like a linear one, if the input vectors x_i's are replaced by their corresponding feature vectors $\phi(x_i)$. Projecting the training data into a very high dimensional space is computationally expensive. Due to the exclusive dot product form in Equation 4.8, the computation complexity involved in obtaining the projected training datasets can be avoided with the help of a "kernel trick" (Boser et al., 1992). The kernel function is represented as follows:

$$k(x_i, x_j) = \left\langle \phi(x_i), \phi(x_j) \right\rangle \tag{4.9}$$

Using the kernel function suggested in Equation 4.9, for the nonlinear case, the

function f can be transformed to:

$$f(\phi(x)) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^\star) \langle \phi(x_i), \phi(x) \rangle + b = \sum_{i=1}^{m} (\alpha_i - \alpha_i^\star) k(x_i, x_j) + b \qquad (4.10)$$

analogous to the linear case given in Equation 4.7.

The most used kernel functions are the Gaussian RBF-kernel, $k(x_i, x_j) = e^{-\gamma ||x_i - x_j||^2}, \gamma > 0$; and the polynomial kernel with an order of d, $k(x_i, x_j) = (\gamma x_i^T x_j + constant)^d, \gamma > 0$. It can be seen that the linear kernel is a polynomial kernel with order equal to one ($d = 1$).

In this work, a Gaussian RBF kernel is used, as it is a very useful kernel and its application to support vector regression problems is widespread (Chitralekha and Shah, 2010). Application of the RBF-kernel based SVR is demonstrated, in this work, by building a multivariate sensor for monitoring the biochemical composition of a microalgal bioreactor.

4.4 Materials and Methods

4.4.1 Experiment Setup

The algae *Auxenochlorella protothecoides*, UTEX B25, was cultured heterotrophically in a 2L bioreactor (Sartorious Biostat A plus). The experimental setup of the bioreactor is shown in Figure 4.2. A solid-state fiber Bragg grating stabilized laser, with an excitation wavelength of 785 nm and output power equal to 300 mW, was used for obtaining the Raman spectra. The Raman spectrometer consisted of an f/4 symmetrical crossed Czerny-Turner monochromator, with a 50 μm wide slit, and a 1024 x 58 pixels (2D array) Hamamatsu detector. Raman spectra were acquired using an immersion probe inserted in one of the upper ports of the bioreactor. The stainless steel immersion probe was chemically sterilized by submerging it in a mixture of benzalkonium chlorides (Roccal-D) for at least 15 minutes prior to its installation in the bioreactor. The bioreactor temperature was kept constant at

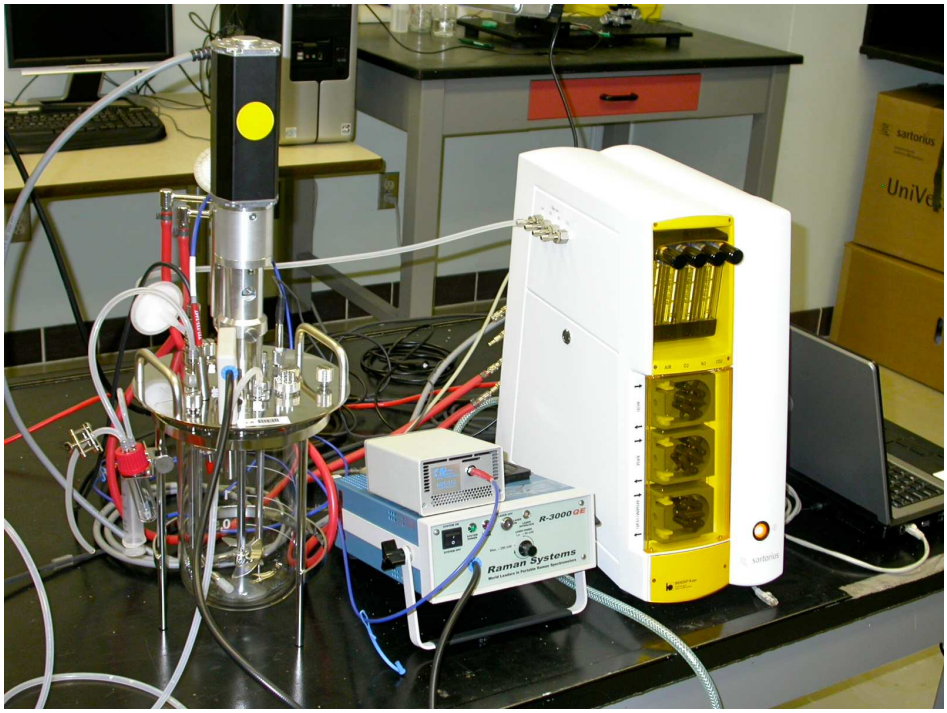

Figure 4.2: A picture depicting the 2L bioreactor system (on the left) and the digital control unit (on the right)

25 °C. Raman spectra were collected every 10 minutes, and recorded and processed using MATLAB. For each measurement, the spectrometer grating channel was left with a laser source turned off, in order to record the background radiation. These background radiation spectra were subsequently subtracted from the spectra of the culture media.

For model building and validation, samples were withdrawn from the bioreactor at four hour intervals and analyzed, to determine the algal concentration, oil content, and substrate concentration. A model was built, using the calibration dataset, for measuring the concentration of three main components in the bioreactor, namely, biomass, glucose and oil content. A brief description of the procedures used for the off-line measurement of the concentrations of biomass, glucose and oil content is provided below.

Biomass concentration was determined as total suspended solids (TSS), by centrifuging 1.4 mL of cell suspension (RCF = 9335 g) for 10 minutes. The obtained pellets were washed twice with a saline phosphate buffer solution (pH 6.2). The washed pellets were centrifuged again and the resulting precipitates were vacuum dried at a temperature of 50°C and a pressure of 0.1 bar until the precipitate attained a constant weight. The clear supernatant from the centrifugation was filtered using a 0.22 μm syringe filter in order to remove any residual cells.

Glucose concentration in the filtered supernatant was measured by high performance liquid chromatography (Agilent 1200 Series HPLC), using a SupelcoGel Pb carbohydrate column at 70 °C (Internal diameter 7.8 mm, length 30 cm) with a guard column. Sample injection volume was 10 μL; eluent was deionized, sterile water (MilliQ, MilliPore); elution flow-rate was set at 0.5 mL/min, and a refractive index detector (RID) at 35 °C was used.

Oil content in the cells was determined by fluorospectrometry of cells stained with Nile Red. In this method, florescence intensity is linearly correlated to the total neutral lipid content of the cells. A 10 μL aliquot of a 10 μg/mL Nile Red solution in ethanol was added to the individual wells of a 96-microplate containing 10 μL samples of 10 g/L algal cells. The volume in each well was completed to 200 μL by adding a 30% (v/v) ethanol solution in water. Samples were incubated

at 40 °C for 10 min, and fluorescence emissions were recorded with a multiplate reader spectrophotometer (Fluoroskan Ascent, Thermo Labsystems). Excitation and emission wavelengths were selected at 530 nm and 604 nm, respectively. Nile Red oil measurements were calibrated using algal cells for which oil content had been previously determined gravimetrically, following the method developed by Hara and Radin (1978). An algal sample of known oil content, as determined gravimetrically, was used in each micro-plate run as internal standard. Fluorescence measurements were performed in triplicate, and the average standard error was 5.6%.

Three different datasets were generated by running the reactor in fed-batch mode starting at different initial conditions and by varying the feeds flowrate. The first dataset (DS1) corresponds to a D-Optimal run, as reported in (Surisetty et al., 2010). In this case, algae were cultured over a period of 360 h, and glucose, glycine, and minerals were supplemented to the reactor in order to generate significant perturbations in the bioreactor response. For the second dataset (DS2), feed flow followed a pseudo-random binary profile, as presented in (De la Hoz et al., 2011). In the third dataset (DS3), culture conditions were modified in order to maximize biomass production. A summary of the three datasets is presented in Table 4.1, where the range of the three measured variables, and the number of data points in each data set is presented.

Table 4.1: Number of samples and range of concentration values for biomass, glucose, and oil content for all three datasets

Dataset	Range of concentration values			No. of samples
	Biomass (g/l)	Glucose (g/l)	Oil content (% w/w)	
DS1	0.75-39.36	0-101.90	14.27-65.07	79
DS2	0.50-38.20	0.01-52.30	19.40-79.10	78
DS3	2.40-144.29	0.05-45.59	32.88-82.06	57

4.4.2 Preprocessing Methods

Raman Spectra Preprocessing

Before building a chemometric model, preprocessing of Raman spectra is performed. The advantages of preprocessing the Raman spectra using various smoothing and transformation methods have been extensively studied (Afseth et al., 2006; Chau et al., 2004). Chau et al. (2004) and Martens and Naes (1989) provide a basic description of the preprocessing methods applied in the current study. The preprocessing techniques applied in this work are:

- Savitzky-Golay (SG) filtering: Savitzky-Golay filter is a smoothing filter based on polynomial regression. For the Savitzky-Golay method, a third order polynomial with a section size of 7 points is used.
- Standard normal variate (SNV) transformation: A standard normal variate transformation is performed to the Raman spectra such that the resulting spectra have mean zero and unitary variance.
- Linear polynomial baseline correction (Polyfit): In the linear polynomial baseline correction method, a peak selection algorithm is used to identify the peaks. A linear polynomial is fitted to the baseline values for each of these obtained peaks. The resulting polynomial curve (line) is then subtracted from the raw Raman spectra.
- Combination of standard normal variate transformation and linear polynomial baseline correction method (SNV&Polyfit): The Raman spectra is first transformed using standard normal variate and then linear polynomial baseline correction is performed on the resulting transformed spectra.
- Combination of Savitzky-Golay smoothing filter and standard normal variate transformation (SG&SNV): The Raman spectra is first smoothed using the Savitzky-Golay filter and the resulting spectra is transformed using the standard normal variate transformation.

- Combination of Savitzky-Golay smoothing filter, standard normal variate, and linear polynomial baseline correction (SG&SNV&Polyfit): Raman Spectra is first smoothed using the Savitzky-Golay smoothing filter and the resulting spectra are preprocessed using the standard normal variate and linear polynomial baseline correction techniques.

Preprocessing of the Measured Concentrations

The concentrations of the chemical components in the bioreactor system are susceptible to measurement noise. Therefore, a preprocessing technique in the form of a filter is necessary to reduce the effect of measurement noise in the data used for model building. To this end, robust LOESS (locally weighted quadratic regression) method (Cleveland, 1979; Cleveland and Devlin, 1988; Hastie and Tibshirani, 1986) is used for smoothing the measurements along different samples. The filtered data is subjected to further preprocessing by performing standard normal variate (SNV) transformation on the data. The final measured data, after smoothing and normalization, is used for model building purposes.

4.4.3 Optimal Selection of Model Parameters

For building a chemometric model, it is necessary to determine the optimal number of model parameters to obtain an accurate model, while avoiding overparameterization. The non-linear radial basis function support vector regression algorithm used in this work possesses three adjustable parameters: the soft margin (C) for the regression cost function, the threshold parameter (ε), given in Equations 4.5a and 4.5b, and the radial basis function kernel parameter (γ). To determine the optimal value of these three parameters, a systematic grid search (refer to (Hsu et al., 2003) for details) was performed in combination with a 10-fold cross-validation method using the predicted residual sum of squares (PRESS) statistic.

In the 10-fold cross-validation method, the calibration dataset is divided in 10 subsets. The regression model is calibrated using 9 of these subsets and the resulting

model is evaluated in the remaining subset. The calibration is repeated 10 times, leaving out at each iteration a different subset. The PRESS statistic is computed for each one of the 10 regression models constructed, and the average PRESS value is used as a measure of the goodness of fitting provided by the combination of C, ε, and γ values. This procedure was performed for every value in the parameter space, to determine the parameter combination that reduces the average PRESS.

4.4.4 Model Building

For building a robust sensor for biomass, the datasets obtained from the three experiments (DS1, DS2, and DS3 mentioned in Section 4.4.1) are divided into calibration and validation datasets. The calibration dataset is obtained using the equal-weighting (EW) method described as follows:

Step 1. The samples from the first dataset (DS1), the second dataset (DS2), and the third dataset (DS3), mentioned in Table 4.1, were combined together into a single combination dataset of 214 samples.

Step 2. The combined dataset was sorted in increasing order of concentration of the component to be modeled.

Step 3. The sorted combined dataset was partitioned into 130 equal subgroups based on the maximum and minimum values in the component concentration. That is, the i^{th} subgroup is the partition that includes the samples whose component concentration value falls in the range,

$$\left[\; \mathrm{MN} + (i-1) * \frac{\mathrm{MX}-\mathrm{MN}}{130} \quad \mathrm{MN} + i * \frac{\mathrm{MX}-\mathrm{MN}}{130} \; \right]$$

where MN and MX are the maximum and minimum concentration values of the components to be modeled. A point to be noted is that the partition of subgroups based on this approach can lead to the possibility of some of the subgroups being empty (containing zero samples).

Step 4. One sample was chosen from each of the 130 subgroups unless the subgroup

was empty. Due to presence of these empty subgroups, less than 130 samples were obtained for the calibration.

Step 5. The remaining samples from the combined dataset were stored in the residual dataset (RS).

The residual dataset (RS) obtained was used for validating the model. In total, 60 samples were obtained for calibration and the remaining 154 samples were used for validation. Figure 4.3 shows a flow chart that illustrates the procedure of obtaining the calibration and validation datasets for the biomass concentration.

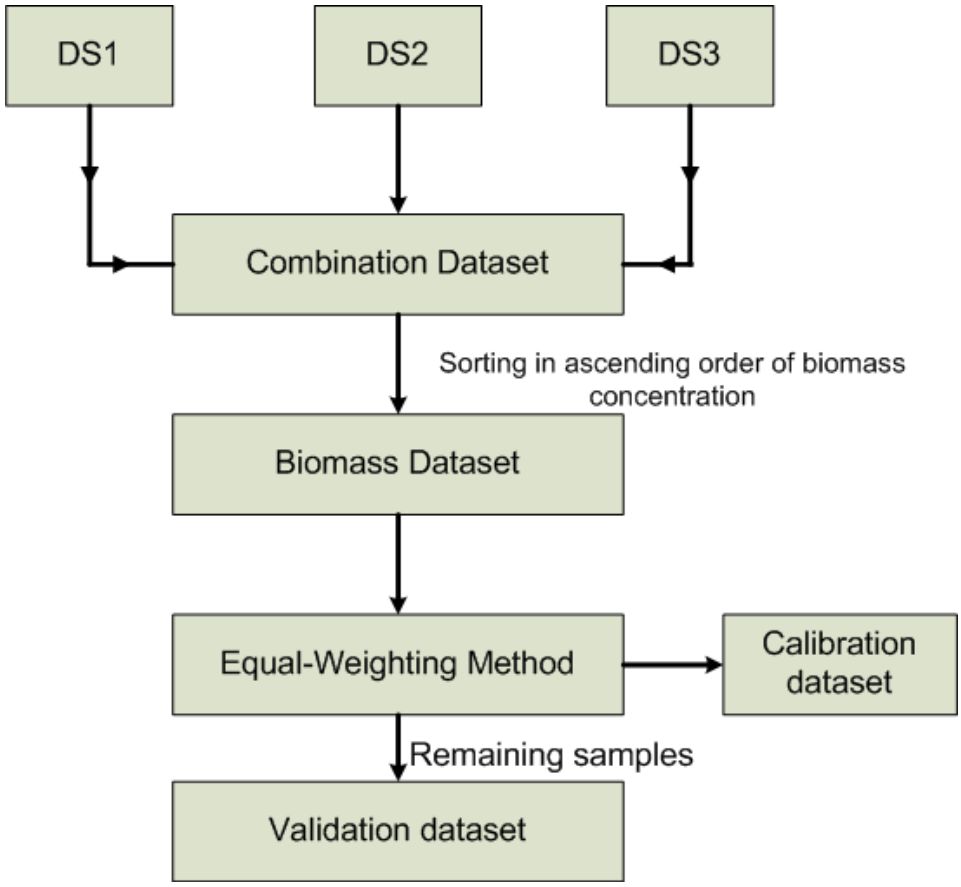

Figure 4.3: Flowchart illustrating the method used for obtaining the calibration and validation datasets for the biomass concentration

The partition of the data is carried out to guarantee that the measurements selected for model building cover the entire range of biomass concentrations. The biomass concentration increased monotonically following a sigmoidal profile with regions overloaded with data with a minor variation in the measurement values and regions with large variations in the measurements and very few data points. In this case, the application of a random selection method could lead to a choice of a large portion of measurements with not enough variations and thereby building a non-robust model.

For building a sensor for glucose concentration and oil content, however, dataset (DS3) was chosen for building calibration and validation datasets. The reason for choosing only the third dataset (DS3) for model building and validation is due to the existing correlation between glucose concentration and oil content with biomass concentration, as discussed below.

The effect of biomass on the Raman spectra is predominant compared to that of the glucose concentration and the oil content. Therefore, building a sensor for glucose concentration and oil content using a dataset where there is a strong correlation between these two variables and the biomass concentration could lead to a bias in the model. In the first two datasets (DS1 and DS2), the change of concentration in glucose and oil content is found to be related to the change in the biomass concentration. Table 4.2, shows the correlation coefficient values for glucose concentration and oil content with the biomass concentration, for the three datasets (DS1, DS2, and DS3).

Table 4.2: Correlation coefficient value between glucose and biomass concentrations and oil content and biomass concentrations for all three datasets (DS1, DS2, and DS3)

Datasets	Correlation Coefficient	
	Biomass and Glucose	Biomass and Oil
DS1	0.4708	0.5197
DS2	0.6020	0.4353
DS3	0.0026	0.0088

Based on the correlation coefficients, shown in Table 4.2, there exists a strong correlation between glucose and biomass concentrations for datasets DS1 and DS2. Hence, it cannot be ascertained with a high confidence that a model built for glucose concentration will not incorporate some of the relationship (trend) that exists between biomass and glucose concentration in the two datasets. Given the existing cross correlation in the calibration dataset, it is not possible to achieve a total deconvolution for the individual effects of biomass and glucose on the Raman spectra. Likewise, the significant correlation coefficient values, as shown in Table 4.2, between oil content and biomass concentration lead to a similar conclusion of model bias.

The calibration and validation datasets for glucose concentration and oil content are built by random data selection method, using only the third dataset (DS3). As per the random data selection method, a subset of 30 random samples was chosen from the dataset DS3 for building the calibration dataset and the remaining 27 samples were used for validating the model.

4.5 Results and Discussion

The unprocessed Raman spectra of microalgal cultures are highly complex due to the presence of thousands of components in the media. This is highlighted in Figure 4.4, where the Raman spectra for two different algal samples with varying compositions are shown.

It is difficult to readily associate changes in the characteristics for the two spectra, presented in Figure 4.4, with changes in the culture compositions even though the respective concentrations of biomass, glucose, and oil content are very different. For an experimentalist untrained in advanced signal processing techniques, trying to use Raman spectra for estimating the chemical composition of the algal cultures, it is practically impossible to extract the significant peaks for each of the components in the sample matrix.

Additionally, the presence of other factors including sample fluorescence (as men-

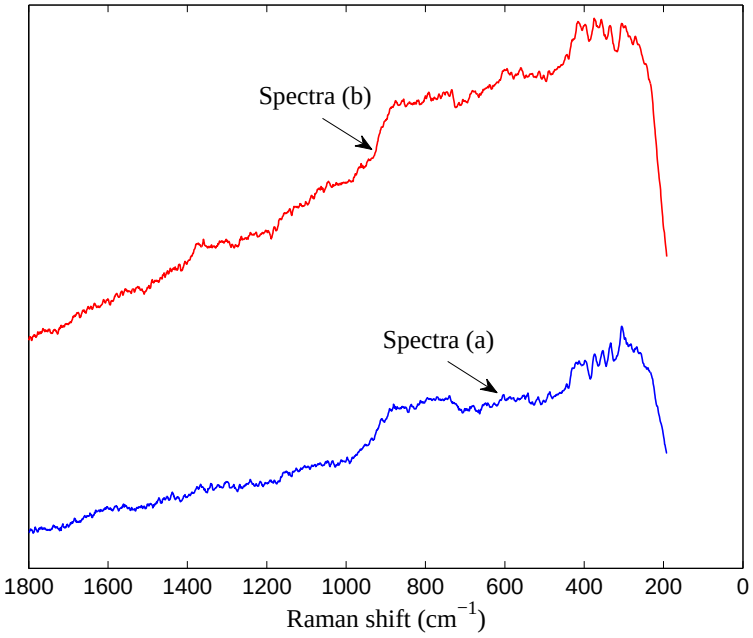

Figure 4.4: Unprocessed Raman spectra of A. protothecoides liquid cultures. The concentration of glucose and biomass in the media, and the intracellular content of oil were determined offline as: a) (Blue curve in online version) 33.3 (in g/l), 2.07 (in g/l), and 35.3 (% w/w) respectively; and b) (Red curve in online version) 108.8 (in g/l), 40.0 (in g/l), and 53.0 (% w/w) respectively

tioned in Section 2), turbidity, and bubbles in the system, causes variations in the intensity of the spectra and introduces spurious peaks. These disturbances add to the difficulty in the extraction of relevant information for model building. To emphasize the effect of fluorescence on Raman spectra, the spectra of freeze-dried cells of A. protothecoides were collected ten times, at regular intervals. After each collected spectra, the total exposure time of the sample to the laser source was consequently higher. In Figure 4.5, it can be seen that the total Raman count decreased as exposure time to the laser increased. This result indicates that there was background fluorescence coming from the sample, as photo-bleaching usually results in a significant reduction in the intensity of the fluorescent background.

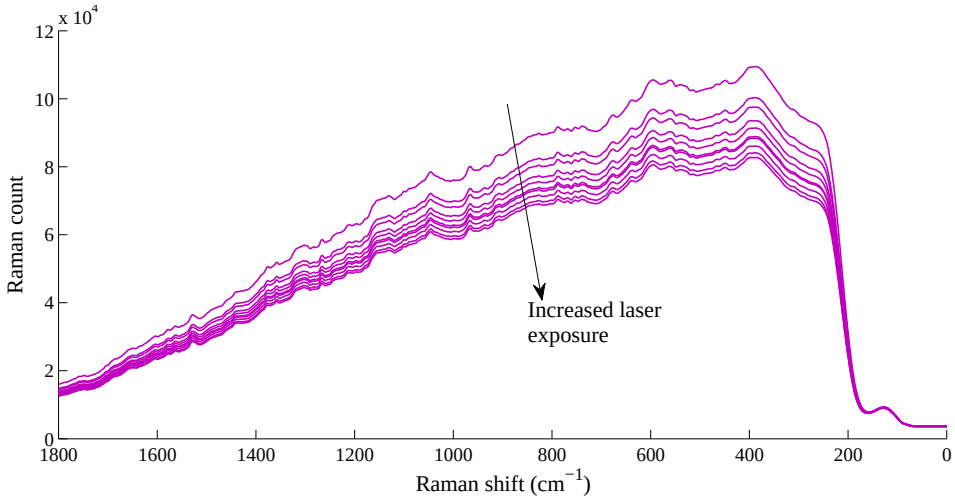

Figure 4.5: Raw Raman spectra of algal biomass powder. Spectra were collected one after the other, increasing at each collection the total exposure time to the laser. Background fluorescence decreases with increasing exposure time.

Signal processing techniques facilitate the extraction of meaningful information out of the spectra. For instance, baseline removal and spectral normalization enhance several spectral features, allowing the identification of some of the features as associated to the biochemical composition of the culture. For example, the peaks around 1440 and 1655 cm^{-1} are related to the oil content in the cells. Similarly, the peaks around 423, 516, 900, and 1360 cm^{-1} were found to be dependent on the glucose concentration in the culture media at a 95% confidence level. The peaks

identified here are in concordance with previously reported values for pure media components and for oil (Alfano et al., 2008; Zou et al., 2009).

4.5.1 Effect of Processing on Correlation Coefficient

The Raman spectra may be affected by the physical and chemical properties of the sample matrix as well as various other unknown disturbances in the system (Afseth et al., 2006). Signal preprocessing is performed to remove the effect of noise while retaining the maximum amount of information. Even though the primary objective of a preprocessing method is removal of noise, it cannot be guaranteed that all the information in the signal is retained. Therefore, an appropriate preprocessing technique needs to be chosen to de-noise the spectra and retain most of the information.

The R^2 values for the calibration dataset, for different preprocessing techniques, are shown in Table 3. Overall, it can be said, based on the R^2 values, that the SNV transformation provides the best model for all the three variables of interest. The Savitzky-Golay (SG) filter, however, produced a marginally better model in the case of glucose. Whether this is due to a structural reason in terms of spectral characteristics associated with glucose or due to differences in the nature of the error of the off-line measurements, was not investigated. Nevertheless, it is relevant to highlight that the intensity of the spectral bands that are due to the glucose molecule was lower than the intensity of those peaks associated with the oil and biomass. Furthermore, the fluorescence background due to the algal cells almost hid the presence of glucose peaks. A technique, such as SG filtering, that reduces the noise in the spectra while preserving the peak features might, in the case of glucose, be more suitable than one that scales up all the spectra.

Table 3 indicates that the combination of preprocessing techniques, generally, show a poor performance (lower R^2 value) than the individual techniques. The loss of information that results when multiple preprocessing methods are used for noise reduction proves to be costly during the model building procedure.

Table 4.3: R^2 value of the calibration dataset for different preprocessing techniques. The cells in the table highlighted in **bold** indicate the preprocessing techniques chosen for each of the three components.

	R^2 values for calibration dataset		
Preprocess	Biomass	Glucose	%Oil
	Datasets 1,2, & 3	Only Dataset 3	Only Dataset 3
SG	0.9950	**0.9956**	0.8741
Polyfit	0.9932	0.9878	0.9978
SNV	**0.9975**	0.9926	**0.9985**
SNV&Polyfit	0.9971	0.9602	0.9965
SG&SNV	0.9972	0.9905	0.9799
SG&SNV&Polyfit	0.9967	0.9880	0.9981

4.5.2 Model Validation

To assess the performance of the model built using SVR, the measured concentrations (for biomass, glucose, and oil content) are compared with the predicted concentrations. Plots of measured versus predicted concentrations of biomass, glucose, and oil content are shown in Figures 4.6, 4.7, and 4.8.

Figure 4.6 shows the measured versus predicted values for the biomass concentration. The standard normal variate transformation was used for preprocessing the Raman spectra, as it provides the highest R^2 value for calibration, as per Table 4.3. Datasets DS1, DS2, and DS3 were used for model building and validation, as indicated in Section 4.4.4.

The correlation coefficient, R^2, for the validation dataset between the measured and the predicted biomass concentrations was 0.9822 (from Figure 4.6), which is comparable with the R^2 value of 0.9975 obtained for the calibration dataset. This indicates that the RBF-kernel based support vector regression is a satisfactory method for sensor development, as it gave a correlation coefficient close to unity for both the calibration and validation datasets. Also, the comparable correlation coefficient for both datasets implies that there was insignificant model overfitting.

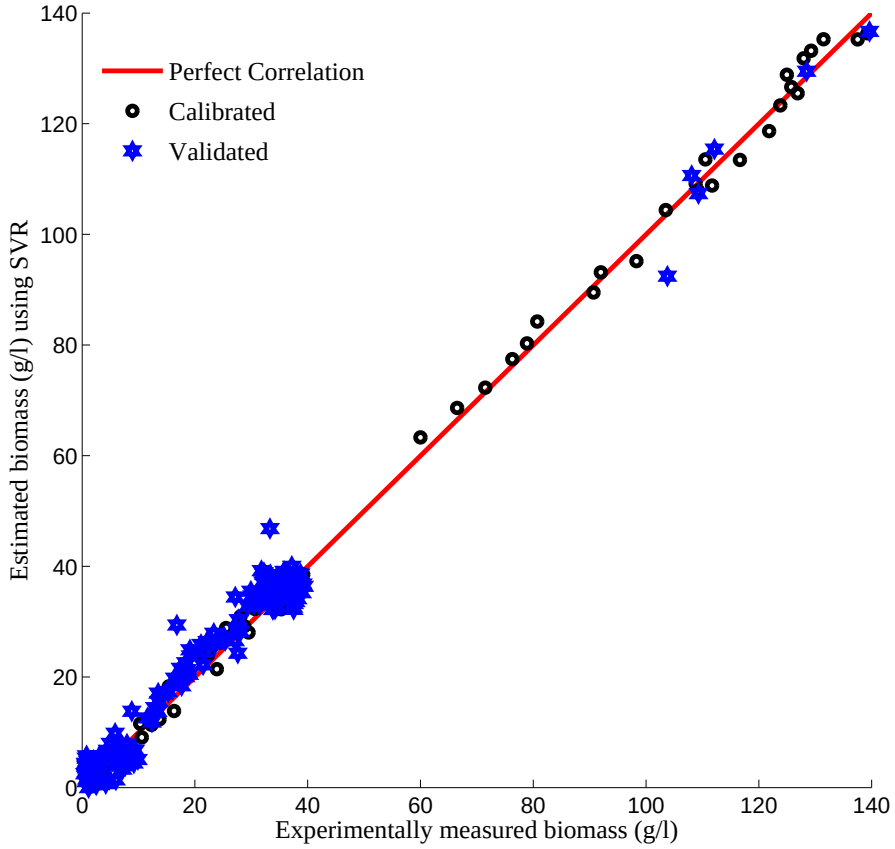

Figure 4.6: Measured versus predicted biomass concentration using the standard normal variate transformation for preprocessing the Raman spectra. RMSE value: 3.51 (R^2 value: 0.9822)

The developed sensor is robust for the full range of the biomass concentrations ($0.50 - 144.29$ g/L) considered in this study. The performance of the method is quite remarkable, given that the algal bioreactors are complex systems with undefined chemical composition and, in addition, the algal cells change their chemical composition along a single batch. These complexities in the sample matrix introduce unknown interferences in the Raman spectra. Figure 4.7 shows the measured versus predicted values for the glucose concentration. For preprocessing the Raman spectra, the Savitzky-Golay filtering was used for building the model for glucose concentration and only dataset DS3 was used for model building and validation, as indicated in Section 4.4.4.

The correlation coefficient value for the validation dataset between the measured and the predicted glucose concentrations was 0.8081, which is quite satisfactory but not very close to the R^2 value of 0.9956 obtained for the calibration dataset. This indicates that there is significantly more overfitting in the glucose model than in the case of biomass. Nonetheless, the predictive capability of the model built for the glucose concentration is fairly good, based on both the R^2 value and the observations in Figure 4.7.

The lower prediction accuracy of the glucose model could be due fewer number of measurements available for model building. More experimentally measured glucose concentrations (decoupled with biomass concentration) could prove beneficial in the model building exercise. These data could be obtained with cells growing under nitrogen limited conditions which favour the conversion of glucose to bio-oil rather than to biomass.

Figure 4.8 shows the measured versus predicted concentration curve for the oil content. The standard normal variate transformation was used for preprocessing the Raman spectra. Only dataset DS3 is used for model building and validation, as indicated in Section 4.4.4.

In Figure 4.8, it can be seen that the predicted oil content has a positive correlation with respect to the experimental measurement. The correlation coefficient (R^2 = 0.6422) for the validation dataset, however, was significantly lower than that of the calibration dataset. The lower correlation coefficient implies that caution

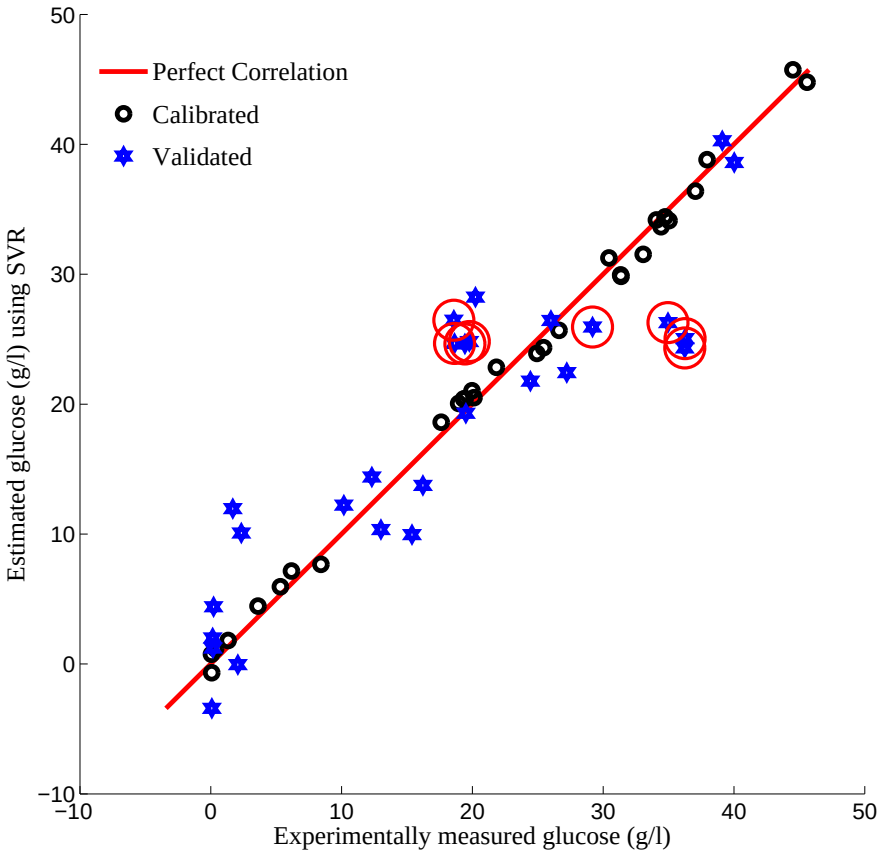

Figure 4.7: Measured versus predicted glucose concentration using Savitzky-Golay filtering for preprocessing the Raman spectra, for DS3. The circles (red in the online version) marked around the points correspond to the measurements in the initial lag phase (these are initial measured values; see the text in Section 5.4 for details) which are predominantly outliers. Excluding the outliers the RMSE value for predicted glucose concentration reduced from 5.64 to 4.31. (R^2 value: 0.8081 to 0.8867)

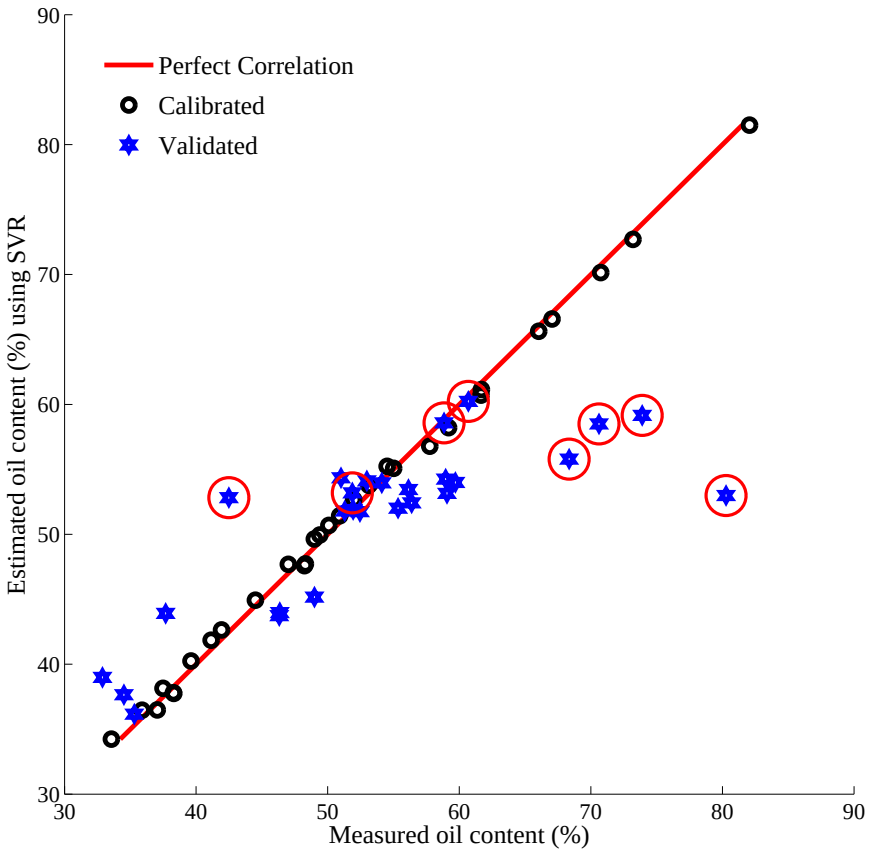

Figure 4.8: Measured versus predicted oil content using standard normal variate transformation for preprocessing the Raman spectra, for DS3. Again, the marked circles (red in the online version) corresponded to the measurements in the initial lag phase (these are initial measured values; see the text in Section 5.4 for details) which are predominantly outliers. Excluding the outliers the RMSE value for predicted oil content reduced from 7.56 to 3.63.(R^2 value: 0.6422 to 0.8597)

should be exercised when using Raman spectroscopy for estimation of oil content in microalgae.

The low correlation coefficient for the validation dataset for oil might be due to noise in the experimental measurements. Oil content quantification can be performed using several different techniques (Chen et al., 2009; Wawrik and Harriman, 2010; Halim et al., 2011). However, there is no one widely accepted standard method, due to complexity of all methods and the relatively high error (Lee et al., 1998; Christie, 1993). Lee et al. (1998) compared the estimated oil content in microalgae using different extraction solvents and cell disruption systems and found up to 50% relative difference in the estimated oil content. Likewise, Chen et al. (2009), compared the relative error of fluorescence and gravimetric based oil content determination method and found a standard deviation of approximately 5 %. The relative error associated to this standard deviation for the sample reported by Chen et al. (2009) is 25%, at the 95% confidence level. In general, all the existing experimental procedures for measuring oil are prone to high error. In this work, as mentioned in Section 4.4.1, the average standard error for the measurements was 5.6%, which corresponds to an average relative error of 22.43% at the 95% confidence level.

The average relative error for the validation dataset using the Raman-based method was 8.9%, assuming that the calibration measurements are free of error. This value is lower than the average relative error associated with the Nile Red based measurements obtained in this work, and to the reported error for both gravimetric and fluorescence based oil quantitation methods. Consequently, the support vector Raman spectroscopy based sensor can at least be considered at the same level of accuracy as the existing experimental procedures. In order to improve the support vector regression model performance, it would be required to reduce the relative error in the calibration dataset. It is expected that, by using a calibration dataset generated with a more precise oil measuring technique, a higher correlation coefficient can be achieved.

In summary, the proposed method was able to satisfactorily predict the three main components in the algal bioreactor, namely, biomass, glucose, and oil content,

within the normal error bounds. However, there is still scope for improvement, particularly in the case of glucose concentration and oil content. Performing more experiments, in which the concentration measurements of the three predicted components (biomass, glucose and oil content) could lead to an improved sensor building. Improving the signal to noise ratio in the Raman spectrometer, could also have a positive effect on the accuracy of the measurements.

4.5.3 Comparative study with other statistical methods

A comparative study is performed to illustrate the advantage of the applied support vector regression method over other statistical methods including principal components regression (PCR), partial least squares (PLS) regression, and kernel principal components regression (KPCR) for building an online monitoring sensor. Both PCR and PLSR techniques are used to convert a set of highly correlated variables to a set of independent variables by using linear transformations, applying feature reduction for large datasets. Kernel PCR is a nonlinear extension of the principal component regression method. The KPCR method uses the same principle referred to as the “kernel” trick, as mentioned in Section 3.1.

From Table 4.4, it can be seen that all of the four techniques have a comparably low root mean square error (RMSE) for the biomass concentration. For glucose composition and oil content, however, the performances of the PCR, the PLS, and the KPCR methods are significantly poorer when compared to the SVR method, as seen by their higher root mean squares prediction errors (RMSE). In general, for all components, the SVR method shows a consistently superior performance when compared to the other three methods.

4.5.4 On-line Estimation of the Compositions in the Bioreactor

In Section 5.2, Raman spectra was successfully correlated with the concentrations of biomass, glucose, and oil content in the cells. In this section, the use of Raman spectroscopy as an on-line, real-time multivariate sensor is tested for microalgal

Table 4.4: Root mean square error (RMSE) value for the different statistical techniques used. The cells in the table highlighted in **bold** indicate the method chosen for each of the three components.

	RMSE values for prediction estimates		
Sensors	Biomass SNV	Glucose SG	%Oil SNV
PCR	4.07	9.56	9.07
PLS	4.31	9.58	10.0
Kernel PCR	3.80	10.9	8.02
Nonlinear SVR	**3.51**	**5.64**	**7.56**

applications. For this purpose, the green microalgae *A. protothecoides* was cultured in a 2L bioreactor, with the Raman probe inserted in the reactor to collect the spectra *in situ*. Chemical composition was estimated using the proposed SVR sensor. The dataset DS3 was created using off-line measurements from this experiment.

Raman spectra was collected every ten minutes with an integration time equal to 20 seconds. The average time required for transforming the information from spectra to chemical composition was 0.058 seconds. The average total time, including spectra collection, was around 20 seconds with the prediction of the composition taking an insignificant amount of time compared to the spectral integration time. The estimation of chemical properties of an algal culture is, therefore, solely determined by the integration time. Compared to the algal culture dynamics, which can be of the order of hours or sometimes days, the prediction time (around 20 seconds) is insignificant. Therefore, Raman spectra can be used for real time on-line estimation of the composition in the algal bioreactors.

The predicted profiles for biomass, glucose, and oil content are shown in Figures 4.9, 4.10, and 4.11 respectively. For comparison purposes, the off-line experimental measurements are also included in the plots. There is a good match between the Raman spectra based predictions and the experimental measurements for the full range of concentrations.

During the initial 50 hours of culture time, the variance between contiguous Raman

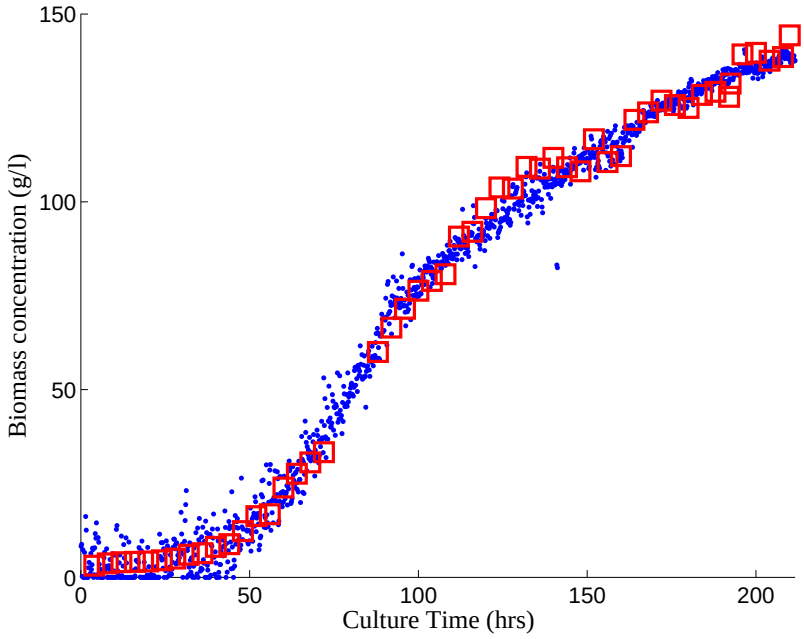

Figure 4.9: Biomass concentration profile for an algal culture: (·) support vector Raman spectroscopy -based measurement; (□) Off-line experimental measure

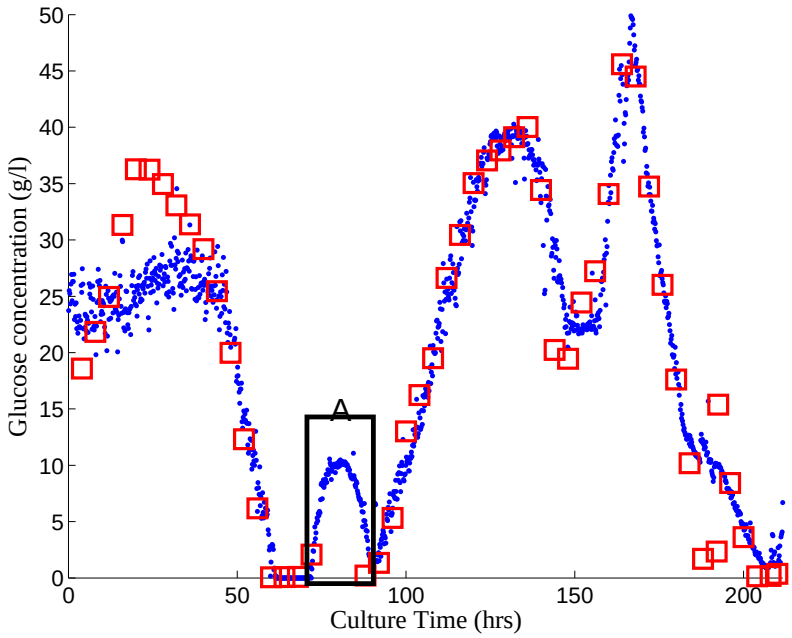

Figure 4.10: Glucose concentration profile for an algal culture: (·) support vector Raman spectroscopy-based measurement; (□) Off-line experimental measure. For explanation regarding measurements enclosed in box 'A' refer to the text in Section 3.5.4

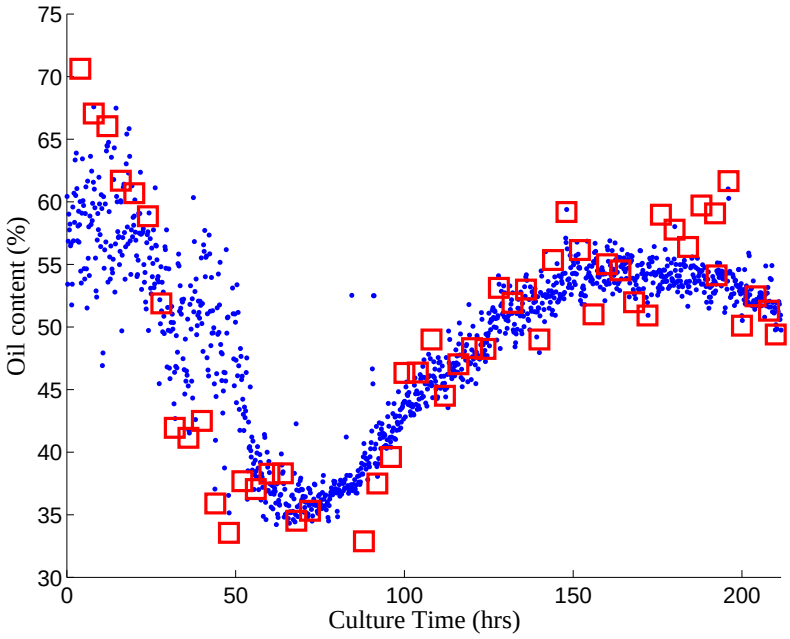

Figure 4.11: Profile for the oil content in the algal cells: (·) support vector Raman spectroscopy-based measurement; (□) Off-line experimental measure

based predictions was higher than in the subsequent culture times. This indicates that there are significant interferences from the sample matrix at the start-up of the culture. At the start of the culture (lag phase), there are important changes in both the chemical composition of the culture media and the biochemical composition of the algal cells. Rapid changes in the concentration of trace elements in the initial culture medium, cell size and morphology, and cell pigmentation could be responsible for the high variance observed in the initial spectra measurements. Therefore, the spectral based estimations should be used with utmost precaution during the lag phase. It is suggested that a moving average window be used to reduce the fluctuation in the predictions. In Figures 4.7 and 4.8 the data points corresponding to the lag phase are circled. It can be seen that the deviation between the off-line measures and the estimated values is significantly higher for this subset of the data than for the other data points. Ignoring the lag-phase data, the accuracy of the prediction is improved as indicated by a reduction in the RMSE values (and an increase in the R^2 values).

From Figure 4.10, it can be seen that glucose estimates (after lag phase) based on Raman spectroscopy have a smoother profile than the experimental measurements. Although, experimental measurements for glucose based on HPLC have in general a high precision, the samples drawn from the reactor and analyzed in the HPLC might not be representative of the bioreactor contents. This is because, the conditions in the sampling line may not be the same as the conditions in the reactor. Furthermore, the sample obtained from the reactor might undergo changes during the time lapsed for preparing the sample for HPLC and other analysis. These could lead to reduced accuracy and reliability of the off-line measurements.

An additional advantage of the on-line Raman based method is that the composition measurements can be taken at a considerably higher frequency compared to the off-line experimental measurements, given that the Raman based method does not require the removal of a sample from the reactor. The reduced frequency for the removal of a sample, in turn, reduces the chance of contamination from faster growing bacteria and fungi. A higher measurement frequency helps in observing the changes in the composition that will otherwise be overlooked. For example,

the variation in the glucose concentration between approximately 75 and 90 hours (enclosed in box ‘A’ in Figure 4.10) is not readily apparent from the off-line experimental measurements. Whereas, the Raman spectroscopic method is able to clearly identify these changes.

From Figure 4.11, it can be seen that the variance of contiguous oil estimates using the Raman spectra is lower compared to the one obtained using off-line measurements. This implies that the estimates provided by the Raman spectra are more reliable than the off-line experimental measurements.

Therefore, it can be concluded that Raman spectroscopy can be used for predicting the glucose and oil estimates after the lag phase. From Figure 4.9, it can be seen that for biomass estimates, however, Raman spectroscopy can be used for the entire range of culture times (including the lag phase).

4.6 Conclusion

Appropriate monitoring and control of culture conditions in microalgal bioreactors are required in order to maximize oil productivity. Raman spectra in combination with support vector regression can be used for building a multivariate sensor for the online-monitoring of the concentrations of the three main components in the bioreactor, namely, biomass, glucose, and oil content in the cells. In heterotrophic algal cultures, the substrate (glucose) concentration is usually the control variable. Therefore, a control law can be defined for optimizing the concentration of biomass and the oil content.

The advantages of the proposed online sensor include: a reduction in the time taken to obtain an estimate of the biochemical composition of the system and thereby enabling the use of several well known control strategies; a smaller variance in the oil estimates was observed with Raman based measurements compared to the off-line measurements; and a solution for the problem of disparity between the measured sample and the reactor contents is achieved, thus providing a more reliable measurement than traditional off-line analysis.

The effect of preprocessing techniques including Savitzky-Golay filtering, baseline correction, and standard normal variate transformation on the model building exercise was studied. Standard normal variate is the most suitable preprocessing technique for estimating biomass concentration and the oil content. Similarly for a suitable estimation of glucose concentration, it is necessary to use the Savitzky-Golay filter.

Acknowledgements

The authors gratefully acknowledge the financial support provided by Canada's Natural Sciences and Engineering Research Council (NSERC), and Alberta Innovates Technology Futures.

References

Y. Chisti, Biodiesel from microalgae, Biotechnol. Adv. 25 (2007) 294–306.

A. Singh, P. S. Nigam, J. D. Murphy, Renewable fuels from algae: An answer to debatable land based fuels, Bioresour. Technol. 102 (2011) 10–16.

W. Xiong, X. Li, J. Xiang, Q. Wu, High-density fermentation of microalga *Chlorella protothecoides* in bioreactor for microbio-diesel production, Appl. Microbiol. Biotechnol 78 (2008) 29–36.

J. Liu, J. Huang, Z. Sun, Y. Zhong, Y. Jiang, F. Chen, Differential lipid and fatty acid profiles of photoautotrophic and heterotrophic *Chlorella zofingiensis*: Assessment of algal oils for biodiesel production, Bioresour. Technol. 102 (2011) 106–110.

Y. Liang, N. Sarkany, Y. Cui, Biomass and lipid productivities of *Chlorella vulgaris* under autotrophic, heterotrophic and mixotrophic growth conditions, Biotechnol. Lett. 31 (2009) 1043–1049.

H. De la Hoz, A. Ben-Zvi, R. Burrell, W. McCaffrey, The dynamics of heterotrophic algal cultures, Bioresour. Technol. 102 (2011) 5764–5774.

S. Marose, C. Lindemann, T. Scheper, Two-dimensional fluorescence spectroscopy: a new tool for on-line bioprocess monitoring, Biotechnology Progress 14 (1998) 63–74.

E. Skibsted, C. Lindemann, C. Roca, L. Olsson, On-line bioprocess monitoring with a multi-wavelength fluorescence sensor using multivariate calibration, Journal of Biotechnology 88 (2001) 47 – 57.

D. Landgrebe, C. Haake, T. Höpfner, S. Beutel, B. Hitzmann, T. Scheper, M. Rhiel, K. Reardon, On-line infrared spectroscopy for bioprocess monitoring, Applied Microbiology and Biotechnology 88 (2010) 11–22.

K. S. Y. Yeung, M. Hoare, N. F. Thornhill, T. Williams, J. D. Vaghjiani, Near-infrared spectroscopy for bioprocess monitoring and control, Biotechnology and Bioengineering 63 (1999) 684–693.

M. Dabros, M. Amrhein, D. Bonvin, I. W. Marison, U. von Stockar, Data reconciliation of concentration estimates from mid-infrared and dielectric spectral measurements for improved on-line monitoring of bioprocesses, Biotechnology Progress 25 (2009) 578–588.

Y. Y. Huang, C. M. Beal, W. W. Cai, R. S. Ruoff, E. M. Terentjev, Micro-Raman spectroscopy of algae: Composition analysis and fluorescence background behavior, Biotechnol. Bioeng. 105 (2010) 889–898.

T. Shope, T. J. Vickers, C. Mann., The direct analysis of fermentation products by Raman spectroscopy., Appl. Spectrosc. 41 (1987) 908–912.

Y. Xu, J. F. Ford, C. K. Mann, T. J. Vickers, Raman measurement of glucose in bioreactors materials, Proc. SPIE 2976 (1997) 10–19.

A. D. Shaw, N. Kaderbhai, A. Jones, A. M. Woodward, R. Goodacre, J. J. Rowlan, D. B. Kell, Noninvasive, on-line monitoring of the biotransformation by yeast

of glucose to ethanol using dispersive Raman spectroscopy and chemometrics., Appl. Spectrosc. 53 (1999) 1419–1428.

C. Cannizaro, M. Rhiel, I. Marison, U. von Stockar, On-line monitoring of *Phaffia rhodozyma* fed-batch process with in situ dispersive Raman spectroscopy., Biotechnol. Bioeng. 83 (2003) 668–680.

H. L. T. Lee, P. Boccazzi, N. Gorret, R. J. Rama, A. J. Sinskey, In situ bioprocess monitoring of *Escherichia coli* bioreactions using Raman spectroscopy, Vib. Spectrosc. 35 (2004) 131–137.

H. Wu, J. V. Volponi, S. Singh, Single-cell diesel mining on microalgae: Direct and quantitative monitoring of microalgal oil production in vivo by Raman spectroscopy, Biophys. J. 98 (2010) 744a.

A. Abbas, M. Josefson, K. Abrahamsson, Characterization and mapping of carotenoids in the algae *Dunaliella* and *Phaeodactylum* using Raman and target orthogonal partial least squares, Chemom. Intell. Lab. Syst. 107 (2011) 174 – 177.

F. Estienne, D. L. Massart, Multivariate calibration with Raman data using fast principal component regression and partial least squares methods, Anal. Chim. Acta 450 (2001) 123–129.

M. J. Goetz, G. L. Coté, R. Erckens, W. March, M. Motamedi, Application of a multivariate technique to Raman spectra for quantification of body chemicals, IEEE T. Bio-Med. Eng. 42 (1995).

U. Thissen, M. Pepers, B. Üstün, W. J. Melssen, L. M. C. Buydens, Comparing support vector machines to PLS for spectral regression applications, Chemom. Intell. Lab. Syst. 73 (2004) 169–179.

M. G. Kendall, A course in multivariate analysis, Girffin, London, 1957.

H. Hotelling, The relations of the newer multivariate statistical methods to factor analysis, Brit. J. Stat. Psy. 10 (1957) 69–79.

R. Marbach, H. M. Helse, Calibration modeling by partial least-squares and principal component regression and its optimization using an improved leverage correction for prediction testing, Chemom. Intell. Lab. Syst. 9 (1990) 45–63.

T. Naes, H. Martens, Principal component regression in NIR analysis: Viewpoints, background details and selection of components, J. Chemom. 2 (1988).

M. Blanco, J. Coello, H. Iurriaga, S. Maspoch, J. Riba, E. Rovira, Kinetic spectrophotometric determination of Ga(III)-Al(III) mixtures by stopped-flow injection analysis using principal component regression, Talanta 40 (1993) 261–267.

Y. Tan, L. Shi, W. Tong, C. Wang, Multi-class cancer classification by total principal component regression (TPCR) using microarray gene expression data., Nucleic Acids Res. 33 (2005) 56–65.

H. Wold, Estimation of principal components and related models by iterative least squares, New York: Academic Press, 1966.

M. Sjostrom, S. Wold, W. Lindberg, J. A. Persson, H. Martens, A multivariate calibration problem in analytical chemistry solved by partial least squares models in latent variables, Anal. Chim. Acta 150 (1983) 61–70.

S. Wold, M. Sjostrom, L. Eriksson, PLS-regression: a basic tool of chemometrics, Chemom. Intell. Lab. Syst. 58 (2001) 109–130.

A. R. McIntosh, N. J. Lobaugh, Partial least squares analysis of neuroimaging data: applications and advances, NeuroImage 23 (2004) S250–S263.

B. S. Dayal, J. F. MacGregor, Recursive exponentially weighted PLS and its applications to adaptive control and prediction, J. Process Control 7 (1997) 169–179.

A. Demiriz, K. P. Bennet, C. M. Breneman, M. J. Embrechts, Support vector regression in chemometrics, Comput. Sci. Stat.: Proc. 33rd Symposium on Interface (2001).

S. Khatibisepehr, B. Huang, F. Ibrahim, J. Xing, W. Roa, Data-based modeling and prediction of cytotoxicity induced by contaminants in water resources, Comput. Biol. Chem. 35 (2011) 69–80.

I. Barman, C. R. Kong, N. C. Dingari, R. R. Dasari, M. S. Feld, Development of robust calibration models using support vector machines for spectroscopic monitoring of blood glucose, Anal. Chem. 82 (2010).

B. Scholkopf, A. J. Smola, Learning with kernels, MIT press; Cambridge, 2002.

B. Boser, I. Guyon, V. Vapnik, A training algorithm for optimal margin classifiers, Proc. 5th Annu. ACM Workshop Comput. Learn. Theor. (1992) 144–152.

S. B. Chitralekha, S. L. Shah, Application of support vector regression for developing soft sensors for nonlinear processes, Can. J. Chem. Eng. 88 (2010) 696–709.

I. B. Khediri, C. Weihs, M. Limam, Support vector regression control charts for multivariate nonlinear autocorrelated processes, Chemom. Intell. Lab. Syst. 103 (2010) 76–81.

K. Xu, L. Wencong, J. Shengli, L. Yawei, C. Nianyi, Support vector regression applied to materials optimization of sialon ceramics, Chemom. Intell. Lab. Syst. 82 (2006) 8–14.

U. Thissen, R. van Brakel, A. P. de Weijer, W. J. Melssen, L. M. C. Buydens, Using support vector machines for time series prediction, Chemom. Intell. Lab. Syst. 69 (2003) 35–49.

V. N. Vapnik, Statistical learning theory, John Wiley and Sons, New York, 1998.

M. Pontil, S. Mukherjee, F. Girosi, On the noise model of support vector machines regression, Technical Report, MIT (2000).

J. T. Kwok, I. W. Tsang, Linear dependency between ε and the input noise in ε-support vector regression, IEEE T Neural Network 14 (2003) 544–553.

A. Hara, N. S. Radin, Lipid extraction of tissues with a low-toxicity solvent, Anal. Biochem. 90 (1978) 420–426.

K. Surisetty, H. De la Hoz, W. C. McCaffrey, A. Ben-Zvi, Robust modeling of a microalgal heterotrophic fed-batch bioreactor, Chem. Eng. Sci. 65 (2010) 5402–5410.

N. K. Afseth, V. H. Segtnan, J. P. Wold, Raman spectra of biological samples: A study of preprocessing methods, Appl. Spectrosc. 60 (2006) 1358–1367.

F. T. Chau, Y. Z. Liang, J. Gao, X. G. Shao, Chemometrics, John Wiley and Sons, New Jersey, 2004.

H. Martens, T. Naes, Multivariate calibration, John Wiley and Sons, Chichester, 1989.

W. S. Cleveland, Robust locally weighted regression and smoothing scatterplots, J. Am. Stat. Assoc. 74 (1979) 829–836.

W. S. Cleveland, S. J. Devlin, Locally weighted regression: An approach to regression analysis by local fitting, J. Am. Stat. Assoc. 83 (1988) 596–610.

T. J. Hastie, R. J. Tibshirani, Generalized additive models, Stat. Sci. 1 (1986) 297–310.

C. W. Hsu, C. C. Chang, C. J. Lin, A practical guide for support vector classification, Technical Report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei., http://www.csie.ntu.edu.tw/ cjlin/libsvm/, 2003.

R. Alfano, W. Wang, A. Doctore, Detection of glucose levels using excitation and difference Raman spectroscopy at the IUSL, City University of New York, New York (2008).

M. Q. Zou, X. F. Zhang, X. H. Qi, H. L. Ma, Y. Dong, C. W. Liu, X. Guo, H. Wang, Rapid authetication of olive oil adulteration by Raman spectrometry, J. Agric. Food Chem. 57 (2009) 6001–6006.

W. Chen, C. Zhang, L. Song, M. Sommerfeld, Q. Hu, A high throughput Nile red method for quantitative measurement of neutral lipids in microalgae, J. Microbiol. Methods 77 (2009) 41–47.

B. Wawrik, B. H. Harriman, Rapid colorimetric quantification of lipid from algal cultures, J. Microbiol. Methods 80 (2010) 262–266.

R. Halim, B. Gladman, M. K. Danquah, P. A. Webley, Oil extraction from microalgae for biodiesel production, Bioresour. Technol. 102 (2011) 178–185.

S. J. Lee, B. D. Yoon, H. M. Oh, Rapid method for the determination of lipid from the green alga Botryococcus braunii, Biotechnol. Tech. 12 (1998) 553–556.

W. W. Christie, Preparation of lipid extracts from tissues, Oily Press, Dundee, 1993.

5

Identifying Candidate Biomarkers for Early Detection of Heart Transplant Rejection Using Real Time Reverse Transcription Polymerase Chain Reaction (RT-PCR)

5.1 Introduction

Organ transplantation is one of the rapidly developing fields in biomedical studies. Graft rejection remains to be a major barrier in organ transplantation. The rejec-

tion process involves an immune response against the foreign tissue antigens. An early detection of rejection is mandatory to effectively treat and prevent cardiac dysfunction (Morris and Delves, 1998; Dallman and Delves, 1998). Assessment of gene expression levels has produced insights for identification of allograft rejection (Erickson et al., 2001, 2004, 2003). Gene expression microarrays emerged as a important tool in the 1990s for measuring the gene expression levels of protein coding mRNA transcripts within a tissue (Schena et al., 1995). Over the past 20 years they have become the dominant source used in transcriptomics, the study of said mRNA transcripts. Researchers often compare expression levels of mRNAs across different types of tissues to find biomarkers that are differentially expressed, i.e. they produce different amounts of mRNA. The theory is based on the assumption that different levels of mRNA in the tissues cause a similar difference in the amount of proteins produced (Quackenbush, 2002). Differing levels of protein can in turn lead to, or indicate the manifestation of, sickness, disease or damage to the tissue. Knowing which biomarkers are differentially expressed is of great importance to many applications: Pharmaceutical companies could develop drugs that target these biomarkers (Walker., 2001). In clinical settings these biomarkers could be used in diagnostic systems to aide doctors and clinicians (Mueller et al., 2007). They could also be used as good starting points for future research in biology (Schena et al., 1996; Carulli et al., 1998).

Much research effort has focused on identifying differentially expressed genes (candidate biomarkers) from microarray datasets (Li et al., 2002; Miller et al., 2003). Researchers use class comparisons analysis to obtain lists of genes and gene sets that are differentially expressed between the classes of interest. To validate the results a secondary analysis with a more accurate technology, such as northern blotting or real-time polymerase chain reaction (RT-PCR) is used (Chuaqui et al., 2002). RT-PCR method is faster and more robust to small changes in expression. Furthermore, the microarray datasets are highly corrupted with noise and higher reliability of RT-PCR measurements provides a robust identification of candidate biomarkers (Allanach et al., 2008).

In this study, a novel procedure for obtaining candidate biomarkers from a time

series RT-PCR data for early detection of heart allograft rejection. To the best knowledge of the author and collaborators, this is first such study of identification of biomarkers using time series data. These chosen biomarkers were validated by applying the k-means clustering algorithm on various independent renal allograft microarray datasets obtained from the ncbi-geo website. Hypothesis testing is a commonly used technique in statistical analysis that can be used for making decisions on the data. A test of hypotheses (test procedure) is a method for using sample data to decide between two competing claims. Hypothesis testing has found applications in the field of systems biology (Venkat et al., 2011) and organ transplantation (Paya et al., 2004). Likewise, clustering algorithms have been applied to analyze gene expression datasets in diverse biomedical applications including skin biopsies (Whitfield et al., 2003), diabetes (Koulmanda et al., 2008), and kidney transplant rejection (Flechner et al., 2004). Horwitz et al. (2004) has used hierarchical clustering to demonstrate the ability of their chosen candidate markers to distinguish control, rejection, and post rejection samples. The k-means clustering algorithm, used in this study, has found application in structure identification of the dataset and recognizing any potential mislabeling in post-operative liver transplant monitoring (Melvin et al., 1997).

In this study, the syngeneic and allogeneic heart transplant patients were used to differentiate the innate from the adaptive immune response that helps in identifying robust markers of rejection. Additionally, the work considers the identified robust markers of rejection from heart transplant animal models as a precursor to working with human data for use in medical problems.

5.2 Methods and Materials

5.2.1 Study Design

Serial changes in the transcript levels of 82 genes were analyzed by real-time reverse transcriptase polymerase chain reaction (RT-PCR) at 14 different, non-equally spaced time points, $t \in t_1 < t_2 < \ < t_{14}$ (refer to Appendix A for the genes and

actual times when expression values were measured) during the first 7 days after allogeneic and syngeneic murine heart transplantation.

Mice: Eight to 12 wk old male BALB/cByJ (BALB/c) (H-2d), C57BL/6J (B6) (H-2b) mice were obtained from Jackson Laboratory (JAX, Bar Harbor, Maine) and housed under standard conditions in a pathogen free facility.

Transplant model: Heart grafts were transplanted in a heterotopic cardiac transplant model as previously described (Corry et al., 1973). Briefly, hearts were harvested from freshly sacrificed donors and immediately transplanted into recipients anaesthetized via intra peritoneal injection with 60 mg/kg of pentobarbital sodium. The donor aorta was anastomosed to the recipient abdominal aorta by end-to-side anastomosis. The donor pulmonary artery was anastomosed end-to-side to the recipient vena cava. All surgical procedures were completed in less than 60 minutes from the time that the donor heart was harvested. Donor hearts that did not beat immediately after reperfusion or stopped within 1 day following transplantation were excluded ($>$ 95% of all grafts functioned at day 1 following transplantation). The recipient's native heart was not surgically manipulated and remained functional. Donor allograft hearts were harvested immediately after transplantation (0 time point) and at 1, 3, 6, 9, 12, 15, 18, 21 hours and at 1, 2, 3, 4, 5, 6 and 7 days following transplantation. Three BALB/c and three B6 un-transplanted hearts served as controls. The allografts were divided into equal sections for extraction of RNA and tissue sections for histology. Altogether 99 hearts (93 transplant, 6 control hearts) were harvested and analyzed. In the allogeneic transplant model BALB/c donor hearts were transplanted into B6 recipients (BALB/c into B6), in the syngeneic transplant model B6 donor hearts were transplanted into B6 recipients (B6 into B6) to analyze the innate response.

A-priori gene selection:

82 genes were selected for the kinetic analysis based on prior microarray studies (Mueller et al., 2003). All genes were manually classified according to the biological processes they contribute to using the gene ontology GO annotation system

(Ashburner et al., 2000). If a gene contributed to more than one distinct biological process, the most appropriate for the current experimental settings was chosen. 12 functional classes were defined and further grouped into immune-related and non-immune related gene sets. Information regarding the 82 individual genes, the 12 designated biological classes they are assigned to, their Gene ID and GenBank numbers, gene symbols, gene names and the sequences of the primer pairs used are presented in supplementary material of Mueller et al. (2003)'s work.

5.2.2 Applied method of transcript measurements

Real-time RT-PCR: Primer pairs were designed using Primer Express software (Applied Biosystems, Foster City, CA). Forward (FW) and reverse (RE) primers were chosen to have a length between 18 and 22 base pairs and designed to amplify an amplicon length of 51 base pairs. All primer pairs were tested in both immune-rich tissue samples and non-template controls for specificity, primer-dimer formation, and reproducibility.

RNA extracted from the individual tissue samples was analyzed individually to control for both technical and biological variability. Total murine RNA was isolated from three hearts per time-point using TRI Reagent (Sigma-Aldrich Corp., St. Louis, MO). All samples were treated with deoxyribonuclease to eliminate DNA (Deoxyribonuclease I, Amplification Grade, Invitrogen Life Technologies, Carlsbad, CA) contamination. 10 μg of RNA were reverse transcribed using SuperScript II RNase Reverse Transcriptase (Gibco, Carlsbad, CA). The single cDNA reaction product was aliquoted for the target and control amplifications. For all target primers the same cDNA sample was used.

The GeneAmp 5700 Sequence Detection System (Applied Biosystems, Foster City, CA) was used to perform RT-PCR using 250 ng of template cDNA, 5 μM of forward and reverse primer and 10 μL of 10X SYBR Green PCR Master Mix (Applied Biosystems, Foster City, CA) per well in a MicroAmp Optical 96-well reaction plate (Applied Biosystems, Foster City, CA). The gene-specific PCR products were continuously measured by the increase in fluorescence due to the binding of SYBR

Green to double-stranded DNA during 40 cycles. Dye ROX, included in the SYBR Green PCR Master Mix, served as a passive reference to normalize for non-PCR-related fluctuations in fluorescence signal.

Based on the constitutive expression across various experiments and further validated by the microarray results glyceraldehyde-3-phosphate dehydrogenase (Gapdh) was chosen as endogenous control for normalization.

The relative quantitation of the amount of gene target was based on the ΔC_T method [Manual GeneAmp 5700, Applied Biosystems, Forster City, CA]. The difference in the cycle threshold (C_T) value of each target gene is calculated relative to the C_T value of the endogenous reference Gapdh. The relative amount of target gene transcript is expressed as percentage of Gapdh, which is set to 100%. The quantities of the individual target gene in each experimental sample are expressed as n-fold difference relative to its quantity in the calibrator sample, i.e. the un-transplanted control hearts (BALB/c in the allogeneic experiments, B6 in the syngeneic). All real-time RT-PCR experiments were run in triplicate, analyzing samples from 3 animals per group.

Analysis of the transcript measurements:

At each time stamp, t, the transplantation procedure is replicated. The C_T values are obtained experimentally for all the replicates at these 14 time samples. The ΔC_T values for 85 genes for all replicates are calculated at these time instant $t_i = 1, 2, ..., 14$. Tables 1 to 16 in Appendix A, show the values for the genes at different times for isograft (syngeneic) and allograft (allogeneic) patients. In this work, the $\Delta\Delta C_T$ values are calculated and analyses were performed on them for early detection of allograft rejection. The definitions and the procedure for obtaining C_T, ΔC_T , and $\Delta\Delta C_T$ values are as follows (Thiel et al., 2002; Pfaffl, 2001):

- C_T**: The cycle values of the target gene.**
  The C_T value is the experimentally measured value for the transplantation patient.

- ΔC_T**: The difference in threshold cycles for target and reference.**
  The ΔC_T values are calculated by subtracting the C_T values of the target gene with respect to the house keeping gene, GAPDH. That is, for the k^{th} gene at time t_i, the corresponding ΔC_T value is given by the Equation 5.1

$$\Delta C_T(t_i,k) = C_T(t_i,k) - C_T(t_i,R) \tag{5.1}$$

  where $\Delta C_T(t_i,k)$, $C_T(t_i,k)$ are the ΔC_T, C_T values for k^{th} gene at time t_i respectively and $C_T(t_i,k)$ is the C_T value for the reference gene, in this case GAPDH, at time t_i.

- $\Delta\Delta C_T$**: The difference in normalized threshold cycles for experimental and calibrator sample** The $\Delta\Delta C_T$ values are calculated by subtracting the ΔC_T values of the experimental sample of a given gene with the mean ΔC_T values of the calibration sample of the same gene. For the k^{th} gene the corresponding $\Delta\Delta C_T$ value is given by the Equation 5.2

$$\Delta\Delta C_T(k) = \Delta C_T(k,e) - \bar{\Delta C}_T(k,c) \tag{5.2}$$

  where $\Delta C_T(k,e)$, is the ΔC_T value for the k^{th} gene for the experimental sample and $\bar{C}_T(k,c)$ is the mean ΔC_T value for the k^{th} gene for the calibration sample.

In the case of a missing ΔC_T values for a particular gene, for one of its replicates, an average value across the other replicates is chosen. For statistical analysis, the ΔC_T values for the genes at all time instants are required. Therefore, genes with zero ΔC_T values for all the replicates at any given time are removed from the analysis. The gene **FOLbp3** has a lot of missing ΔC_T values (refer Appendix A) and therefore is removed from analysis. Similarly, gene **CK** which has missing ΔC_T values on day 4 for all replicates is also removed (refer Appendix A). In the $\Delta\Delta C_T$ values, the impact of the house gene GAPDH is zero, and therefore is removed for the analysis.

5.3 Statistical Methods and Data Analysis

To identify and validate the potential biomarkers that can detect allograft at early stages, statistical analysis including hypothesis testing and k-means clustering are applied. Brief descriptions of the two methods are given below:

5.3.1 Hypothesis Testing:

A standard approach to the hypothesis testing problem consists of a series of steps given below:

Step 1. Research and define the test hypothesis along with choosing the variable to be used in sample data.

Step 2. State the null hypothesis (H_0) and the alternate hypothesis (H_1).

Step 3. Select the significance level for the test and decide which test is appropriate while stating the relevant test statistic T.

Step 4. Consider the statistical assumptions being made about the sample in doing the test.

Step 5. Compute all quantities appearing in the test statistic and then the value of the test statistic.

Step 6. Determine the p-value associated with the observed value of the test statistic.

Step 7. State the conclusion (which is to reject H_0 if p-value and not to reject H0 otherwise) as per the context of the problem and the level of significance.

There are two kinds of hypothesis testing that can be performed for identifying the differences between the two classes, namely, t-test which is a test statistic for the differences between the means of two distributions and F-test which specifically tests for difference in variances between the two classes. The procedure for obtaining both the allogeneic and syngeneic datasets is the same. The assumption that the

variances in both the allogeneic and syngenic datasets, caused due to errors in the measurements, is equal is a reasonable one. Therefore, t-test seems an appropriate method for comparing the means of the two distributions assuming that the variance of the two classes is known.

5.3.2 K-means clustering:

K-means Clustering is a method of cluster analysis (unsupervised classification) which aims to partition n observations into K clusters in which each observation belongs to the cluster with the nearest mean. K-means method constructs these partitions so that the squared Euclidean distance between any object and the centroid of its respective cluster is at least as small as the squared distances to the centroids of the remaining clusters. This procedure consists of the following steps (Ray and Turi, 1999; Steinley, 2006):

Step 1. Choosing the K initial cluster centers,$\mu_1^1, \mu_2^1, ..., \mu_K^1$.

Step 2. The squared Euclidean distance, , between the l^{th} object and the j^{th} cluster is obtained as shown:

$$d^2(l, j) = \sum_{j=1}^{K} (x_{lj} - \mu_j^{(1)})^2 \tag{5.3}$$

Objects are allocated to the cluster where 5.3 is minimum

Step 3. After initial object allocation, cluster centroid is obtained for each cluster, then objects are compared to each centroid (using $d^2(l, j)$) and moved to the cluster whose centroid is closest.

Step 4. New centroids are calculated with the updated cluster membership (by calculating the centroids after all objects have been assigned).

Step 5. Steps 2 and 3 are repeated until no objects can be moved between clusters.

The K value is obtained by prior knowledge of the number of clusters present in the data. In this work, only two clusters are considered, namely, allogeneic and syngeneic.

5.4 Results

5.4.1 Potential Biomarkers

Identification of potentially diagnostic genes

It is a well known biological fact, that prior to rejection (at initial time) both the allogeneic and syngeneic patients have the same gene expression value. Therefore, it is necessary to find biomarkers (genes) that do not indicate any inherent difference between the gene expression value for allogeneic and syngeneic patients at initial time (t = 0 hr). A pre-processing and feature selection procedure is required to obtain biomarkers which have similar gene expression values, for both classes, at the initial time (t = 0 hr) and significantly different gene expression values around the final time (t = 7 days).

In this work a new approach to obtain biomarkers is proposed which exploits the diverging trend between allogenic and syngeneic datasets with time for illustrating a transplant rejection. The pre-processing/feature selection technique involves a statistical comparison of the population means for the two classes of data (allogeneic and syngeneic), for each gene, at each time step. The genes/biomarkers which indicate that the allogeneic and syngeneic samples are from the same population at initial time and from differing populations at later time are chosen.

The hypothesis testing method using a student t-distribution was used to compare the two population means. The assertion that $\mu_1 - \mu_2$ (difference of the population means of the two classes) is equal to zero is the null hypothesis. The alternative hypothesis is that $\mu_1 - \mu_2$ is not equal to zero. In this study, a normal distribution will be assumed for the difference of the population means at varying significance level, α = 0.01 (99 % confidence level) to 0.99 (1 % confidence level) in increments of 0.01.

For each time t, the maximum % confidence level at which the two population means are statistically indistinguishable, is calculated. The biomarkers (genes) that show a higher confidence level for similarity of the means at initial time, t_1,

compared to the one at time t_2, which in turn show a higher % confidence level for the similarity of means compared to the one at t_3, so on so forth are chosen. In short, the biomarkers which shows a decreasing level of confidence for similarity of means with increasing time is considered to be the best marker for allograft rejection.

Metric Definition

For a given gene, Let $y = \{y_1, y_2, ..., y_{14}\}$ be a sequence defined such that y_i is the maximum confidence level for the similarity of means at time t_i. As stated in Section 5.3, allogeneic and syngeneic patients do not indicate any inherent difference at initial time and hence a higher confidence level for similarity of the means is required at the initial time $t_1 = 0$ hr. Therefore, a threshold of 0.5 is chosen for the y_1 value at time $t_1 = 0$ hr. A set of 20 genes are obtained as indicated in Table 5.1.

Table 5.1: The 20 genes obtained

Genes	
4 granz B	7 MLC-2
3 TNF-a	3 G-CSF R
1 Pro-C5a	2 MBL-2
4 perforin	8 GSH Px
2 SAA4	4 serglycin
1 C4	3 IFN-b
9 MTHFD2	5 TLR-7
10 B2-M	12 rp S24
3 IL-1b	9 sepiapterin R
12 rp L8	
2 SAP	

For the purpose of obtaining a metric G, various other parameters are defined as shown:

$$\Delta_{i,j} = (y_i - y_j + \zeta) \quad \forall\, i = 1,2,..,n-1 \text{ and } j = i+1,..n \tag{5.4}$$

$$\Delta_{i,j} = 0 \qquad \forall\, i = 1,2,..,n-1 \text{ and } \forall\, j \leq i \tag{5.5}$$

Ideally the RHS term in Equation 5.4 should be greater than zero for any $\zeta >= 0$ (tolerance value). This might not be the case when dealing with a real biological data such as the one in this study. A metric G is defined for a given gene as follows:

$$G = \frac{E}{(E+F)} \tag{5.6}$$

$$\tag{5.7}$$

where E and F are total number of positive and negative values in the Δ_k matrix, respectively. Δ_k is a $(n-1) \times n$ matrix with elements $\{\Delta_{i,j}\}$ as given in Equations 5.4 and 5.5. The ζ value is chosen based on the standard deviation of the confidence values as suggested in Equation 5.8:

$$\zeta = \frac{0.1}{n} \sum_{i=1}^{n} \sigma_i = 0.03 \tag{5.8}$$

where σ_i is the standard deviation of the confidence values of the 20 genes, indicated in Table 5.1, at time i. The gene with a higher value of metric G, shows a better decreasing trend in the confidence level and hence is a better marker for early detection of allograft rejection compared to the gene with a lower value of G. Table 5.2, gives a list of the 20 genes in the decreasing order of the G value. Table 5.2 shows the ranking of the selected 20 genes according to their suitability to differentiate allogeneic and syngeneic over the whole time course. In Table 5.2, three genes are chosen based on the criteria of them being greater than 95 % of the maximum achievable value of the metric G (=1).

Figure 5.1, gives the plot for the confidence level of these three chosen genes which show a generally decreasing trend from time t_1 to t_{14}. The three best genes out of the selected 20 were chosen. The figure shows the degree of similarity or dissimilarity of these genes between allogeneic and syngeneic patients. The higher the confidence level (y axis) the more similar is the gene between the two groups. The metric/figure shows that these 3 genes are most similar at the earliest time

Table 5.2: The estimated metric *G* for the 20 genes

	Genes	**Metric G**
1	1 Pro-C5a	0.9560
2	3 TNF-a	0.9556
3	4 granz B	0.9535
4	4 perforin	0.8791
5	2 SAA-4	0.7802
6	9 MTHFD2	0.7753
7	10 B2-M	0.7692
9	3 IL-1b	0.7555
8	12 rp L8	0.7555
10	2 SAP	0.7356
11	7 MLC-2	0.7111
12	3 G-CSF R	0.6923
13	2 MBL-2	0.6889
14	8 GSH Px	0.6813
15	4 serglycin	0.6593
16	1 C4	0.6555
17	3 IFN-b	0.6373
18	5 TLR-7	0.6067
19	12 rp S24	0.6043
20	9 sepiapterin R	0.3297

points and very dissimilar at the later times. As early as 3 hrs these genes become dissimilar/differentiate allogeneic from syngeneic.

The three genes chosen for analysis are as follows:

The complement product Pro-C5a, the official name is hemolytic component (Hc), aliases used are C5 or C5a: The protein encoded by the C5 gene plays an important role in inflammatory and cell killing processes. The C5a gene is an anaphylatoxin that possesses potent chemotactic activity and is derived from the alpha polypeptide via cleavage with a convertase. In the literature it is indicated that the anaphylatoxin C5a might have potential as an early and reliable marker for acute renal allograft rejection (Mueller et al., 1997)..

Tumor Necrosis Factor- (TNFa): The TNF gene encodes a multifunctional pro-inflammatory cytokine that belongs to the tumor necrosis factor (TNF) superfamily. Increased levels of TNF were demonstrated within the blood of patients during episodes of renal allograft injection and thus have been suggested as a useful early and discriminatory marker of rejection (Tuschida et al., 1992).

Granzyme-B (GZMB): The protein encoded by GZMB gene is crucial for the rapid induction of target cell apoptosis (programmed cell death) by CTL in cell-mediated immune response. The accurate diagnosis of acute rejection by measuring granzyme B mRNA in urinary cells, have been successfully demonstrated. Furthermore, it is stated that measuring the levels of granzyme B could be used predict the development of acute rejection (Lo et al., 2001).

The allogeneic and syngeneic time trends of the chosen biomarkers indicate the difference between successful and failed transplantation. Figure 5.2, shows a graphical representation of the average $\Delta\Delta C_T$ values of all replicates at each time instant for the three chosen biomarkers aforementioned. The divergence of the two curves (allogeneic and syngeneic) from time $t_1 = 0$ hr till $t_{14} = 7$ days indicates the occurrence of allograft rejection.

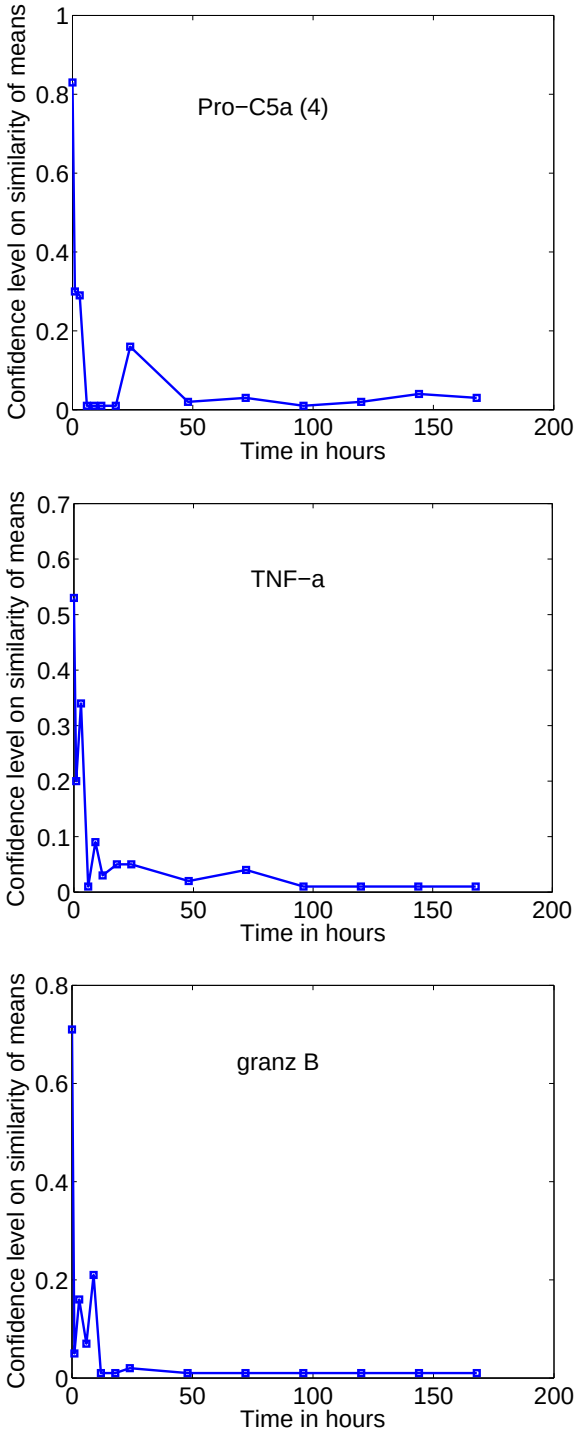

Figure 5.1: The chosen biomarkers based on hypothesis testing method suggested in this section. The chosen biomarkers are used for further validation

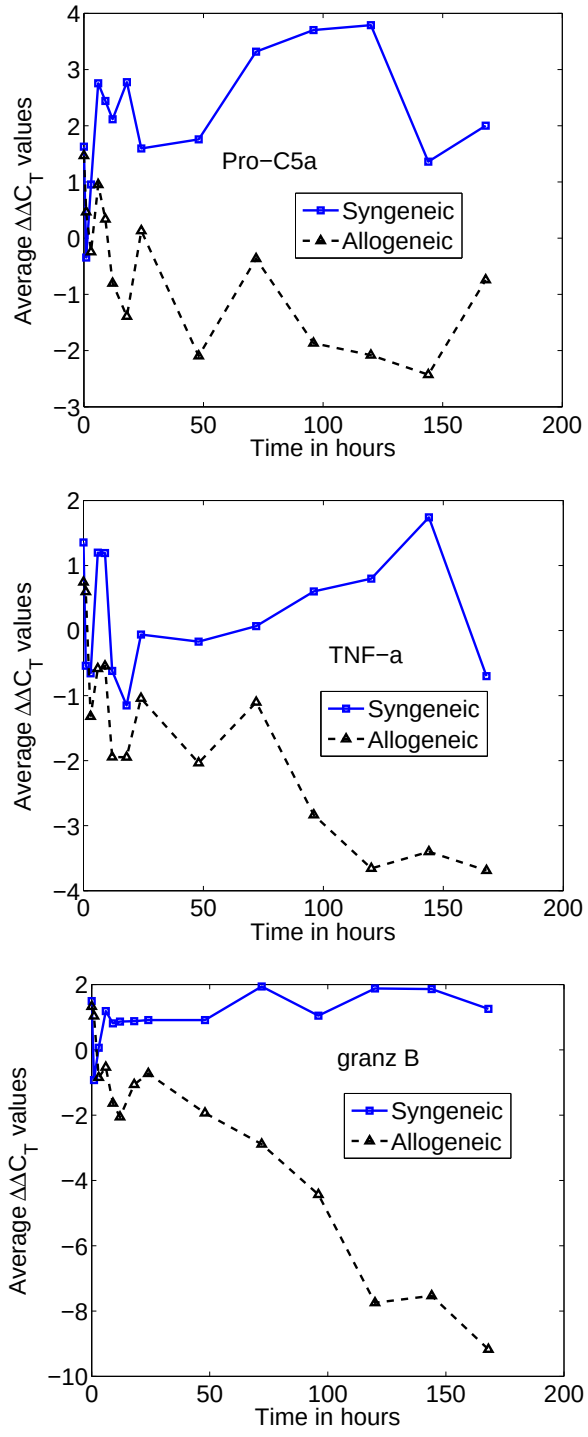

Figure 5.2: Time trend indicating the difference between the data obtained from allogeneic and syngeneic patients.

5.4.2 Application of the three markers in a multivariate framework

The three chosen biomarkers in Section 5.3, are subjected to various preliminary graphical analysis to illustrate the advantage of using a multivariate framework for detection of allograft rejection.

Firstly, the three individual biomarkers are viewed in a univariate framework. At each time instant t_i, and for each of the three chosen biomarkers, mean and 95% Confidence level of the replicates are calculated separately for each class. Figures 5.4.2 and 5.4.2, indicates the mean and 95% confidence level for the $\Delta\Delta C_T$ values of the replicates grouped together for each of the 3 biomarkers from initial time (0 hr) to 12 hr mark. An overlap between the allogeneic and syngeneic confidence regions indicates a non-separable case.

Figures 5.4.2 and 5.4.2 indicate that the separation between the allogeneic and syngeneic classes happen on only at the 12 hr mark and only in TNF-α gene. Also, the figures show that no separation between the allogeneic and syngeneic classes is achieved at any time prior to 12 hours, for any of the three biomarkers when viewed individually.

Similarly, in Figure 5.5 it can be seen that the three chosen biomarkers when viewed in a multivariate framework shows a class separation as early as 6 hours. In Figure 5.5, 2 dimensional ellipses for all the possible combination of biomarkers($\{$Pro-C5a(4), TNFa$\}$, $\{$Pro-C5a(4), Granzyme B$\}$, and $\{$TNFa, Granzyme B$\}$) are plotted. The ellipse are plotted with the mean $\Delta\Delta C_T$ value as center and 95% Confidence level, of the biomarkers, as the major and minor axes.

Based on the aforementioned results from the plots, it can be concluded that a multivariate framework of using the biomarkers for detection of allograft rejection is advantageous. The three biomarkers chosen are validated using independent renal transplantation microarrays obtained from the ncbi-geo website.

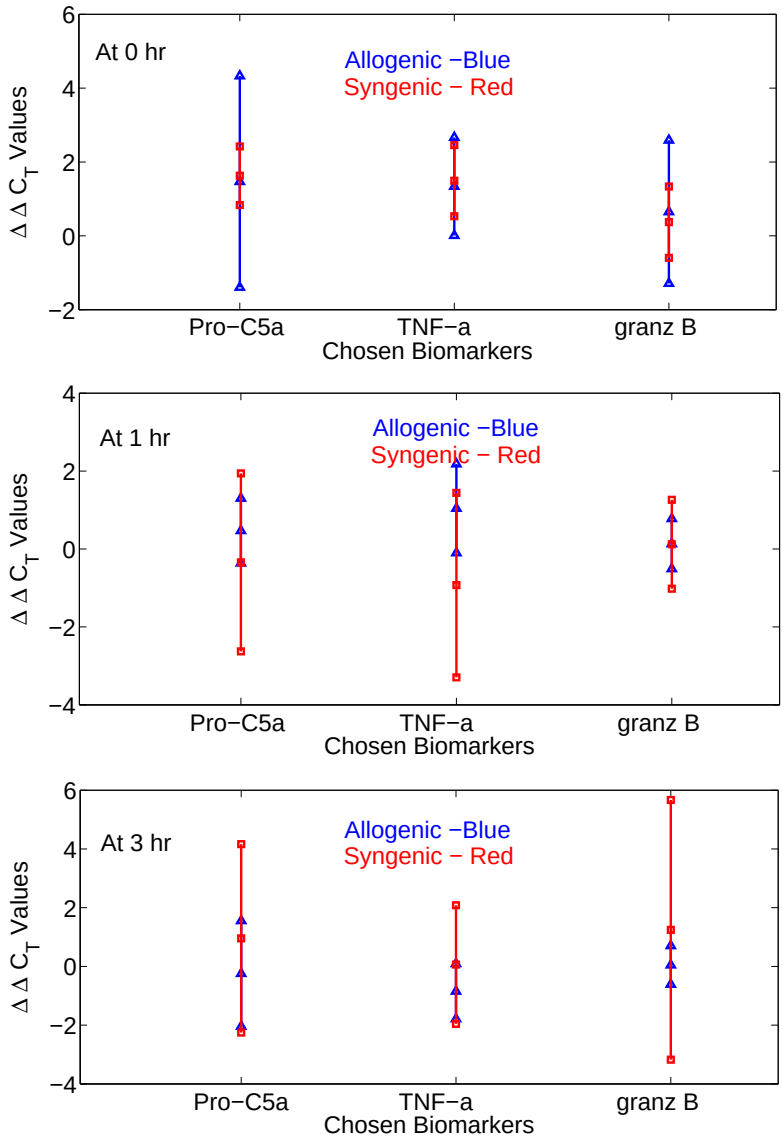

Figure 5.3: A univariate plot showing the mean $\Delta\Delta C_T$ values and 95% confidence level (based on the replicates), for the three chosen biomarkers, at times 0 hr, 1 hr, and 3 hr.

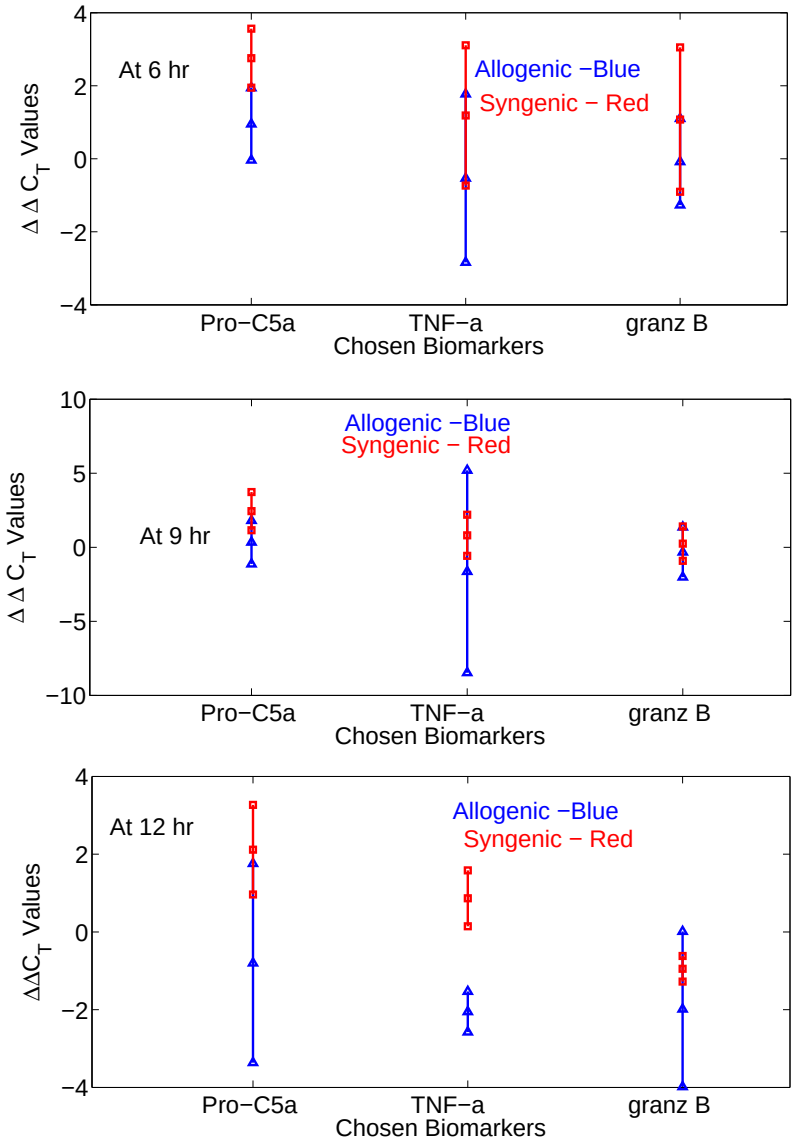

Figure 5.4: A univariate plot showing the mean $\Delta\Delta C_T$ values and 95% confidence level (based on the replicates), for the three chosen biomarkers, at times 6 hr, 9 hr, and 12 hr.

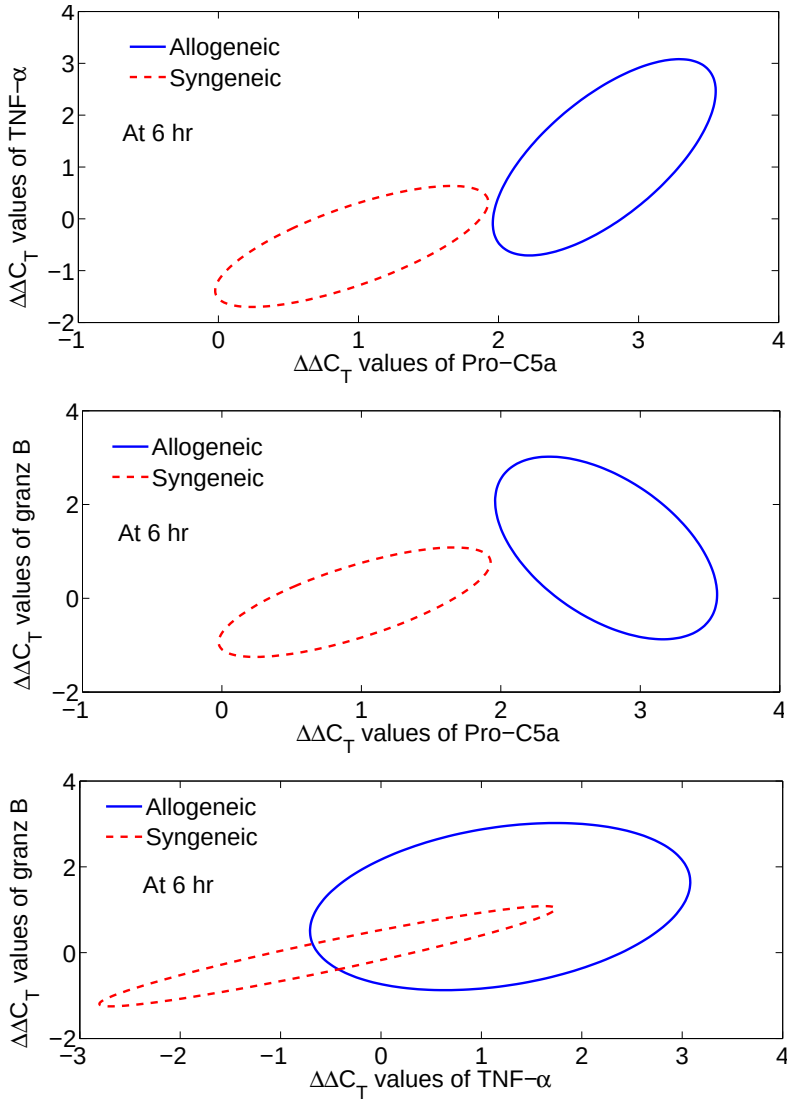

Figure 5.5: A bivariate plot showing the mean $\Delta\Delta C_T$ values and 95% confidence level (based on the replicates) for combinations of two biomarkers at 6 hr mark. Due to lack of sufficient data a 3-D model could not be built and a 2-D projection is shown for all combinations of two genes.

5.4.3 Validation of the biomarkers in independent data sets

The obtained candidate biomarkers are validated to three new and publicly-available microarray measurements on both human and animal renal transplantation patients for rejection. Before applying k-means algorithm to the microarray datasets for separating the classes (allogeneic and syngeneic), the datasets are normalized using a standard normal variate (SNV) transformation across all genes in the microarray.

Rat renal transplantation dataset

A well defined rat kidney transplantation model with strict transplant and sample preparation procedures to analyze genome wide changes in gene expression four days after syngeneic and allogeneic transplantation (Edemir et al., 2008). From the huge microarray dataset, the three chosen biomarkers, in this study, are used for analysis. K-means clustering algorithm mentioned in Section 5.3.2 is used to cluster the rat renal transplant dataset of 10 samples (With 5 samples each of allogeneic and syngeneic rats). Figure 5.6 shows the separated clusters for allogenic and syngeneic rat patients clustered using the k-means algorithm.

From Figure 5.6 it can be seen that by using the three chosen biomarkers the microarray dataset obtained for rat experiments is clearly separated into two groups (allogeneic and syngeneic) with an accuracy of 100%. There is an underlying fact that an animal experimental setup produces clean data and the 82 genes chosen for calibration were sufficient for identifying the candidate biomarkers and thus help in obtaining a clear demarkation between the allogeneic and syngeneic patients.

Human renal allograft dysfunction dataset

The Renal transplant data is received from recipients who have undergone diagnostic biopsies after transplantation from April 2004 to December 2006 (Mao et al., 2011). A set of 61 patients were chosen for analysis with 34 of them showing acute rejection and the remaining 27 of them showing a stable renal function. For the purpose of this study only patients undergoing acute rejection and patients with

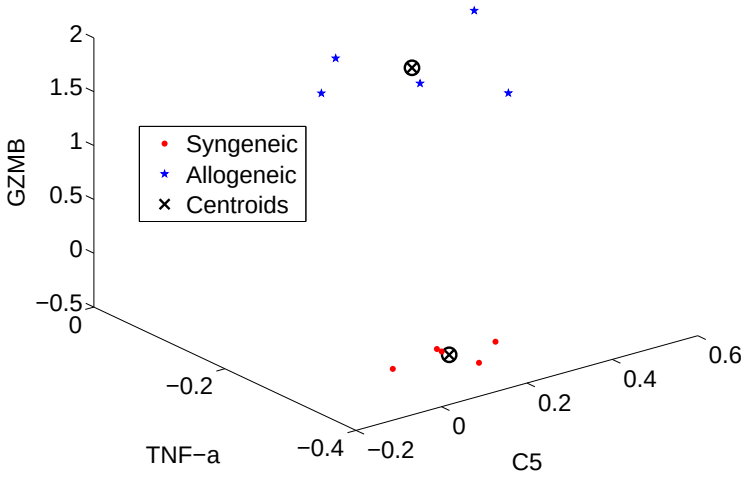

Figure 5.6: Plot indicating the two separated clusters for allogeneic and syngeneic rat samples using k-means algorithm for rat renal transplantation dataset. The three chosen biomarkers, as suggested in Section 5.4.1, are used for separation of the clusters. The centroids for the two clusters are also plotted.

stable renal functions were chosen. The patients undergoing acute tubular necrosis were not selected for analysis. Figure 5.7 shows the separated clusters for allogenic and syngeneic human patients clustered using the k-means algorithm.

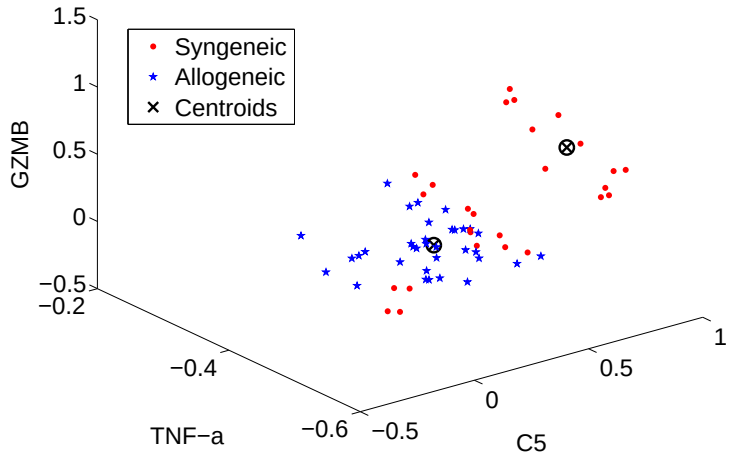

Figure 5.7: Plot indicating the two separated clusters for allogeneic and syngeneic human samples using k-means algorithm for renal allograft dysfunction dataset. The three chosen biomarkers, as suggested in Section 5.4.1, are used for separation of the clusters. The centroids for the two clusters are also plotted.

For this human renal allograft dysfunction dataset, k-means clustering algorithm achieved a separation between the two classes with an accuracy of around 75%. From Figure 5.7 it can be seen that by the three chosen biomarkers does correctly predict the patients undergoing allograft rejection. However, a small amount of patients undergoing successful transplantation is also classified as transplant rejection. The reason for this misclassification could be because of the fact that the gene expression profile for 82 genes obtained from RT-PCR dataset is not sufficient for choosing the candidate biomarkers needed for detection of transplant rejection in human patients.

Human renal allograft dysfunction dataset

The Renal transplant data is received from recipients who have undergone diagnostic biopsies after transplantation from April 2004 to December 2006 (Mao et al., 2011). A set of 74 patients were chosen for analysis with 26 of them showing renal dysfunction and the remaining 48 of them showing stable renal function. For the purpose of this study only patients undergoing acute rejection and patients with stable renal functions were chosen. The patients undergoing borderline rejection or presumed rejection were not selected for analysis. Figure 5.7 shows the separated clusters for allogenic and syngeneic human patients clustered using the k-means algorithm.

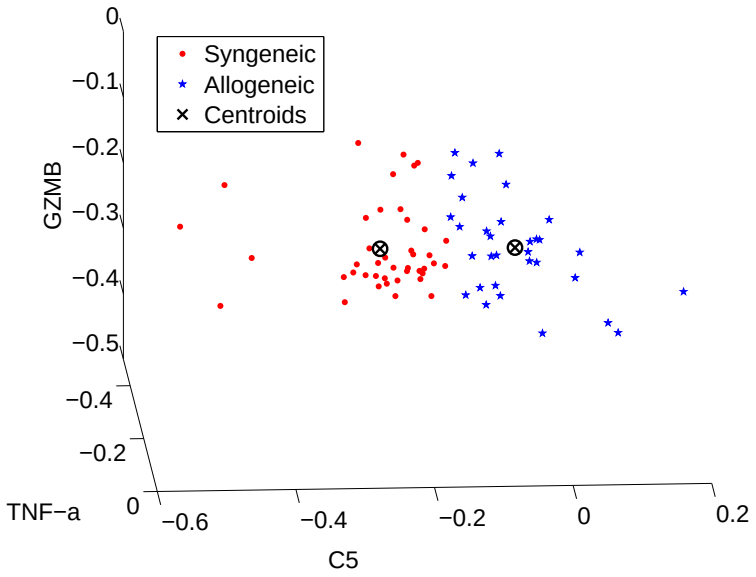

Figure 5.8: Plot indicating the two separated clusters for allogeneic and syngeneic human samples using k-means algorithm for renal allograft dysfunction dataset. The three chosen biomarkers, as suggested in Section 5.4.1, are used for separation of the clusters. The centroids for the two clusters are also plotted.

For this human renal allograft dysfunction dataset, k-means clustering algorithm achieved a separation between the two classes with an accuracy of around 68%. Again, from Figure 5.8 it can be seen that by using the three chosen biomarkers a clear separation between the allogeneic and syngeneic patients cannot be achieved.

5.5 Discussion

The procedure proposed in this work exploits the link between changes in the RNA extracted from individual tissue samples was analyzed yielding candidate biomarkers for transplantation. Furthermore, the three candidate biomarkers obtained using the proposed method is validated to new and publicly-available microarray measurements on renal transplantation patients for rejection. The results shown in this study demonstrate that the gene expression measurements from time series RT-PCR dataset can be a powerful and a fast strategy for discovering candidate biomarkers for transplant rejections. Also, the biomarkers chosen from the gene expression profiles obtained from heart transplantation patients were robust in identifying rejection even when applied on independent renal transplant patients.

5.6 Conclusion

Heart transplantation is one of the fastest developing area in the biomedical field. Heart allograft rejection is one of the biggest challenges during transplantation. Therefore, an early detection of allograft rejection can help in preventing transplantation rejection. In this work, a novel algorithm has been built for early detection of transplant rejection.

The study is divided into two parts, firstly, a set of three candidate biomarkers were chosen from a time series RT-PCR dataset, obtained from eight to 12 wk old male BALB/cByJ (BALB/c) (H-2d), C57BL/6J (B6) (H-2b) mice patients, using hypothesis testing. A metric G is defined as a part of this work for quantifying the chosen biomarkers.

Secondly, the chosen candidate biomarkers were validated to three new and publicly-available microarray measurements, from ncbi-geo website, on both human and animal renal transplantation patients for rejection. The chosen biomarkers were able to separate the allogeneic and syngeneic classes with a 100% accuracy in the case of rat patients.

For human patients, however, a separation between the allogeneic and syngeneic classes is not clear. This is due to the fact that the human genome has more genes compared to the mouse genome and the candidate biomarkers chosen from the 82 relevant genes for mouse patients may not be exhaustive for human patients. A comprehensive set of genes are needed for identifying the relevant biomarkers for human patients.

References

P. J. Morris, P. J. Delves, Transplantation, Elsevier, Oxford, 1998.

M. J. Dallman, P. J. Delves, Graft Rejection, Elsevier, Oxford, 1998.

L. M. Erickson, X. F. Yang, F. Pan, M. Kobayashi, H. Jiang, Gene expression profiles of rat heart allografts, Transplantation Proceedings 33 (2001) 562–566.

L. M. Erickson, G. Crews, F. Pan, O. Fisniku, M. S. Jang, C. Wynn, M. Kabayashi, H. Jiang, Unique gene expression profiles of heart allograft rejection in the interferon regulatory factor-1-deficient mouse, Transplant Immunology 13 (2004) 169–175.

L. M. Erickson, F. Pan, A. Ebbs, M. Kobayashi, H. Jiang, Microarray-based gene expression profiles of allograft rejection and immunosupression in the rat heart transplantation model, Transplantation 76 (2003) 582–588.

M. Schena, D. Shalon, R. W. Davis, P. O. Brown., Quantitative monitoring of gene expression patterns with a complementary dna microarray., Science 270 (1995) 467–470.

J. Quackenbush, Microarray data normalization and transformation., Nature Genetics 32 (2002) 496–501.

M. G. Walker., Pharmaceutical target identification by gene expression analysis, Mini Reviews in Medicinal Chemistry 1 (2001) 197–205.

T. F. Mueller, G. Einecke, J. Reeve, B. Sis, M. Mengel, G. S. Jhangri, S. Bunnag, J. Cruz, D. Wishart, C. Meng, G. Broderick, B. Kaplan, P. F. Halloran, Microarray analysis of rejection in human kidney transplants using pathogenesis-based transcript sets, American Journal of Transplantation 7 (2007).

M. Schena, D. Shalon, R. Heller, A. Chai, P. O. Brown, R. W. Davis, Parallel human genome analysis: Microarray-based expression monitoring of 1000 genes., Proceedings of the National Academy of Sciences 93 (1996) 10614–10619.

J. P. Carulli, M. Artinger, P. M. Swain, C. D. Root, L. Chee, C. Tulig, J. Guerin, M. Osborne, G. Stein, J. Lian, P. T. Lomedico., High throughput analysis of differential gene expression., Journal of Cellular Biochemistry 72 (1998) 286–296.

J. Li, Z. Zhang, J. Rosenzweig, Y. Y. Wang, D. W. Chan, Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer., Clinical Chemistry 48 (2002) 12961304.

J. C. Miller, H. Zhou, J. Kwekel, R. Cavallo, J. Burke, E. B. Butler, B. S. Teh, B. B. Haab, Antibody microarray profiling of human prostate cancer sera: Antibody screening and identification of potential biomarkers., Proteomics 3 (2003) 5663.

R. F. Chuaqui, R. F. Bonner, C. J. Best, J. W. Gillespie, M. J. Flaig, S. M. Hewitt, J. L. Phillips, D. B. Krizman, M. A. Tangrea, M. Ahram, W. M. Linehan, V. Knezevic, M. R. Emmert-Buck, Post-analysis follow-up and validation of microarray experiments., Nature Genetics 32 (2002) 509–514.

K. Allanach, M. Mengel, G. Einecke, B. Sisa, T. M. L. G. Hidalgo, . P. F. Hallorana, Comparing microarray versus rt-pcr assessment of renal allograft biopsies: Similar performance despite different dynamic ranges., Americal Journal of Transplantation 8 (2008) 1006–1015.

N. Venkat, A. Ben-Zvi, S. L. Shah, Inferring gene networks using robust statistical techniques., Statistical Applications in Genetics and Molecular Biology 10 (2011).

C. Paya, A. Humar, E. Dominguez, K. Washburn, E. Blumberg, B. Alexander, R. Freeman, N. Heaton, M. D. Pescovitz, Efficacy and safety of calganciclovir vs. oral ganciclovir for prevention of cytomegalovirus disease in solid organ transplant recipients, Americal Journal of Transplantation 4 (2004) 611–620.

M. L. Whitfield, D. R. Finlay, J. I. Murray, O. G. Troyanskaya, J. T. Chi, A. Pergamenschikov, T. H. McCalmont, P. O. Brown, D. Botstein, M. K. Connolly, Systemic and cell type-specific gene expression patterns in scleroderma skin, Proceedings of the National Academy of Sciences 100 (2003) 12319–12324.

M. Koulmanda, M. Bhasin, L. Hoffman, Z. Fan, A. Qipo, H. Shi, S. Bonner-Weir, P. Putheti, N. Degauque, T. A. Libermann, H. A. Jr., J. S. Flier, T. B. Strom, Curative and β cell regenerative effects of α1-antistrypsin treatment in autoimmune diabetic NOD mice, Proceedings in National Academy of Science 105 (2008) 16242–16247.

S. M. Flechner, S. M. Kurian, S. R. Head, S. M. Sharp, T. C. Whisenant, J. Zhang, J. D. Chismar, S. Horvath, T. Mondala, T. Gilmartin, D. J. Cook, S. A. Kay, J. R. Walker, D. R. Saloman, Kidney transplant rejection and tissue injury by gene profiling of bipsies and peripheral blood lymphocytes, Americal Journal of Transplantation 4 (2004) 1475–1489.

P. A. Horwitz, E. J. Tsai, M. E. Putt, J. M. Gilmore, J. J. Lepore, M. S. Parmacek, A. C. Kao, S. S. Desai, L. R. Goldberg, S. C. Brozena, M. L. Jessup, J. A. Epstein, T. P. Cappola, Detection of cardiac allograft rejection and response to immunosuppressive therapy with peripheral blood gene expression, Circulation 110 (2004) 3815–3821.

D. G. Melvin, M. Niranjan, R. W. Prager, A. K. Trull, V. F. Hughes, Neurocomputing applications in post-operative liver transplant monitoring. (1997).

R. J. Corry, H. J. Winn, P. S. Russell, Primarily vascularized allografts of hearts in mice: the role of h-2d, h-2k, and non-h-2 antigens in rejection., Transplantation 16 (1973) 343–350.

T. F. Mueller, C. Ma, J. A. Lederer, D. L. Perkins, Differentiation of stress, metabolism, communication, and defense responses following transplantation, Journal of Leukocyte Biology 73 (2003) 379–390.

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

C. T. Thiel, C. Kraus, A. Rauch, A. B. Ekici, B. Rautenstrauss, A. Reis, A new quantitative PCR multiplex assay for rapid analysis of chromosome 17p11.2-12 duplications and deletions leading to HMSN/HNPP, Eur. J. Human Genet. 11 (2002) 170–178.

M. W. Pfaffl, A new mathematical model for relative quantification in real time RT-PCR, Nucleic Acid Research 49 (2001) 2002–2007.

S. Ray, R. H. Turi, Determination of number of clusters in k-means clustering and application in colour image segmentation., Proc. in 4th Int. Conf. on Adv. Pattern Recog. and Digital Tech. (1999).

D. Steinley, K-means clustering: A half-century synthesis, British Journal of Mathematical and Statistical Psychology 59 (2006) 1–34.

T. F. Mueller, M. Kraus, C. Neumann, H. Lange, Detection of renal allograft rejection by complement components c5a and tcc in plasma urine, Journal of Labaratory and Clinical Medicine 129 (1997) 62–71.

A. Tuschida, H. S. N. Thomson, W. W. Hancock, Tumor necrosis factor production during human renal allograft rejection is associated with depression of plasma protein c and free protein s levels and decreased intragraft thrombomodulin expression, The Journal of Experimental Medicine 175 (1992) 81–90.

B. Lo, C. Hartono, R. Ding, V. K. Sharma, R. Ramaswamy, B. Qian, D. Serur, J. Nouradian, J. E. Schwartz, M. Suthanthiran, Noninvasive diagnosis of renal-

allograft rejection by measurement of messenger rna for perforin and granzyme b in urine, The New England Journal of Medicine 344 (2001) 947–954.

B. Edemir, S. Kurian, M. Eisenacher, D. Lang, C. Muller-Tidow, G. Gabriels, D. Salomon, E. Schlatter, Activation of counter-regulatory mechanisms in a rat renal acute rejection model, BMC Genomics 9 (2008) 71.

Y. Mao, H. yang, M. Wang, W. Peng, Q. He, Z. Shou, H. Jiang, J. Wu, Y. Fang, H. Dong, J. H. Che, Feasibility of diagnosing renal allograft dysfunction by oligonucleotide array: Gene expression profile correlates with histopathology, Transplant Immunology 24 (2011) 172 – 180.

6

Conclusions, Summary, and Future Work

6.1 Concluding Remarks

Biological engineering has many important applications involving design, control and operation of biological systems. Biological engineering encompasses a wide range of fields including bioprocess engineering, biomedical engineering, systems biology, cellular engineering, genetic engineering, etc. In this thesis bioengineering problems are viewed in the framework analogous to chemical process engineering problems and statistical and machine learning tools are applied in their analysis. The three aspects of process systems engineering that have been studied as a part of this work are modeling, monitoring, and fault detection. Robust learning algorithms indeed have shown the potential to be used as tools to develop and evaluate the

performance of bioengineering systems. Several statistical and machine learning tools were used to solve some of the common complexities associated with bioengineering systems.

1. Obtaining a reduced complexity model:

    a. For inferring the gene network, the number of connections per gene was reduced using the Akaike information criterion. The AIC method achieves a trade-off between model accuracy and model complexity.

    b. For building multivariate sensors for monitoring the microalgal culture conditions the model complexity was minimized using support vector regression.

    c. For identifying candidate biomarkers, the proposed method was able to choose the minimum number of biomarkers based on the defined quantification approach.

2. Obtaining statistically significant results:

    a. In the gene network inference problem, statistically insignificant connections (spurious connections) were eliminated using leave-one-out jackknifing.

    b. In the Raman based sensor for monitoring the culture conditions, a statistically significant relation between the experimental measurement and predicted outcomes (sample correlation coefficient) was used for choosing the suitable preprocessing technique.

    c. For identifying candidate biomarkers, a statistically significant confidence level based approach was used to test the separation of the two clusters (allogeneic and syngeneic).

3. Obtaining a noise-insensitive solutions:

    a. The signal component of the gene expression measurements obtained was extracted using the partial least squares (PLS) approach. Thus the

obtained connectivity matrix using the extracted signal component was insensitive to the noise in the measurements.

b. The noise in Raman spectral measurements obtained due to turbidity, bubbles in the system, and background fluorescence were reduced using preprocessing techniques including Savitzky-Golay filtering, SNV transformation, etc.

c. The presence of noise in the RT-PCR dataset measurements were taken into account while defining the quantification measure for choosing the biomarkers.

4. Obtaining a strategy for overcoming data scarcity:

   a. For inferring the gene network, the proposed PLS/leave-one-out jackknifing/AIC algorithm used the knowledge of sparsity of the gene connectivity matrix to obtain a robust estimate.

   b. For building a multivariate sensor, the advantage of SVR algorithm in working with small sample datasets was exploited to obtain a model for predicting the concentrations of glucose and oil content.

   c. For identifying candidate biomarkers, the biological knowledge of separation between the allogeneic and syngeneic clusters with time was incorporated. This approach was able to overcome the data deficiency as small number of measurements were enough to identify relevant biomarkers for separation.

6.2 Summary

The thesis has presented three representative biological engineering systems and new robust learning algorithms have been developed. The following points summarize the contributions outlined in this thesis:

6.2.1 Inferring Gene Networks

- In this thesis a new algorithm has been proposed for reverse engineering gene networks from linear ODEs using bilinear transformation and a combination of well known statistical tools including partial least squares (PLS), leave-one-out jackknifing, and the Akaike information criterion (AIC).
- The proposed algorithm was tested on various simulated networks and the improved performance over the current existing techniques in the literature was highlighted.
- Due to the underdetermined nature of the ODE system, various challenges and limitations in inferring gene networks from microarray datasets are also addressed.
- Finally, the proposed algorithm applied to an experimental nine-gene network for *E. Coli* was able to successfully outperform methods currently available in the literature.

6.2.2 Monitoring a Bioreactor System

- In this thesis, an online multivariate sensor to monitor concentrations of biomass, glucose and oil content in microalgal cultures has been built. An algorithm combining Raman spectroscopy and support vector regression was used for building the multivariate sensor. Even though, the combined use of Raman spectroscopy and support vector regression has recently been reported for monitoring of blood glucose (Huang et al., 2010), monitoring of cellular and intracellular metabolites concentration is a more complex task.
- The sensor built using support vector regression is compared with other techniques including principal components regression (PCA), partial least squares regression (PLSR), and kernel PCA and the superior performance of the proposed method is quantified.

- As a part of the study, the effect of preprocessing techniques including Savitzky-Golay filtering, baseline correction, and standard normal variate transformation on the model building exercise were assessed. Suitable preprocessing technique for estimating the concentrations of biomass, glucose, and oil content were also obtained based on the goodness of fit.
- The proposed sensor was able to successful monitor and predict the concentrations of biomass, glucose, and oil content.

6.2.3 Detection of Transplant Rejection

- In this thesis, a novel technique for choosing candidate biomarkers for detecting the allograft rejection is presented. The method uses hypothesis testing to obtain a set of candidate biomarkers from a time series RT-PCR dataset, obtained from eight to 12 wk old male BALB/cByJ (BALB/c) (H-2d), C57BL/6J (B6) (H-2b) mice patients. A metric G is defined for quantifying the chosen biomarkers.
- The chosen candidate biomarkers were validated using three publicly-available microarray datasets, from ncbi-geo website, on both human and animal renal transplantation patients.
- The chosen biomarkers gave a good separation between the allogeneic (transplant rejection) and syngeneic (successful transplant) classes in the case of rat patients. However, for separation between the classes of human patients more work needed to be done to ensure the chosen genes

6.3 Future Work

The following areas of future work are suggested:

- **Implementation of a control strategy in the algal bioreactor system for maximizing the oil productivity:** Optimizing the oil productivity requires

building a model, developing a sensor, and defining a control law. In a previous work, De la Hoz et al. (2011) focused efforts on building a model for the microalgal biotransformation. The present work involved developing a robust multivariate sensor for monitoring the concentrations of the biomass, glucose, and oil content in a microalgal bioreactor. The future work involves defining a control law for optimizing the concentration of biomass and the oil content by using the glucose concentration as the control variable.

- **Suggesting an good experimental Strategy for obtaining candidate biomarkers:** As mentioned earlier, obtaining candidate biomarkers for prediction of transplant rejection is one of the biggest challenges in biomedical studies. A good experimental strategy can help in obtaining a good set of biomarkers for early detection of rejection. For this purpose, a plot of the daily and hourly values of the average $\Delta\Delta C_T$ (defined in Section 5.2.2) values are presented. Significant variations between the hourly and daily plot indicate the necessity for performing more hourly experiments for early detection of transplant rejection. Figure 6.1, show the daily and hourly plot for the gene TNF-α indicating the need for more hourly experiments for early detection of transplantation failure.

  Implementation of the proposed method, in Chapter 5, on the suggested dataset could lead in obtaining a superior set of candidate biomarkers for effective identification of transplant rejection.

References

Y. Y. Huang, C. M. Beal, W. W. Cai, R. S. Ruoff, E. M. Terentjev, Micro-Raman spectroscopy of algae: Composition analysis and fluorescence background behavior, Biotechnol. Bioeng. 105 (2010) 889–898.

H. De la Hoz, A. Ben-Zvi, R. Burrell, W. McCaffrey, The dynamics of heterotrophic algal cultures, Bioresour. Technol. 102 (2011) 5764–5774.

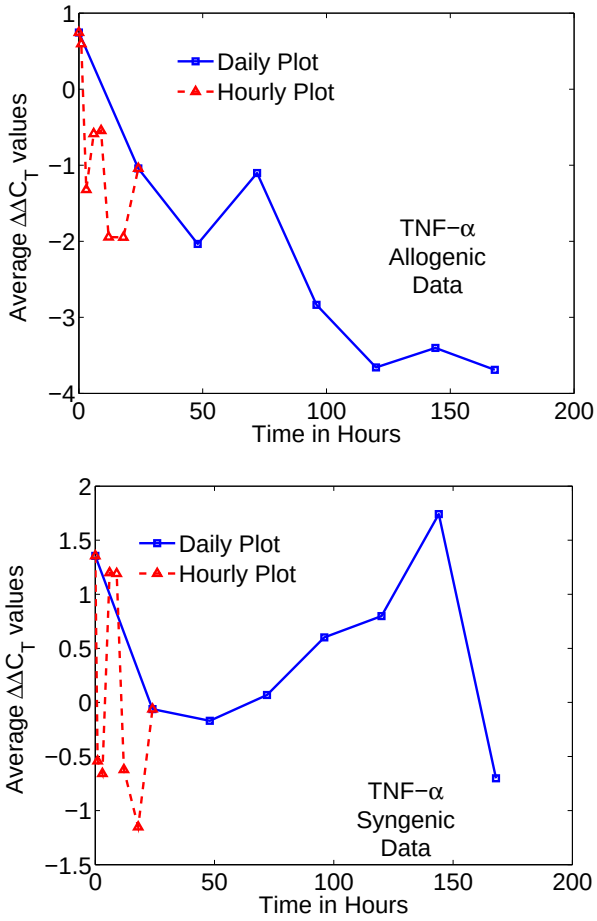

Figure 6.1: Daily and Hourly Plot of the TNF-α for developing a good experimental strategy. Variations between the hourly and daily plots indicate the need for conducting more experiments on the hourly basis.

A

Heart Allograft Rejection Dataset

In this appendix, the list of genes and their gene symbols are presented along with the actual times when the ΔC_t (refer Chapter 4 for definition) values are calculated. As suggested in Chapter 4, the gene **FOLbp3** has a lot of missing ΔC_T values and therefore is removed from analysis. Likewise, the gene **CK** which has missing ΔC_T values on day 4 for all replicates is also removed.

Table A.1: ΔC_T values for first 31 genes for the isograft Patients

Gene Symbol	ref C_0 avg	0h	0h	0h	1h	1h	1h	1h	3h	3h	3h
GAPDH	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
C1q-a	**6.3**	6.2	6.3	5.5	5.6	6.2	5.7	6.4	5.6	6.1	6.7
C1q-b	**5.1**	4.2	4.5	4.0	3.9	4.1	4.2	4.4	3.7	4.0	4.7
C1q-c	**7.1**	6.8	6.9	6.0	6.2	7.2	6.5	7.0	6.1	6.2	7.1
C1-Inh	**3.0**	2.7	2.8	1.9	2.4	2.3	2.6	2.8	2.5	2.2	2.9
C3	**4.7**	5.2	5.0	2.8	4.9	3.8	4.2	4.5	4.2	4.2	5.2
C3aR	**7.7**	7.3	7.0	7.3	7.0	7.7	6.7	7.7	6.3	6.8	7.9
C4	**7.1**	7.6	7.5	5.4	7.7	6.7	6.5	7.3	8.1	7.1	7.8
C5aR	**7.0**	6.5	7.0	6.9	5.8	6.5	6.2	6.8	5.5	5.4	6.1
C9	**9.3**	9.5	8.8	9.3	10.7	9.1	6.9	7.8	7.5	7.8	8.5
compl H	**3.5**	2.7	2.1	2.4	2.8	2.9	2.6	3.1	2.8	2.1	2.9
DAF-1	**6.8**	6.1	5.9	5.4	5.9	6.3	6.7	6.9	6.3	6.3	7.3
Pro-C5a	**10.4**	12.2	12.1	11.6	11.6	10.8	8.4	9.3	12.3	9.9	11.9
properdin	**6.1**	6.2	6.3	5.3	5.9	5.1	5.5	6.0		6.2	6.8
APP	**2.3**	1.7	1.8	1.5	1.8	2.2	2.4	2.8	1.7	1.2	2.1
CRP	**8.3**	10.1	9.9	11.1	9.6	8.4	6.3	6.8	9.5	7.5	9.4
MacManR	**6.4**	5.4	5.7	5.4	5.4	6.5	6.7	6.6	5.3	5.3	5.8
Man6-PR	**5.2**	4.6	4.5	4.4	4.7	4.9	4.5	4.8	4.6	4.1	5.0
MBL-2	**8.9**	10.2	10.0	11.1	10.2	8.7	6.7	7.6	9.7	7.8	9.7
SAA-2	**8.4**	10.3	8.8	8.5	9.8	8.6	7.1	7.6	7.0	7.5	9.3
SAA-4	**9.6**	10.7	10.7	11.4	10.8	9.3	7.6	8.1	9.7	8.8	10.4
SAP	**8.8**	10.9	11.4	12.1	10.1	9.1	7.1	7.5	10.8	8.8	10.2
G-CSF R	**8.1**	8.5	8.1	7.9	7.6	8.6	6.6	7.7	7.3	7.9	8.2
GM-CSF R2a	**8.1**	9.2	9.2	9.8	7.2	7.8	5.8	7.2	8.0	7.1	7.7
IFN-b	**9.4**	11.5	11.4	12.6	10.9	9.8	7.5	8.1	10.6	8.6	10.2
IFN-g	**5.7**	5.3	5.4	5.3	5.3	5.4	5.3	5.7	5.5	4.9	5.7
IL-1a	**7.6**	7.9	6.9	7.1	7.2	7.4	5.9	6.5	6.3	5.9	6.8
IL-1b	**8.4**	7.9	8.4	9.7	4.8	5.5	5.8	7.0	4.6	3.5	5.0
IL-2	**7.1**	8.5	6.9	7.7	7.3	9.3	6.3	6.8	5.7	6.3	8.1
IL-6	**7.8**	7.5	7.1	7.7	3.8	4.0	5.0	5.8	3.3	1.8	2.4
IL-10	**8.9**	11.3	11.1	11.6	9.7	8.8	7.4	8.3	10.1	8.2	9.7

Table A.2: ΔC_T values for first 31 genes for the isograft Patients

6h	**6h**	**6h**	**9h**	**9h**	**9h**	**12h**	**12h**	**12h**	**12h**	**18h**	**18h**	**18h**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
6.5	6.7	6.4	6.7	6.5	6.9	6.4	6.8	6.6	7.1	6.6	6.2	6.0
4.2	4.6	4.5	4.4	4.3	4.5	4.2	4.5	4.2	4.7	4.4	3.6	3.8
6.1	6.7	6.9	7.0	6.6	7.5	6.7	7.2	6.9	7.7	7.1	6.4	6.4
2.9	2.8	3.2	3.2	2.6	2.8	2.9	2.5	2.8	3.1	2.6	2.3	2.6
5.9	5.2	5.6	5.9	5.6	4.4	4.6	4.3	4.2	5.3	4.8	3.6	4.3
7.9	8.1	7.2	8.4	7.6	7.9	6.6	7.0	6.5	7.3	7.7	6.0	6.0
8.4	7.6	8.5	8.3	8.3	7.0	7.6	7.0	7.0	7.5	7.8	6.0	7.1
5.6	6.0	6.7	6.4	5.2	5.5	4.0	4.0	4.4	4.7	5.4	3.3	3.3
9.9	10.4	8.3	9.4	9.1	9.0	10.3	9.0	10.2	10.6	10.6	11.1	10.6
3.4	3.0	3.3	3.6	3.2	3.1	2.9	2.8	3.0	3.2	3.4	2.7	2.6
6.6	7.5	6.8	7.5	6.8	7.5	7.4	7.4	7.4	7.5	7.4	6.1	6.6
12.8	13.1	13.5	12.9	12.3	13.3	12.6	11.4	12.9	13.0	13.1	13.0	13.3
6.1	6.4	7.1	7.3	6.3	5.5	4.4	5.1	4.8	5.4	5.1	3.9	3.7
2.1	2.5	2.3	2.5	2.6	2.5	1.9	2.0	2.1	2.3	1.9	1.3	1.5
10.1	11.3	10.5	11.1	9.1	10.9	10.3	9.1	9.8	9.8	9.4	10.1	9.7
5.1	5.3	5.8	5.7	5.6	5.7	5.0	5.9	5.1	5.4	5.4	5.0	5.5
4.8	5.1	5.2	5.1	5.2	5.1	4.4	4.6	4.6	4.5	4.5	3.7	4.0
10.7	12.2	10.8	11.2	9.4	11.3	10.1	9.5	10.1		10.3	10.2	10.6
9.3	5.4	6.2	8.8	8.9	6.9	9.7	8.8	9.7	10.3	10.6	9.0	10.6
11.8	10.8	10.0	11.6	10.8	12.6	10.9	10.2	11.7	11.6	12.1	12.5	12.0
11.3	11.9	11.4		9.9	11.6	10.7	9.6	10.9	10.9	11.0	10.9	9.8
8.0	7.9	7.6	8.3	6.5	6.5	4.9	4.7	4.8	5.2	6.1	4.2	4.2
7.0	6.9	7.7	7.4	8.3	8.1	6.8	7.5	7.2	7.5	7.9	7.0	6.4
11.5	13.1	11.5	12.5	10.5	12.4	11.3	10.2	11.4	11.0	10.7	11.9	11.1
5.3	5.1	5.6	5.4	5.2	4.5	4.7	4.6	5.0	5.4	5.1	4.1	4.1
7.1	7.8	7.5	8.6	7.7	8.5	7.3	6.7	7.9	8.4	6.4	6.3	7.0
4.2	3.5	4.8	4.6	2.9	3.7	3.4	2.5	4.3	4.0	3.0	2.3	2.5
9.5		6.7	8.8	9.5	9.5	7.9	8.6	9.1	9.1	8.2	9.2	8.2
3.2	2.9	3.1	3.7	2.4	3.0	4.1	4.0	4.2	4.1	4.0	3.4	4.0
10.2	10.9	10.6	11.7	9.5	9.7	10.4	9.3	10.8	11.1	11.0	10.1	10.5

Table A.3: ΔC_T values for first 31 genes for the isograft Patients

d 1	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
6.0	6.9	6.6	5.0	5.0	5.8	5.4	4.9	5.5	5.7	5.8
4.6	5.0	4.8	2.7	2.2	3.3	2.9	2.6	2.9	3.3	3.1
6.7	7.8	7.4	5.9	5.9	6.6	5.8	6.0	5.7	5.4	6.2
3.2	3.4	3.0	2.3	2.5	3.2	2.8	2.0	2.3	2.3	2.5
4.8	6.0	4.4	4.6	4.8	5.0	4.7	3.5	3.4	4.0	4.1
6.9	8.3	7.5	4.7	4.8	6.0	5.9	5.7	5.3	5.8	5.1
7.4	8.3	6.8	6.5	6.3	6.6	6.5	5.8	5.4	6.2	6.3
5.0	6.0	5.3	4.0	4.5	4.7	4.8	5.2	4.6	5.2	4.4
9.6	10.3	9.4	12.0	10.0	9.8	10.5	9.1	11.9	13.0	10.2
3.0	4.0	3.4	3.0	3.0	4.3	3.3	3.4	3.6	3.5	3.5
7.2	8.3	7.1	6.8	6.7	7.8	6.9	6.4	6.6	6.8	6.8
12.4	12.2	11.3	14.2	12.9	11.2	12.1	10.3	13.7	15.3	12.1
4.7	5.6	5.0	3.3	3.7	3.8	3.8	3.8	3.3	3.9	3.8
2.2	2.5	2.2	1.6	1.9	2.1	2.1	1.8	1.7	2.2	1.7
10.3	10.1	9.1	11.1	11.1	9.7	10.1	8.7	11.9	14.2	10.9
6.7	7.1	7.0	5.1	5.8	6.6	5.8	5.7	5.5	5.8	5.6
4.8	4.9	4.6	3.9	4.2	4.7	4.7	4.1	4.4	4.9	4.2
10.1	10.6	9.4		11.4	9.6	10.3	8.6	11.5	13.5	10.5
9.8	10.3	8.9	11.9	11.9	10.3	10.5	9.2	11.5	13.2	11.4
10.6	11.3	10.5	13.2	12.0	11.3	11.2	9.9	13.3	14.9	11.7
10.5	10.9	10.0	12.7	11.1	9.9	10.6	9.3	12.2	14.3	10.7
5.1	6.5	6.6	5.9	5.7	5.3	5.6	5.9	6.5	7.0	6.6
6.4	7.9	7.9	7.8	7.3	7.8	7.6	7.7	8.0	8.6	8.3
11.1	11.1	10.1	12.1	13.2	10.5		9.9	12.6	15.4	11.9
4.8	5.4	5.1	4.6	4.8	5.4	5.1	5.1	5.1	5.6	5.3
7.5	8.2	7.1	9.0	8.8	6.2	8.7	7.7	8.6	9.2	8.4
3.7	4.6	4.0	9.5	5.8	3.2	6.3	5.5	5.2	5.6	5.4
7.8	9.5	8.9	11.4	10.7	9.5	9.4	8.3	10.8	12.3	10.7
4.6	5.3	4.9	6.8	6.6	6.1	6.5	6.3	6.2	7.7	6.8
9.8	11.2	9.5	11.3	10.4	9.8	12.8	9.2	11.6	11.7	10.6

Table A.4: ΔC_T values for first 31 genes for the isograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
5.3	5.2	5.5	5.3	5.7	4.9	5.8	5.6	5.1	4.3	4.8	4.9
2.9	3.2	3.2	3.0	3.4	2.5	3.4	3.1	2.7	1.7	2.5	2.5
5.4	5.5	6.3	5.4	5.6	6.1	5.9	6.6	5.6	5.0	4.5	5.7
1.5	2.0	2.2	2.2	2.3	1.8	2.4	2.2	1.7	0.9	1.7	1.4
3.4	4.0	3.5	3.8	4.3	3.5	4.0	4.1	3.8	2.2	3.7	2.9
5.3	6.3	6.0	5.6	6.8	5.6	6.7	6.7	6.2	3.6	5.2	5.6
5.2	6.0	6.9	5.9	7.4	7.4	7.1	7.1	5.8	4.5	5.4	5.6
5.3	5.5	5.1	4.8	5.6	5.1	5.9	5.5	5.7	2.5	4.7	4.3
12.1	10.7	11.1	11.5	11.1		13.1	9.1	9.1	12.5	10.2	11.0
2.6	3.2	3.3	3.3	4.0	3.8	2.9	2.9	1.8	2.4	2.6	3.1
6.1	6.7	6.2	5.9	6.8	6.2	6.7	6.5	6.3	4.9	5.5	5.5
12.7	12.8	12.0	12.9	13.0	13.4	12.6	7.9	8.5	12.0	9.1	10.2
4.2	4.2	4.8	4.4	5.0	3.7	5.6	4.3	4.8	2.1	3.9	3.7
1.5	2.1	2.0	1.6	2.2	1.3	1.7	1.7	1.7	0.1	1.1	1.2
13.2	12.6	11.5	12.4	12.3	10.8	11.3	7.5	7.4	11.2	9.2	9.2
5.5	6.2	5.7	5.7	6.7	6.1	6.7	6.7	5.5	3.9	5.3	6.1
4.4	4.7	4.5	4.6	5.3	4.1	3.6	3.6	3.3	2.7	3.6	4.1
14.1	11.9	11.0	12.2	11.9		11.8	7.8	7.9	10.7	9.2	10.1
11.4	9.9	11.6	11.1	11.6	11.6	11.7	6.2	5.5	8.4	10.1	6.0
13.7	12.9	12.1	11.9	11.9		13.2	9.6	9.6	13.3	11.0	12.1
14.7	12.6	11.7	12.3	11.6	12.9	12.4	8.9	8.9	11.6	10.0	10.6
6.7	7.6	7.6	6.5	8.1	-1.7	8.5	7.4	7.6	3.8	7.0	5.5
8.3	8.3	8.5	8.0	8.8	8.0	9.0	8.6	8.3	6.5	7.8	7.5
14.0	13.7	12.1	14.2		16.8	12.0	8.3	8.5	13.3	9.8	11.1
4.7	5.5	5.5	5.0	5.8	5.4	6.0	6.1	5.4	3.4	4.5	4.6
8.2	8.7	8.8	8.9	8.8	8.6	10.1	9.9	8.8	7.8	9.6	7.5
4.7	6.4	6.7	5.8	7.7	3.5	4.8	5.0	4.4	6.4	6.7	
11.9	10.1	11.0	11.7	11.3	10.5	10.5	7.7	7.2	10.7	9.2	9.9
6.4	7.8	7.5	7.3	8.6	6.2	8.6	6.1	8.4	3.7	7.7	6.7
12.3	11.8	11.3	10.5	11.0	10.4	12.6	11.9	11.4	9.7	11.0	9.7

Table A.5: ΔC_T values for next 29 genes for the isograft Patients

Genes	ref C_0 avg	0h	0h	0h	1h	1h	1h	1h	3h	3h	3h
IL-11	**8.9**	10.2	8.7	9.0	9.2	8.7	6.9	7.6	7.9	7.8	9.3
IL-12 p35	**9.0**	9.1	8.2	10.3	9.5	8.6	6.8	7.8	9.5	7.9	10.0
IL-12 p40	**8.3**	10.5	10.2	10.7	9.6	8.8	6.7	7.5	10.1	7.7	9.3
TNF-a	**9.9**	11.2	10.9	11.7	10.6	10.1	7.9	8.9	9.0	8.9	9.9
granz B	**9.1**	10.9	10.8	10.2	10.0	8.7	6.7	7.3	9.3	8.3	9.9
granz D	**6.9**	9.2	9.4	10.1	8.5	7.0	4.8	5.3	9.4	6.5	8.6
granz E	**6.3**	8.7	8.6	9.8	7.9	6.5	4.4	4.8	8.5	5.7	8.1
granz G	**6.2**	8.7	8.5	9.6	8.1	6.2	4.1	4.6	8.8	5.5	8.0
perforin	**8.6**	9.6	9.2	9.5	9.0	8.4	6.7	7.5	8.0	7.0	9.1
serglycin	**4.8**	4.5	3.9	4.4	3.4	3.9	4.1	4.3	3.6	3.1	3.5
TLR-1	**8.6**	12.1	12.2	12.9	12.6	12.6	11.0	11.5	11.1	11.1	12.4
TLR-2	**9.0**	8.9	9.9	9.2	7.4	8.2	7.5	8.3	7.6	6.7	7.3
TLR-3	**11.0**		10.7	10.6	10.5	12.0	9.6	10.6	12.3	10.2	11.2
TLR-4	**6.5**	6.1	6.8	6.8	7.1	7.3	6.8	6.8	6.1	5.8	6.2
TLR-5	**6.5**	6.1	4.9	5.3	6.0	6.7	5.4	6.4	4.8	4.8	5.8
TLR-6	**8.8**	10.2	9.8	10.4	9.4	9.0	7.1	7.8	10.5	8.4	9.6
TLR-7	**7.4**	7.2	7.5	7.6	7.6	7.1	7.2	7.3	7.5	6.0	6.6
TLR-8	**9.0**	9.7	9.7	9.8	9.6	9.2	7.7	8.5	9.8	8.6	9.4
TLR-9	**9.5**	10.1	10.3	9.9	10.5	9.8	9.0	8.7	10.3	9.6	10.1
Aldo-a	**-1.0**	-1.1	-1.2	-1.1	-1.1	-1.0	-0.3	-0.4	-0.9	-1.5	-1.1
CARAT	**8.4**	7.9	7.6	7.8	7.5	8.8	8.3	8.6	7.6	7.6	7.9
Cat D	**0.8**	0.5	0.4	0.5	0.6	0.6	1.3	1.1	0.7	0.3	0.6
CK	**-0.5**	-1.1	-1.4	-1.0	-1.5	-0.4	0.1	-0.3	-1.3	-1.8	-1.2
GDH	**5.8**	5.5	5.2	3.6	5.4	4.9	3.9	4.5	6.1	5.2	6.2
IDO	**8.2**	9.3	8.9	8.9	8.9	8.8	6.9	7.8	8.3	9.1	8.8
LDH-2B	**0.7**	0.7	0.2	0.3	0.1	0.7	0.9	2.3	0.3	2.1	0.2
MEP	**2.9**	2.3	2.2	2.4	2.1	2.6	2.9	3.2	2.7	2.0	2.5
ANK-1	**9.0**	10.5	9.4	10.3	9.6	9.5	7.8	8.3	10.4	9.0	10.1
b-actin	**2.3**	1.5	1.9	1.5	1.5	1.9	2.4	3.0	1.3	1.1	2.1

Table A.6: ΔC_T values for next 29 genes for the isograft Patients

6h	**6h**	**6h**	**9h**	**9h**	**9h**	**12h**	**12h**	**12h**	**12h**	**18h**	**18h**	**18h**
9.5	9.7	8.0	9.9	10.2	10.0	7.6	8.6	6.9	7.2	6.7	6.6	7.2
10.4	9.0	9.0	10.3	9.7	10.0	9.6	8.6	9.6	10.1	8.9	9.4	8.8
10.8	11.3	9.7	11.3	9.4	10.7	9.8	8.6	9.5	9.4	9.2	9.5	9.6
11.1	11.4	10.8	11.8	10.2	11.3	9.2	8.6	9.6	9.8	9.2	8.3	8.8
10.1	9.6	11.1	10.6	9.7	9.5	10.0	9.3	10.1	10.4	10.3	9.8	9.9
9.6	11.1	9.7	10.4	8.5	10.0	9.2	8.2	9.0	9.3	8.9	8.8	9.3
9.3	10.7	9.3	10.0	8.1	9.2	8.4	7.4	8.6	8.5	8.4	8.4	8.7
9.0	10.1	9.3	9.8	8.2	9.6	8.4	8.0	8.6	9.0	8.3	8.3	8.9
10.0	9.4	8.0	9.7	9.4	10.4	9.3	9.1	9.9	10.1	9.3	9.7	9.6
3.0	2.2	3.6	3.1	2.8	1.7	1.6	1.1	0.7	1.0	1.9	0.6	0.9
12.8	11.5	10.8	12.3	13.1	13.4	11.1	12.3	11.4	12.4	12.9	11.5	11.2
7.0	6.9	7.7	8.1	7.4	7.0	6.5	6.8	7.0	7.1	7.7	5.9	5.9
11.0	10.7	10.5	10.7	12.3	11.9	12.4	12.9	12.5	11.9	12.9	12.4	12.9
6.8	5.1	5.0	7.3	7.3	6.6	5.7	6.9	6.1	6.1	6.6	6.2	6.2
6.0	6.6	3.9	5.8	6.8	7.3	6.7	6.8	7.0	7.3	7.1	6.8	6.1
10.0	9.8	10.3	10.2	9.3	9.6	8.6	8.0	8.1	8.6	9.1	8.1	8.1
6.0	6.5	7.1	6.6	6.6	7.0	7.8	7.7	7.5	8.8	8.7	7.2	7.3
8.8	8.8	9.4	8.8	9.2	8.6	8.8	8.8	8.7	9.2	9.2	8.6	8.3
10.9	10.6	10.7	10.5	10.2	10.3	9.8	9.9	9.8	11.8	11.6	9.7	9.8
-1.1	-0.8	-0.8	-1.1	-0.9	-0.9	-0.7	-0.9	-1.1	-0.9	-0.8	-0.9	-0.9
8.2	7.0	6.6	6.9	9.8	9.4		9.5	8.9	10.0	9.6	9.5	9.4
0.5	0.9	0.8	0.7	0.8	0.7	0.5	0.7	0.3	0.5	0.7	0.4	0.2
-1.1	-0.9	-1.1	-1.1	-0.4	-0.8	0.1	0.0	-0.5	-0.2	0.1	0.2	-0.4
6.0	6.0	6.6	5.9	6.0	4.9	6.1	5.6	5.7	5.7	6.7	6.0	5.3
9.3	9.8		9.0	8.8	8.7	9.4	9.2	9.7	10.0	10.0	9.8	9.8
0.0	0.5	0.5	0.2	0.2	0.4	1.5	0.8	0.5	0.6	1.3	2.9	0.4
2.5	2.7	2.5	2.5	2.4	2.6	2.2	2.1	2.1	2.5	2.8	2.0	1.7
10.9	11.2	10.5	10.5	10.2	10.7	10.1	9.6	10.1	9.7	11.0	10.1	10.2
2.2	2.1	2.6	2.1	2.6	1.2	0.2	0.8	0.1	0.1	0.6	-0.5	-0.5

Table A.7: ΔC_T values for next 29 genes for the isograft Patients

d 1	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
6.5	8.1	7.4	9.4	9.4	9.5	9.8	9.0	9.4	11.3	10.1
9.5	10.2	9.3	11.3	9.8	9.5	9.9	8.8	9.2	12.0	10.0
9.7	10.1	9.0	11.9	10.6	9.3	10.4	9.0	11.6	13.0	10.2
10.2	10.0	9.4	10.2	10.5	8.3	10.5	9.3	9.8	10.6	9.5
10.4	10.1	9.5	10.2	10.2	10.0	10.7	9.0	10.6	11.9	10.6
9.6	9.3	8.3	10.5	9.7	8.7	9.2	7.4	10.5	13.2	9.3
8.7	8.9	7.6	10.1	9.6	8.1	8.8	6.6	9.9	12.0	8.7
8.8	8.5	7.8	10.0	9.4	7.9	8.6	6.8	10.1	11.8	9.0
8.7	10.0	9.3	10.6	9.7	9.7	10.0	8.7	11.1	12.1	10.2
1.4	2.6	1.8	2.5	2.7	2.8	2.9	3.0	3.2	4.2	3.4
10.4	12.5	12.6	10.6	9.9	7.7	7.9	7.6	7.6	8.9	11.1
7.2	8.4	7.6	6.9	7.3	5.9	6.7	6.1	5.9	6.8	7.3
11.7	11.8	12.2	11.1	11.9	8.9	9.0	8.3	8.6	9.8	11.9
5.5	6.6	7.1	6.2	6.4	5.5	5.0	5.1	5.0	6.1	7.0
7.0	7.5	7.0	6.6	6.3	7.3	6.8	6.4	6.9	7.8	7.3
8.7	10.0	9.0	8.9	8.5	8.6	8.9	8.4	9.1	10.1	9.2
7.8	8.6	8.1	5.8	5.9	6.2	6.1	5.9	6.3	6.5	6.5
9.3	10.0	9.3	7.5	7.5	8.3	8.2	7.8	8.4	8.5	8.2
10.2	10.9	10.7	8.6	7.8	8.6	7.8	7.0	8.0	8.4	8.7
-0.3	-0.4	-0.3	-0.4	-0.2	0.2	0.0	-0.6	-0.1	0.1	0.1
8.9	9.2	9.5	9.1	8.7	8.3	7.6	7.0	8.1	8.0	9.4
0.6	0.6	1.0	-0.5	-0.3	0.5	0.4	-0.1	0.3	0.6	0.6
0.5	0.1	0.7	1.0	1.5	2.4	1.7	1.5	2.6	3.2	3.3
6.7	6.2	6.5	5.2	5.6	6.6	6.6	6.5	7.6	7.7	7.5
9.5	21.2	9.3	11.1	10.7	10.8	10.2	8.8	10.7	12.7	10.3
2.4	1.2	1.3	1.1	1.5	4.0	2.6	1.7	3.1	2.8	3.5
3.5	3.0	2.7	1.6	1.8	2.7	2.9	2.1	2.6	3.0	2.6
10.4	10.8	10.2	10.8	10.0	10.1	10.6	9.2	10.7	12.2	10.7
0.7	0.9	0.9	0.3	0.8	0.5	0.8	0.8	0.4	1.2	0.7

Table A.8: ΔC_T values for next 29 genes for the isograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**
11.2	9.9	10.2	10.7	11.8	11.3	11.6	10.7	11.4	7.6	11.4	9.9
11.4	8.9	9.8	9.4	9.5	10.8	11.5	10.5	10.3	9.2	10.6	8.9
11.7	12.0	10.8	11.5	11.4	13.4	12.4	12.0	11.2	10.3	11.4	9.1
9.9	10.7	11.0	10.7	11.5	10.0	12.3	11.8	11.0	8.8	10.4	8.5
9.2	11.3	10.0	10.9	11.0	11.1	11.2	11.8	9.9	10.3	11.4	9.3
12.0	10.8	9.5	10.0	10.7	11.8	10.0	6.5	6.5	8.9	9.1	7.8
11.9	10.1	9.3	9.6	10.3	10.7	10.1	9.9	10.0	8.8	9.3	7.7
11.4	10.3	9.4	9.7	10.2	10.3	10.1	9.9	10.2	8.6	9.4	7.7
9.9	10.9	9.9	10.6	10.9	12.0	11.6	11.6	10.0	9.4	10.6	9.1
4.1	4.0	4.7	4.2	5.0	0.5	4.9	3.7	4.3	1.3	3.8	2.7
11.1	11.7	11.1	11.0	13.0	14.3		12.8	12.6	10.4	10.6	12.3
7.0	7.7	7.7	7.5	9.1	9.3	8.5	8.0	7.7	9.0	15.0	9.5
11.0	12.0	12.4	11.8	13.9	14.9	13.5	13.6	13.1	11.8	11.3	12.5
6.7	6.8	6.6	6.3	7.8		7.5	7.9	6.8	5.6	6.0	7.5
6.8	6.8	7.5	6.4	6.9		7.8	7.7	7.0	5.1	6.6	6.3
9.7	10.6	9.7	9.0	10.5		10.5	9.9	9.7	6.8	9.1	8.5
5.8	6.6	6.2	5.9	6.7		7.0	7.1	6.4	4.9	5.5	6.2
7.9	8.6	8.2	7.8	9.2		9.0	8.9	9.0	6.4	7.6	8.0
8.0	9.0	8.5	8.6	9.7	10.1	9.7	9.5	9.2	7.8	8.5	8.6
-0.3	-0.2	-0.5	-0.4	-0.5		-0.5	-0.2	-0.7	-0.3	-0.3	-0.4
8.4	8.9	10.7	11.3	10.9		12.0	11.9	10.6	11.9	10.9	10.8
0.3	0.6	0.7	0.3	0.9		0.5	0.5	0.2	-0.8	0.0	-0.1
1.5	1.6	1.6	1.7	1.5		1.9	2.0	1.1	4.7	1.6	1.7
6.7	7.3	6.5	7.0	6.8	7.1	7.0	7.6	6.3	7.1	7.2	5.9
10.9	11.0	10.6	10.7	10.3	11.9	10.6	7.8	7.8	10.1	10.1	9.0
2.6	1.9	2.7	2.7	1.8	3.3	2.1	2.6		4.3	2.4	2.2
2.3	2.5	3.1	2.3	3.3	0.0	2.6	2.5	2.3	0.0	2.0	1.8
11.1	10.7	10.3	10.5	11.3		11.0	11.3	10.7	8.8	10.6	9.8
1.1	1.2	1.3	0.6	1.2	0.1	-0.4	-0.6	-0.9	-1.2	0.3	0.0

Table A.9: ΔC_T values for last 22 genes for the isograft Patients

Genes	**ref** C_0 **avg**	**0h**	**0h**	**0h**	**1h**	**1h**	**1h**	**1h**	**3h**	**3h**	**3h**
gelsolin	**0.2**	0.7	0.7	-0.2	-0.3	0.1	0.3	1.3	0.9	2.4	0.0
MLC-2	**-2.7**	-2.2	-3.5	-3.4	-4.0	-3.0	-3.0	-3.1	-3.6	-3.5	-3.6
crystallin	**-1.2**	-0.5	-1.0	-1.1	-1.5	-1.1	-0.7	-1.2	-1.4	3.8	-1.5
GSH Px	**1.3**	1.6	1.1	-0.1	1.8	0.6	1.1	0.8	2.5	2.7	1.6
Hsc70	**-0.1**	1.0	0.6	0.8	-0.1	0.1	0.5	1.5	0.0	1.9	-0.1
iNOS	**10.9**	9.3	9.6	10.3	9.8	10.9	10.9	11.1	10.6	9.8	10.5
MGP	**0.9**	0.4	0.5	-0.2	0.2	0.4	0.4	0.7	0.9	-0.3	0.1
DHFR	**7.5**	8.3	2.7	7.6	8.0	7.8	6.4	7.2	7.4	8.4	7.8
FOLbp3	**8.4**		8.2	7.6	7.9	7.9	6.2	7.1	7.1	7.9	7.2
GTP-CH I	**8.6**	9.7	9.7	9.9	9.1	9.2	7.2	7.9	9.2	11.6	9.2
MTHFD2	**7.4**	7.7	8.2	7.4	8.3	7.8	6.6	7.5	7.8	10.7	7.5
PTPS	**4.5**	4.3	0.8	4.2	4.1	4.6	4.5	4.8	4.0	5.1	3.9
sepiapterin R	**5.6**	6.5	5.7	5.7	5.6	6.2	6.4	6.7	5.3	6.6	5.8
B2-M	**0.7**	1.1	0.8	0.4	0.1	0.4	0.8	0.8	0.5	2.6	0.3
I-A-b one	**5.6**	5.7	5.6	4.7	4.8	4.9	4.8	6.5	5.1	9.6	5.9
I-E-b	**4.5**	5.2	4.9	3.6	3.8	4.0	4.2	4.8	4.9	4.4	5.0
MHC-1	**7.0**	8.5	8.0	7.8	7.7	7.9	6.1	6.8	8.6	9.8	7.8
BLR-1	**9.1**	11.3	10.6	10.3	9.7	8.8	7.5	8.4	11.0	11.1	9.7
EF-1a	**0.2**	0.1	-0.3	-0.5	-0.3	-0.2	-0.2	-0.2	0.2	1.2	-0.1
GAS-6	**4.3**	3.7	3.6	3.7	3.9	4.8	4.6	5.4	4.4	4.2	4.2
rp L8	**0.9**	0.9	0.3	0.7	0.1	0.5	1.1	0.6	0.4	0.9	0.5
rp S24	**0.0**	-0.1	-0.6	-0.3	-0.9	0.1	-0.1	-0.5	-0.6	1.5	-0.6

Table A.10: ΔC_T values for last 22 genes for the isograft Patients

6h	**6h**	**6h**	**9h**	**9h**	**9h**	**12h**	**12h**	**12h**	**12h**	**18h**	**18h**	**18h**
0.9	0.4	0.6	0.7	0.3	0.7	1.4	0.6	1.3	1.4	2.3	1.1	0.8
-2.5	-3.3	-3.6	-3.3	-3.6	-3.2	-2.8	-2.9	-3.2	-3.1	-2.5	-2.7	-3.2
-1.1	-1.6	-1.4	-1.7	-1.4	-1.5	-1.7	-1.5	-2.3	-2.3	-2.1	-1.2	-1.9
1.2	1.7	2.7	2.5	1.3	1.3	0.9	-0.4	1.3	1.6	1.3	0.0	0.7
0.8	0.1	-0.1	0.0	0.6	-0.3	-0.5	0.1	-1.0	-0.9	0.0	0.1	-1.1
10.5	10.7	10.4	10.7	10.9	10.5	9.1	7.7	6.8	8.4	10.2	7.8	9.8
0.8	0.5	1.1	1.2	0.4	1.0	0.9	0.3	0.3	0.3	0.8	-0.5	-0.3
8.9	8.4		8.6	8.5	8.8	8.7	7.9	8.4	8.8	8.9	8.5	8.3
9.1	8.8	6.5	7.4	9.6		9.9	9.4	10.0	10.6	11.5	11.4	10.0
9.0	8.8	8.7	9.2	9.4	9.8	9.5	9.0	9.6	10.0	10.6	9.7	9.2
9.3	7.7	8.4	7.9	7.9	7.1	6.7	6.5	6.4	6.2	7.3	5.6	5.7
5.3	4.7	4.0	4.4	4.6	4.6	4.9	4.8	4.7	5.0	5.6	5.0	4.4
6.8	6.1	4.9	5.5	6.3	6.0	6.6	6.6	6.6	7.0	7.2	8.1	5.9
0.4	0.4	0.6	0.4	0.6	0.1	0.6	0.5	0.2	0.6	-0.2	-0.1	0.3
5.4	5.1	5.0	5.4	5.3	5.5	5.3	6.0	6.1	6.6	6.5	5.3	4.7
7.0	4.8	5.2	4.5	4.1	4.4	4.3	4.8	5.2	5.8	5.7	4.3	4.1
10.2	8.8	8.6	8.2	8.2	8.2	7.6	7.6	7.7	7.9	8.3	8.2	7.5
11.1	11.4	11.2	10.9	10.8	10.1	10.2	8.9	9.6	10.0	9.8	10.3	9.5
0.1	0.1	0.2	0.5	0.3	-0.2	-0.8	-0.7	-0.5	-0.6	-0.1	-0.8	-1.3
4.5	5.1	4.3	4.7	4.5	4.9	4.7	4.7	4.5	4.3	5.0	4.4	4.1
0.6	0.9	0.7	0.7	0.6	0.5	0.6	0.5	0.3	0.3	0.3	0.1	-0.1
0.0	0.1	-0.3	-0.1	0.0	-0.2	-0.2	-0.2	-0.5	-0.3	-0.7	-0.5	-0.8

Table A.11: ΔC_T values for last 22 genes for the isograft Patients

d 1	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
2.0	2.2	1.4	0.9	1.1	3.2	2.0	1.1	2.7	2.5	3.0
-2.8	-2.8	-2.4	-2.4	-2.2	-0.3	-1.9	-2.7	-0.3	-1.6	-0.6
-1.5	-1.4	-1.3	-1.8	-1.6	-0.9	-1.3	-1.8	-1.4	-0.6	-1.0
1.8	2.5	0.4	-0.1	0.2	1.1	0.8	0.0	0.5	0.9	1.1
-0.3	-0.3	0.2	0.0	0.4	1.8	0.4	0.2	1.6	0.9	1.1
8.1	9.8	9.3	9.2	8.7	7.3	9.4	9.4	8.9	10.6	9.5
0.6	1.1	0.6	-0.6	-0.3	1.8	0.1	-0.6	2.1	0.2	1.0
8.8	9.5	8.5	7.4	6.7	7.8	7.6	7.0	8.3	8.5	8.0
10.2	10.7	9.3	8.1	10.1	10.2	9.7	8.3	11.5	13.1	10.9
9.9	10.4	11.3	11.8	10.9	10.8	10.8	9.3	11.7	12.5	10.5
7.2	7.3	6.9	6.8	6.5	8.1	6.9	6.2	7.3	7.4	7.1
5.7	6.2	5.9	5.4	5.3	6.6	6.7	4.9	5.8	6.4	6.5
6.7	7.4	8.2	7.4	6.8	9.3	6.8	6.8	7.8	8.3	8.2
1.6	1.2	0.9	-0.1	0.2	0.9	1.6	0.0	0.1	0.9	1.1
6.1	6.7	6.2	5.2	5.1	6.6	5.6	4.4	5.5	5.7	5.4
5.1	6.8	5.1	4.0	4.3	7.6	4.5	3.3	4.3	5.2	5.0
8.3	8.7	8.3	7.9	7.7	9.1	9.3	8.0	9.1	9.0	8.4
9.8	10.1	9.5	11.2	10.5	9.5	10.6	8.8	10.7	12.2	10.3
0.0	-0.3	-0.3	-1.2	-1.0		-0.6	-1.1	-0.8	-0.2	-0.5
4.8	5.2	5.1	3.3	3.7	4.3	3.8	4.0	4.3	4.7	4.2
0.4	0.3	0.6	0.1	0.3	0.9	1.1	0.2	2.0	1.6	1.3
0.0	-0.3	0.0	-0.6	-0.4	0.4	0.0	-0.7	0.7	1.0	0.4

Table A.12: ΔC_T values for last 22 genes for the isograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**
1.0	2.3	3.3	2.9	2.3	3.3	2.3	2.8		2.3	2.1	1.8
-1.9	-2.4	-2.1	-1.3	-2.3	-0.7	-2.0	-1.5		0.6	-1.9	-2.2
-0.5	-0.7	-1.5	-1.1	-1.6	-1.3	-1.4	-1.1		1.4	-0.4	-0.4
0.7	1.0	1.2	0.4	1.2	0.3	0.4	0.3	0.9	0.6	0.3	0.0
0.6	0.7	0.4	1.1	0.9	0.7	0.8	0.6		-0.1	0.3	0.6
9.3	10.4	9.8	9.6	10.3	9.3	10.1	10.5		6.9	9.9	9.0
-0.9	-0.3	0.7	-0.1	-0.1	-0.9	-0.5	-0.8		-2.0	-1.4	-1.4
8.3	8.2	7.3	7.1	8.2	3.5	8.6	8.1	8.5	5.6	7.7	7.6
12.6	10.0	11.1	11.7	11.9	10.1	11.3	8.6	8.7	11.5	5.4	10.1
11.8	10.9	11.5	11.3	9.8	10.8	10.8	9.1	8.9	10.5	11.0	10.3
7.2	7.2	6.6	7.0	7.0	5.5	7.3	6.9		4.6	6.7	6.4
5.4	5.7	5.8	6.2	5.4	5.6	5.9	6.0		4.9	5.3	4.9
7.7	7.2	7.8	8.0	6.9	7.5	8.0	8.0		8.0	7.8	6.9
0.2	0.7	12.0	0.8	0.9	0.3	0.3	0.5		-0.8	-0.6	-0.3
4.4	5.2	5.7	5.0	5.3	5.1	5.2	5.0	4.7	3.6	3.5	3.7
3.6	3.7	4.4	3.8	4.6	4.2	3.7	3.8		2.4	2.4	2.6
7.8	8.4	8.3	8.5	8.6	9.4	8.5	9.3		6.4	7.3	7.3
12.4	11.2	11.0	11.7	11.8	11.0	12.4	11.8		9.6	11.4	9.4
-0.7	-0.3	-0.6	-0.8	-0.2	-1.3	-0.6	-0.9	-0.6	-2.4	-1.3	-1.1
3.6	4.7	4.6	3.8	4.8	4.2	3.8	3.5	3.0		2.8	3.2
0.9	0.9	1.1	1.0	1.1	0.8	1.3	1.0	0.7		0.7	0.6
0.2	0.1	0.3	0.2	0.4	-0.2	0.5	0.2	-0.3	-1.0	-0.2	-0.2

Table A.13: ΔC_T values for first 31 genes for the allograft Patients

dCT	**ref** C_0 **avg**	**0 h**	**0 h**	**0 h**	**1 h**	**1 h**	**1 h**	**1 h**	**1 h**	**3 h**	**3 h**
GAPDH	**0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
C1q-a	**5.3**	6.4	6.4	7.2	6.2	5.9	5.7	6.2	6.0	5.8	5.7
C1q-b	**4.4**	5.0	4.8	5.7	4.5	4.9	4.6	5.1	5.2	4.3	4.2
C1q-c	**7.0**	7.5	7.6	8.3	7.1	6.1	7.9	7.9	7.7	5.9	7.4
C1-Inh	**2.5**	3.0	2.8	3.7	2.1	3.0	2.8	3.0	3.0	2.4	2.4
compl C3	**3.8**	6.2	4.4	7.2	5.2	7.6	5.6	7.2	7.1	6.4	4.6
C3aR	**7.7**	8.5	8.0	9.0	8.4	9.2	7.4	8.6	8.2	8.0	7.8
C4	**6.3**		5.9		6.5	9.0	7.5	8.1	8.3		5.7
C5aR	**6.7**	7.1	8.0	8.9	7.8	8.8	7.3	8.5	8.5	6.7	5.9
C9	**10.2**	11.4	11.0	13.4	12.8		10.8	10.8	10.1	12.3	11.0
compl H	**3.0**	4.0	3.3	4.1	3.2	4.8	3.0	3.6	3.6	3.8	4.1
DAF-1	**5.7**	7.2	6.9	6.8	6.3		6.1	7.1	6.9	6.6	7.1
Pro-C5a	**12.7**	12.9	14.4	15.2	12.5		14.2	12.7	13.3	12.9	11.8
properdin	**6.1**	7.0	5.6	7.0	6.7	7.0	6.5	7.6	7.5	7.4	4.9
APP	**1.8**	2.4	2.2	3.2	1.8	3.0	1.9	3.0	2.9	1.9	2.3
CRP	**10.6**	11.5	11.6	12.9	12.1	13.6	11.2	10.7	10.8	11.9	10.7
MacManR	**5.4**	6.2	6.1	7.2	6.0	7.4	6.0	6.9	6.9	6.2	6.1
Man6-PR	**4.3**	5.1	5.2	5.6	5.0	6.0	4.9	5.8	6.2	5.7	4.8
MBL-2	**10.6**	11.9	11.7	12.9	14.9	11.8	11.6	10.7	10.8	11.7	10.6
SAA-2	**12.0**	10.7	10.3	13.3		11.7	8.6	10.3	9.7	11.4	10.4
SAA-4	**11.5**	11.1	13.7	14.4	16.0	12.9	11.9	11.2	11.0	12.0	11.5
SAP	**10.6**	11.4	13.4	14.1	14.8	13.1	12.4	11.0	11.2	12.6	10.4
G-CSF R	**9.5**	9.0	9.9		9.2	9.4	8.5	8.9	8.9	8.4	8.7
GM-CSF R2a	**9.0**	8.3	9.4	10.6	9.5	9.8	8.3	9.0	9.1	8.5	6.7
IFN-b	**11.4**	12.2	14.6	14.4	16.2		12.9	12.0	11.8	11.9	10.8
IFN-g	**5.7**	6.2	5.9	7.2	6.3	6.7	5.7	6.6	6.6	6.1	5.9
IL-1a	**8.9**	9.1	8.7	10.2	9.8	8.9	7.3	8.7	8.4	8.1	8.0
IL-1b	**9.6**	8.2	10.8	10.2	9.9		7.7	9.4	8.7	5.7	3.8
IL-2	**11.5**	10.3					7.6	8.9	8.3	11.4	10.6
IL-6	**10.5**	8.1				8.7	6.4	8.9	7.8	3.0	3.1
IL-10	**11.1**	12.2	12.5	13.5			10.1	11.4	10.8	11.0	9.5

Table A.14: ΔC_T values for first 31 genes for the allograft Patients

3 h	**3 h**	**3 h**	**6 h**	**6 h**	**6 h**	**9 h**	**9 h**	**9 h**	**12 h**	**12 h**	**12 h**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
6.5	6.2	6.5	6.3	6.5	6.7	5.6	5.3	5.7	5.3	6.1	4.6
4.5	5.0	5.2	4.5	4.7	4.6	3.9	3.7	4.3	4.0	4.1	2.3
8.0	7.4	7.7	7.6	8.3	7.6	6.8	6.8	7.5	7.7	6.8	5.9
2.3	2.9	3.1	2.6	3.0	2.7	2.7	2.5	2.5	2.9	3.0	2.2
3.9	7.6	8.2	4.9	8.1	5.7	6.2	4.4	5.7	5.4	6.8	3.9
7.5	8.2	9.1	8.2	8.7	8.1	7.3	6.1	8.4	7.6	6.1	4.9
6.0		8.7	7.1	9.5	7.3	7.7	6.9	7.5	7.0	8.3	6.2
5.8	7.5	7.9	5.7	6.9	5.5	5.6	4.6	7.5	5.8	4.3	3.9
11.1	9.2	11.3			11.1			10.5		10.1	9.5
2.9	4.3	4.2	3.3	4.5	2.9	3.0	2.7	3.7	3.9	3.1	2.5
6.0	7.7	7.2	6.9	7.6	7.3	6.7	7.3	7.0	7.7	7.8	6.3
13.4	10.3	14.0	13.6	14.1	13.3	12.4	13.1	13.6	12.5	12.5	10.7
5.4	7.4	7.8	5.8	7.2	6.2	5.8	4.5	5.1	5.5	4.3	3.2
2.0	2.7	3.0	2.1	2.8	2.0	2.0	1.8	2.7	2.4	2.2	1.2
10.9	8.7	11.7	11.0	12.4	11.7	10.4	10.2	10.9	10.3	10.1	10.2
5.3	7.8	7.5	5.7	6.5	5.1	5.2	4.3	6.4	6.3	4.7	3.7
4.5	5.8	6.0	4.8	5.5	4.5	4.9	4.2	5.7	4.9	4.8	3.8
11.0	8.4	11.3	10.8	12.3	11.1	10.8	10.3	10.7	10.8	10.2	9.7
9.2	8.6	9.9	9.6	12.2	10.4	9.3	7.6		9.8		9.0
11.7		12.1	11.9	12.7	12.9	10.9	10.5	11.6	10.9	10.4	10.6
12.0	9.6	11.5	11.8	12.2	12.0	10.9		11.9	11.0	10.5	10.0
8.8	8.6	9.2	8.2		8.7	7.8	6.9	9.4		6.4	6.8
6.3	7.7	9.2	6.6	9.0	6.6	6.2	5.0	9.1	6.4	5.5	4.5
11.7	9.7	12.7	11.7	12.0	12.7	11.3	11.5	11.9	11.5	10.9	10.6
5.9	6.4	6.4	4.7	6.9	5.3	5.2	4.2	6.6	4.1	4.8	4.0
7.3	8.3	9.1	7.9	9.4	8.2	8.1		8.9	8.0	7.2	7.4
3.0	6.9	6.9	3.0	5.2	3.3	3.7	3.2	7.9	3.8	5.4	3.9
9.9	8.6	10.0	9.5	10.8	10.9	8.3	6.6	10.2	9.8	8.0	
1.3	5.2	5.2	1.7	3.9	1.8	3.6	2.1	5.0	2.9	3.7	4.1
9.1	9.2	11.8	9.6	11.4	10.1	10.0	10.2	9.7	9.2	9.8	9.2

Table A.15: ΔC_T values for first 31 genes for the allograft Patients

18 h	**18 h**	**18 h**	**d 1**	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
4.7	5.7	5.3	5.1		5.2	4.2	4.1	4.6	2.9	3.1	3.6
3.0	4.2	3.9	3.8	3.6	3.6	2.7	2.9	3.3	1.7	1.7	2.2
6.0	6.9	7.5	6.3	6.7	6.9	5.3	4.8	6.2	3.8	3.7	5.6
2.2	2.8	3.0	3.3	3.6	2.9	2.9	3.7	3.1	2.3	2.4	1.9
5.2	5.4	7.3	5.0	6.8	5.5	4.0	5.2	5.0	4.6	4.1	3.6
6.5	6.4	7.7	6.3	7.0	5.7	5.7	6.4	5.1	5.4	5.8	4.4
6.6	6.7			9.8	6.3			6.7	6.8	6.9	5.7
4.2	4.4	5.9	4.1	5.0	4.1	4.1		4.9	4.6	4.6	4.2
9.8	10.2	8.7	12.4	11.7	10.0	11.5	8.8	7.6	11.2	11.1	11.3
3.4	3.1	4.2	4.0	5.0	3.0	3.5	4.5	3.4	3.7	3.7	2.9
7.3	7.3	8.7	6.9	6.8	7.3	5.9	7.2	6.8	6.8	6.6	6.0
11.5	12.2	10.3	13.0	14.1	11.5	12.3	10.1	9.5	11.7	12.3	13.1
4.9	4.5	5.4	5.4	5.5	4.6	4.5	5.2	5.0	4.1	4.3	3.7
1.7	2.0	3.2	2.4	2.5	1.9	2.1	2.5	2.3	2.0	1.9	1.5
9.9	10.4	8.7	11.9	10.7	9.9	11.0	8.1	7.5	11.1	10.8	11.2
4.8	5.4	6.1	5.2	5.7	5.1	4.4	5.3	5.2	4.6	4.4	4.2
4.3	4.6	5.4	4.5	5.3	4.2	4.2	4.8	4.6	4.2	4.7	3.7
10.1	10.3	9.0	11.8	11.4	10.3	11.0	8.0	8.3	11.4	11.1	11.8
8.6	10.1	9.3	11.5	10.0	8.6	10.3	7.7	8.0	10.8	10.2	8.1
10.0	11.4	9.3	12.5	11.5	10.8	11.6	9.4	9.2	11.5	11.8	11.3
10.4	11.0	9.1	12.6	11.0	10.0	11.2	8.5	7.9	11.1	11.3	12.2
6.9	7.9	7.8	5.3	6.6	4.7	5.1	5.9	5.6	6.5	6.6	6.6
4.8	4.8	6.0	6.9	8.1	6.5	6.4	6.6	6.4	7.8	7.3	6.8
10.8		9.6	12.6	12.1	10.4	12.0	9.0	8.2	11.8	13.2	12.5
4.6	4.5	5.6	4.7	5.5	5.2	5.0	5.6	3.9	4.0	4.5	3.9
8.3	7.5	7.2	8.0	9.4	7.8	7.7	6.5	6.9	8.9	8.9	7.8
4.5	3.6	3.6	4.2	6.3	4.7	3.6	3.3	6.1	4.2	5.6	5.4
9.5	10.3	8.5	9.9	9.0	5.9	8.6	7.2		9.3	10.1	
4.5	5.6	5.1	5.9	6.8	5.2	4.9	5.6	5.0	5.7	6.0	5.1
9.5	10.2	9.1	11.2	11.1	9.9	10.5	9.0	8.7	10.7	11.0	10.2

Table A.16: ΔC_T values for first 31 genes for the allograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**	**d 7**
0.0	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
4.8	3.7	3.4	3.3	3.0	2.6		2.8	2.6	3.2	2.5	2.6	2.7
2.2	1.0	1.2	2.1	1.4	1.2	0.9	1.2	1.2	0.7	0.8	0.7	0.8
5.8	4.1	5.1	4.0	4.1	5.0	3.2	3.7	5.0	3.1	3.4	3.5	4.8
2.6	1.3	1.3	2.3	1.4	1.9	3.0	2.9	1.9	0.8	2.3	2.4	2.3
	2.3	3.3	3.6	2.0	2.9	5.7		2.9	1.1	3.3	3.1	2.4
6.3	5.4	2.9	6.2	6.1	4.7	4.1	4.9	4.7	4.0	4.7	4.6	3.4
6.7	5.1	5.4	5.7	4.9	5.3	5.4	5.7	5.3	4.5	4.9	4.6	4.7
5.1	4.9	2.7	5.5	5.1	4.1	3.2	3.9	4.1	3.4	3.6	3.7	2.9
	11.1	11.1		11.1	9.6			9.6	12.2	11.6	11.2	
	3.8	3.0	4.0	4.2	5.0	5.9	5.8	5.0	3.5	6.1	5.6	5.0
7.1	7.2	6.3	7.5	7.8	6.1	8.4	8.8	5.9	6.1	8.2	8.0	5.9
9.4	11.3	11.9	9.0	11.7	11.2	9.9	9.8	11.2	13.7	11.9	12.1	10.3
4.7	4.6	2.4	5.0	5.2	3.8	4.1	4.7	3.8	2.4	3.9	4.1	2.9
2.3	1.6	0.6	2.8	1.9	1.7	2.1	2.4	1.7	0.4	2.1	1.9	1.4
7.6	10.5	10.7	7.3	10.5	9.9	8.8	8.3	9.9	10.9	12.0	10.6	9.2
5.7	4.8	3.9	6.5	6.1	6.7	7.0	7.2	6.7	5.3	7.1	6.2	6.2
5.1	3.9	2.8	4.4	3.5	3.5	3.5	3.9	3.5	2.0	3.2	3.2	3.4
8.4	11.0	10.0	7.8	11.0	9.7	8.8	8.3	9.7	11.5	11.9	10.7	8.7
8.3	10.3	9.3	8.1		7.3	9.4	9.0	7.3	11.9	10.4	10.1	6.9
9.0		11.1	8.9	11.7	10.1	9.4	9.2	10.1	12.3	12.2	11.4	9.3
7.8	11.0	11.5	7.5	10.3	10.1	9.1	8.7	10.1	11.9	11.2	10.8	9.8
5.9	6.9	5.6	6.6	6.0	7.1	6.1	6.6	5.8	5.0	7.2	5.1	8.7
7.1	7.9	7.5	6.3	6.0	8.1	6.1	6.7	8.1	5.7	6.6	6.1	6.4
8.2	13.0	11.2	7.3	11.8	9.1	9.0	9.2	12.7	12.3	12.1	11.5	8.8
4.4	2.9	2.6	2.6		1.8	2.7	3.5	1.2	0.7	2.3	2.3	2.2
6.9	9.1	7.5	7.4	8.8	7.3	7.6	7.6	7.3	8.5	8.6	8.8	6.4
3.6	6.0	4.3	5.2	5.6	4.0	4.1	4.6	4.0	3.6	4.1	4.4	2.7
7.2	9.5	4.7	6.9	8.4		8.3	7.8		8.6	8.6	8.7	4.1
6.1	6.3	4.5	5.9	7.2	5.0	5.4	5.1	5.2	4.6	4.9	5.0	4.4
9.3	9.6	10.6	8.3	8.8	7.8	8.6	8.5	7.8	7.4	8.5	8.8	7.8

Table A.17: ΔC_T values for next 29 genes for the allograft Patients

dCT		**0 h**	**0 h**	**0 h**	**1 h**	**1 h**	**1 h**	**1 h**	**1 h**	**3 h**	**3 h**
IL-11	**11.5**	9.3	10.5	12.6	12.2	11.8		9.4	9.1	10.9	10.1
IL-12 p35	**10.8**	10.5	11.6	13.2	13.3		9.7	10.5	9.8	11.6	10.1
IL-12 p40	**10.7**	10.4	13.0	13.1				10.8	10.3	11.5	10.3
TNF-a	**10.9**	10.0	12.3	12.7	12.9	12.8	10.7	10.7	10.5	9.9	9.1
granz B	**10.8**		11.6	12.7	13.2	11.8		10.6		9.9	10.0
granz D	**9.6**	10.2	11.8	11.9	13.5	14.8	10.8	9.6	9.2	10.3	9.3
granz E	**9.4**	9.6	11.3	11.7	15.1		10.7	9.0	8.9	9.7	8.8
granz G	**9.3**	9.0	11.4	11.9	14.9			8.9	8.9	9.8	8.9
perforin	**9.8**	10.1	11.3	11.7	11.0	12.6	9.6	9.1	9.7	9.6	10.5
serglycin	**4.6**	3.6	4.6	4.8	4.1	5.0	3.9	4.8	4.6	3.6	2.6
TLR-1	**12.0**	9.2	9.4	10.2	10.4	11.1	10.7	10.5	10.1	10.3	9.7
TLR-2	**8.5**	7.6	7.9	9.0	8.5	9.5	7.7	8.7	8.6	7.2	6.0
TLR-3	**13.4**	8.1	8.7	9.2	8.1	11.2	8.8	9.3	9.6	8.0	8.8
TLR-4	**7.0**	6.7	6.5	6.4	6.7	8.4	5.5		6.2	6.5	7.3
TLR-5	**6.5**	7.2	6.3	7.8	7.2	7.4	4.7	6.3	6.1	7.4	7.4
TLR-6	**9.0**	9.6	10.0	10.6	10.0	12.0	9.8	10.4	10.4	10.4	9.9
TLR-7	**7.6**	7.2	7.1	7.9	6.9	8.2	7.2	7.8	7.5	6.4	6.0
TLR-8	**11.0**	10.2	9.6	10.2	9.5	11.8	9.4	9.9	8.0	10.1	9.2
TLR-9	**9.7**	9.3	10.0	10.4			10.4	9.7	9.6	10.1	9.8
Aldo-a	**-0.4**	-0.3	-0.7	-0.4	-0.4	-0.4	-0.9	0.1	-0.1	-0.3	-0.7
CARAT	**8.3**	6.1	6.2	6.6	6.8	6.8		6.8	7.0	7.2	6.6
Cat D	**1.0**	1.0	0.7	1.3	1.1	1.1	0.7	1.2	1.2	1.5	0.6
CK	**-0.3**	-0.5	-0.6	0.1	-0.1	-0.3	-1.1	-0.8	-0.6	-0.4	-0.6
GDH	**5.5**	6.9	5.5	8.7	5.6	7.6	7.2	7.3	7.7	6.8	5.1
IDO	**10.6**			11.8	11.8					10.5	10.0
LDH-2B	**0.4**	0.7	0.4	0.6	0.7	0.9	0.1	0.9	1.0	0.7	0.4
MEP	**2.6**	2.9	2.5	2.9	2.3	3.2	2.3	3.3	2.9	2.5	4.1
ANK-1	**11.1**	10.3	11.2	11.9	12.2		10.7	10.0	10.2	10.7	11.2
b-actin	**1.6**	2.5	2.6	3.8	2.1	3.1	2.3	3.2	3.4	2.4	2.2

Table A.18: ΔC_T values for next 29 genes for the allograft Patients

3 h	**3 h**	**3 h**	**6 h**	**6 h**	**6 h**	**9 h**	**9 h**	**9 h**	**12 h**	**12 h**	**12 h**
10.7	9.0	10.0	10.3	11.7	9.5	10.0	7.8	11.2	10.3	8.1	7.5
10.3	8.6	10.9	10.4	12.3	10.4	10.4	9.5	10.7	9.5	9.3	9.5
10.8	8.4	11.8	12.0	11.7	11.3	10.1	10.1	11.3	10.8	9.6	9.7
8.8	9.0	11.2	10.1	10.6	10.2	10.3	9.4	11.4	9.5		8.4
10.0	9.0	11.1	9.3	11.1	10.6	9.8	6.2	11.7	8.6		9.0
9.2	7.0	10.3	10.3	10.9	10.2	9.6	10.0	10.0	9.3	9.1	8.7
8.9	6.3	10.3	9.7	10.2	10.0	8.6	9.1	9.5	8.9	8.6	8.3
9.1	6.4	10.0	9.8	10.6	9.8	8.9	8.8	9.2	8.7	8.6	8.2
9.5	8.3	10.3	9.6	11.1	10.2	9.9	7.5	10.6	9.8	9.2	9.0
2.7	4.2	4.3	1.9	3.8	2.2	2.1	1.0	4.3	1.7	0.5	1.6
9.8	8.4		9.8	11.0	8.8	9.3	7.9	9.6	8.8	7.7	6.2
5.4	7.7	7.9	5.9	6.9	5.2	5.6	5.2	7.9	5.5	6.9	5.3
8.4	8.8	9.3	7.5	8.6	7.5	7.2	8.1	8.7	8.3	9.1	7.7
5.8			5.9	7.4	5.3	5.5	4.5	7.3	5.9	4.1	4.0
6.8		6.4	6.7	7.4	6.7	5.9	5.3	6.9	6.4	5.7	5.8
9.1		10.4	9.0	10.5	9.1	8.8	8.1	10.4	8.9	8.1	7.2
6.0		7.0	5.9	6.3	6.2	5.4	5.9	7.0	5.7	5.5	5.6
9.1		10.4	8.5	9.7	8.3	8.1	7.4	10.1	8.1	8.1	7.1
9.5		10.4	9.3	10.4	9.7	9.2	8.6	10.5	8.4	8.9	6.9
-0.7		-0.1	-0.6	-0.1	-0.4	-0.7	-0.7	-0.1	-0.2	-0.3	-0.5
6.8		6.9	6.8	7.2	6.8	6.6	6.3	7.9	6.5	7.4	6.9
1.1		1.7	1.0	1.8	1.2	1.2	0.7	1.7	1.3	1.4	0.2
-0.3		-0.6	-0.6	-0.2	-0.3	-0.9		0.2	0.1	-0.5	0.3
6.4	8.2	7.2	6.0	8.7	7.2	6.7	6.3	7.7	6.6	7.7	6.8
10.4	8.8		10.1	10.8				10.5	9.6		8.9
1.2	1.9	1.2	0.7	0.9	0.1	0.4	0.2	1.2	0.7	1.1	1.0
2.4	3.2	3.2	2.2	2.7	2.3	1.6	2.0	3.3	2.1	0.7	1.4
11.0		11.3	10.8	11.8	11.1	10.4	10.4	11.2	10.5	10.0	9.6
2.1	3.0	3.1	1.5	2.7	1.3	1.4	0.7	2.3	1.4	0.2	-0.4

Table A.19: ΔC_T values for next 29 genes for the allograft Patients

18 h	**18 h**	**18 h**	**d 1**	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
6.7	7.9	7.5	7.6	9.0	7.4	8.8	8.1	7.9	10.4	11.4	
9.6	10.2	8.9	12.1	11.3	10.0	11.2	8.7	8.6	11.7	11.7	10.4
9.6	10.4	8.8	12.4	12.1	9.2	11.2	8.4	7.6	11.3	12.1	10.1
9.1	8.8	8.9	9.6	10.3	9.7	9.3	8.6	8.7	10.1	9.9	9.4
10.3	9.9	9.2	10.8	9.7	9.8	9.6	8.2		7.8	8.5	7.6
8.5	9.1	7.5	9.7	9.3	8.8	8.3	6.3	6.5	9.3	10.5	9.1
8.1	8.8	6.9	9.5	8.6	8.1	8.1	5.8	5.6	9.2	10.0	9.4
7.9	8.6	6.8	9.0	8.6	8.1		5.7	5.4	8.6	9.9	9.3
9.2	10.3	8.8	10.7	10.8	10.9	9.9	8.4	8.3	9.6	10.7	
0.8	1.0	1.7	1.4	3.4	2.2	2.4	2.9	3.8	2.8	0.6	3.7
7.0	7.2	8.2	8.6	9.9		8.8	9.4	6.5	8.3	8.9	
5.7	6.0	6.9	5.9	8.0	5.8	5.5	6.6	6.1	5.6	6.3	5.1
10.0	9.6	9.5	10.9	11.2		10.3	10.3		10.1		6.9
4.6	4.5	5.8	5.8	7.6		6.0	6.4	4.0	6.2	6.3	3.9
6.7	6.5	6.1	7.1	7.7	6.1	7.1	7.0	5.1	6.9	7.7	5.6
8.0	8.4	8.4	8.0	9.5	7.4	7.9	7.9	7.3	8.7	8.8	7.6
6.4	6.8	7.0	7.4		6.6	6.6	6.6	6.5	6.5	6.2	5.7
8.6	8.3	8.9	11.3		11.0	7.6	8.5	8.0	12.5	9.7	11.7
8.4	8.6	9.4	9.8	10.4	9.5	9.7	10.6	8.5	7.6	9.2	7.6
-0.5	-0.4	0.3	0.2	0.3	-0.6	0.3	0.7	-0.1	0.3	-2.7	-0.6
6.6	6.6	7.9	11.4	10.2	8.4	9.2	10.1	8.5	10.3	7.8	9.8
0.7	0.6	1.4	0.9	1.2	0.6	0.7	1.4	1.2	0.3	-2.1	0.5
-0.1	0.1	0.5	1.4	1.3	0.9	2.6	3.9	1.5	3.7	0.3	2.6
6.9	6.7	7.6	6.3	9.3	6.6	7.0	9.4	7.3	8.6	6.5	7.2
9.2	10.3	8.4	11.3	10.1	10.3	9.8	8.1	8.9	10.3	11.1	
1.1	1.0	1.7	1.2	2.7	0.9	1.7	5.1	1.7	3.4	3.9	2.5
1.5	1.5	2.0	1.5	2.3	1.8	1.9	2.5	2.7	1.4	-0.7	1.8
9.8	10.5	9.3	11.6	11.9	10.4	10.6	8.8	9.3	11.7	9.1	11.9
0.4	0.4	1.5	0.0	-0.3	0.0	-0.3	0.3	1.0	0.0	-0.1	0.2

Table A.20: ΔC_T values for next 29 genes for the allograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**	**d 7**
8.5	9.9	8.6	8.4	11.2	9.5	7.9	7.0	9.5	8.8	7.5	7.8	7.7
8.9	11.4		8.9	11.7	9.0	9.4	9.1	9.0	10.3	12.1	11.4	8.5
7.9	9.5	11.6	7.8	10.1	7.5	8.7	8.7	7.5	9.2	9.9	9.9	8.4
8.3	7.8	8.1	7.9	7.2	6.7	7.6	8.2	6.7	6.6	7.8	7.6	6.9
7.2	6.0	6.1	4.9	3.8	0.7	3.4	3.7	2.8	0.6	1.6	2.6	1.8
6.3	10.0	8.9	6.3	9.2	6.1	7.1	6.8		6.9	7.0	8.8	5.7
5.7	9.4	8.5	5.9	8.6	5.8	6.7	6.5	8.1	6.4	6.6	8.2	5.4
5.7	9.5	8.7	6.0	8.7	5.9	7.1	6.5	7.6	6.3	6.7	8.4	5.5
8.1	8.3	7.9	7.4	6.7	3.8	5.2	6.2	5.1	2.9	4.3	5.2	4.5
3.0	3.3	1.6	3.2	3.2	1.7	1.3	2.4	2.0	1.0	1.4	2.2	0.9
8.7	8.2	5.0	8.1	8.0		8.2	8.9		11.5	7.0	6.9	5.3
6.2	5.5	4.5	5.1	4.9	4.5	4.4	5.2	4.5	5.8	4.1	4.8	4.1
10.3	9.2	6.5	10.0	9.1		11.1	11.6		13.7	10.9	10.5	7.2
6.2	5.8	3.4	6.1	6.1	4.4	5.8	5.9	4.4	7.4	5.4	5.3	3.8
6.5	7.0	4.8	6.8	7.6	5.0	6.5	7.2	5.0	5.8	6.8	7.8	5.0
7.7	8.2	6.8	7.5	8.1	7.2	7.3	7.8	7.2	6.6	7.7	8.0	6.7
6.8	6.2	4.6	6.6	6.9	5.5	5.9	6.4	5.5	5.7	5.5	6.2	5.1
10.4	11.8	9.6	11.5	9.9	11.6	9.1		6.2	5.9	7.1	5.3	7.5
8.6	7.0	6.6	7.3	7.1	5.9	6.4	8.1	5.9	5.6	5.9	6.4	6.1
0.6	0.2	-0.3	1.2	0.0	-0.6	0.3	0.7	-0.5	-0.9	0.3	0.2	-0.4
9.5	10.9	10.0	9.2	11.2	10.1	10.8	11.0	10.8	10.9	11.8	11.6	9.8
0.9	0.7	-0.7	1.5	1.3	0.6	0.9	1.1	0.5	0.2	0.6	0.6	0.4
			4.7	4.2	4.1	5.0		5.2	3.9	6.1	6.7	6.6
7.6	9.2	10.1	8.1	9.1	8.1	8.1	9.2	8.1	6.9	8.5	9.7	8.6
7.8	8.0	6.4	7.5	7.5	4.4	6.8	7.3	6.3	5.2	6.1	6.4	5.0
3.5	5.0	6.0	4.3	4.3	3.3	5.7	6.9	3.3	2.9	4.4	5.9	5.4
2.4	1.9	-0.2	2.6	2.5	3.0	1.8	2.5	2.2	2.1	2.7	2.3	1.1
9.2	10.6	8.6	9.4	11.4	9.6	9.3	9.0	11.6	10.5	12.0	11.1	8.2
0.7	-0.6	-0.8	0.6	-0.4	-0.1	0.2	0.1	-0.5	-1.8	-0.7	-0.6	-0.3

Table A.21: ΔC_T values for last 22 genes for the allograft Patients

dCT	**Co ref**	**0 h**	**0 h**	**0 h**	**1 h**	**1 h**	**1 h**	**1 h**	**1 h**	**3 h**	**3 h**
gelsolin	**0.2**	-0.3	-0.3	0.7	-0.5	0.7	-0.4	0.1	0.4	0.0	-0.7
MLC-2	**-2.2**	-2.8	-2.7	-2.8	-2.8	-2.8	-2.9	-2.7	-2.3	-3.0	
crystallin	**-0.8**	-1.0	-1.4	-0.8	-1.0	-1.5	-1.3	-0.9	-0.8	-1.1	-1.5
GSH Px	**1.0**	1.8	-0.4	1.7	0.8	1.5	0.7	2.4	2.2	1.3	0.7
Hsc70	**0.3**	0.4	0.6	0.6	0.0	1.1	-0.6	0.4	0.4	0.7	0.4
iNOS	**9.0**	9.3	10.3	10.5	9.9	11.3	10.5	10.9	11.3	10.7	10.7
MGP	**1.1**	1.1	0.3	1.8	0.2	1.0	0.8	1.2	1.0	0.9	0.4
DHFR	**8.4**	8.4	8.3	9.3	8.9	9.6		8.4	8.7	9.2	8.5
FOLbp3	**10.6**			11.1		8.7	7.7			10.6	10.3
GTP-CH I	**9.8**	9.5		11.3	10.3	8.9		9.0		9.7	9.5
MTHFD2	**8.3**	8.1	9.1	9.6	7.8	8.7	7.9	8.8	8.8	8.5	7.8
PTPS	**4.3**	5.1	4.6	5.0	4.7	5.0	4.2	4.9	4.8	4.6	4.6
sepiapterin R	**5.8**	6.2	5.9	6.9	6.3	6.4	4.4	5.9	5.6	6.4	6.8
B2-M	**1.2**	1.4	1.0	1.4	0.8	1.8	0.6	1.7	1.4	1.2	0.7
I-A-b one	**9.2**	14.4	15.2	13.9		18.4		14.5		7.9	7.6
I-E-b	**4.4**	5.5	5.2	5.6	5.2	6.5	4.9	5.8	5.8	5.3	4.2
MHC-1	**8.0**	9.0	8.7	9.0	8.7	10.3	8.8	8.9	9.1	9.6	9.4
BLR-1	**11.9**	9.7	10.7	12.1	11.7		11.4	10.2	10.2	11.2	9.2
EF-1a	**0.2**	0.1	-0.4	0.8	-0.4	0.2	-0.2	0.4	0.3	-0.3	0.0
GAS-6	**4.2**	4.7	3.6	5.2	4.0	5.6	4.0	5.1	5.2	4.7	4.5
rp L8	**1.5**	1.0	1.1	1.6	0.8	1.3	0.8	1.3	1.2	0.7	0.8
rp S24	**0.5**	0.0	0.3	0.1	-0.1	0.5	-0.2	0.2	0.3	-0.3	-0.2

Table A.22: ΔC_T values for last 22 genes for the allograft Patients

3 h	**3 h**	**3 h**	**6 h**	**6 h**	**6 h**	**9 h**	**9 h**	**9 h**	**12 h**	**12 h**	**12 h**
-0.3	0.1	0.2	0.3	0.4	0.4	0.1	0.4	0.5	0.4	0.9	0.9
-2.1	-2.0	-2.8	-3.0	-2.8	-2.8	-2.7	-2.6	-2.1	-2.6	-2.7	-1.9
-1.1	-0.5	-0.8	-1.0	-0.1	-0.8	-1.1	-0.5	-0.5	-0.8	-1.4	-1.4
0.0	1.8	1.8	0.7	2.5	1.1	1.3	1.1	1.4	1.4	1.2	0.1
0.2	1.1	0.7	0.3	0.8	0.4	-0.2	-0.5	1.0	0.8	-0.4	-0.6
10.5	12.3	10.3	10.3	11.7	10.2	10.1	9.5	11.7	10.2	9.0	8.0
0.8	1.2	1.1	0.7	1.4	0.6	0.4	1.3	0.9	0.8	0.7	0.1
8.5	8.1	8.9	8.6	8.8	9.5	8.0		8.8	8.6	8.3	6.8
9.8	8.4		9.2	10.6	10.3	8.9	9.3	10.1	9.0		8.9
8.0	8.1	9.2	7.7	8.7	8.1	7.3		8.6	7.2	5.9	5.7
5.1	5.2	5.1	4.8	5.2	4.9	4.9	4.8	5.2	4.9	5.0	5.0
7.1	6.6	6.1	6.6	7.1	7.1	5.5	4.7	6.6	6.5	5.8	5.6
1.1	2.0	1.7	0.2	1.2	1.3	0.7	0.4	1.4	0.4	0.3	0.0
8.0	8.8	9.0	7.3	9.2	8.6	8.2	5.6	8.8	6.8	6.0	5.0
4.7	6.1	6.1	4.2	5.9	5.3	5.8	3.8	5.9	4.7	5.4	4.1
9.0	9.1	9.8	8.7	9.5	8.6	8.2	8.3	9.6	8.4	7.9	7.7
9.7	9.3	12.6	10.1	11.7	11.6	10.5	10.8	11.8	9.5	10.0	9.7
-0.1	0.3	0.1	-0.4	0.5	0.6	-0.6	-0.2	-0.3	-0.1	-1.1	-1.3
4.1	5.8	4.9	4.6	5.4	4.2	4.0	4.3	4.8	4.6	5.4	3.1
1.3	1.2	1.5	0.8	1.3	1.5	0.8	0.7	0.7	0.8	0.1	0.3
0.0	0.0	0.3	-0.1	0.3	0.1	-0.1	-0.1	0.0	-0.1	-0.6	-0.3

Table A.23: $\Delta C_t T$ values for last 22 genes for the allograft Patients

18 h	**18 h**	**18 h**	**d 1**	**d 1**	**d 1**	**d 2**	**d 2**	**d 2**	**d 3**	**d 3**	**d 3**
0.3	1.0	1.2	0.9	1.6	0.8	1.8	3.2	2.1	2.5	-0.9	2.3
-2.2	-2.2	-2.1	-1.2	-1.8	-1.2	-0.5	0.1	-0.9	0.1	-3.2	-0.8
-1.5	-1.0	-0.8	-0.8	-1.2	-1.9	-1.4	-0.5	-1.8	-0.8	-0.5	-1.7
0.6	0.3	1.2	0.4	2.4	0.8	-0.2	1.4	0.7	0.2	-2.6	0.0
-0.4	0.3	0.3	-0.2	1.3	-0.9	-0.3	1.2	-0.4	0.8	1.3	
7.3	7.8	9.3	7.0	9.8	8.9	7.3	9.4	10.6	10.1	11.0	9.4
-0.1	0.8	1.0	0.7	-2.8	4.1	-0.1	0.4	0.3	-0.8		-0.7
8.1	8.5	8.0	8.1	7.2	8.1	6.3	6.8	7.3	7.5	8.2	7.7
9.7			11.9	10.5		9.9	8.2		11.2	11.1	
8.5	9.9	8.9	8.1	9.2	9.3	7.7	7.6	8.0	8.7	9.0	
6.2	6.8	6.6	6.2	6.5	6.3	5.7	6.1	7.5	6.4	7.1	6.3
5.5	5.5	5.9	5.9	6.1	5.7	5.6	6.0	7.7	6.1	6.1	5.6
6.0	6.7	7.0	7.8	8.5	7.1	6.3	7.6	7.4	7.3	7.7	6.7
0.5	0.7	1.4	1.9	1.4	0.8	2.0	1.4	0.0	0.6	0.6	-0.3
6.2	6.4	8.2	5.4	6.0	6.6	4.9	5.6	4.6	3.4	4.0	3.6
5.4	5.4	6.2	5.1	4.6	5.2	4.8	5.3	3.8	2.7		2.5
7.9	9.1	7.9	8.6	4.8	12.5	8.5	7.8	7.0	9.1	6.1	8.8
9.0	9.7	8.9	11.6	11.5	10.6	10.6	8.8	9.9	12.0	11.6	12.1
-1.2	-0.6	-0.3	-0.9	-0.8	-0.8	-0.5	-1.2	-1.0	-1.6	-4.3	-1.4
5.0	4.7	5.9	4.5	5.9	5.5	3.7	5.2	5.9	4.2	2.0	5.3
0.3	0.7	0.8	0.7	0.7	0.6	0.7	0.7	1.2	0.4		0.9
-0.5	-0.2	0.1	-0.1	0.0	-0.1	-0.4	-0.1	-0.3	-0.2	0.1	-0.2

Table A.24: ΔC_T values for last 22 genes for the allograft Patients

d 4	**d 4**	**d 4**	**d 5**	**d 5**	**d 5**	**d 6**	**d 6**	**d 6**	**d 7**	**d 7**	**d 7**	**d 7**
3.3	2.7	2.5	3.6	2.4	3.5	3.6	4.0	3.0	2.6	3.8	3.9	2.8
-0.4	-0.4	4.2	0.7	-0.1	-0.1	2.6	3.7	0.9	-0.9	0.9	1.6	2.6
0.3	0.2	2.2	1.8	0.9	-0.8	1.8	2.0	0.6	-1.3	0.8	1.6	0.9
0.9	0.6	0.4	1.2	0.6	0.8	2.5	2.8	0.5	0.6	1.8	1.7	2.3
1.5	0.5	-0.7	0.9	0.8	-0.6	1.4	1.6	-0.5	-1.0	0.3	1.0	-0.5
9.4	9.3	7.3	8.0	7.2	4.9	4.8	5.2	7.1	4.7	5.0	5.4	3.3
-0.7	-0.6	-1.1	-0.5	-0.3	0.7	1.5	1.6	-0.6	-0.1	1.2	0.9	0.0
7.6	8.0	6.2	7.5	8.2	4.2	7.8	8.1	6.9	5.7	6.9	7.6	3.5
8.5	9.3	9.5	8.7	9.8	5.6	8.1	8.3	7.0	6.8	7.8	7.7	5.5
8.8	11.3	9.5	7.8	9.0	7.3	8.7	8.8		9.3	9.2	9.9	
6.8	6.5	5.1	6.7	6.2	4.4		6.6	5.3	4.2	5.6	6.1	
6.3	6.5	5.6	7.0	6.9	5.3	5.8	7.2	6.8	5.1	6.6	6.6	5.2
9.4	9.3	6.9	9.4	8.4	4.9	9.6	9.3	6.8	7.2	9.3	9.3	4.5
1.1	-1.2	-0.8	-0.2	-1.1	-1.5	-0.6	-0.6	-1.2	-2.8	-0.9	-1.0	-1.2
4.6	2.5	3.9	2.5	1.8	0.5	0.7	1.1	0.5	-0.4	0.4	0.4	0.3
2.9	0.5		1.8	0.6	-0.3	0.0	0.7	-0.3	-1.0	-0.3	-0.5	-0.6
6.7	9.7	7.4	6.4	8.8	8.1	7.9	8.1	8.9	7.5	8.9	8.2	7.6
9.3	10.8	10.4	9.2	9.9	10.5	9.5	9.2	10.6	9.9	10.8	9.9	9.5
-0.9	-1.6	-2.4	-0.8	-1.3	-1.7	-1.2	-0.4	-1.5	-2.1	-1.4	-1.2	-1.7
5.3	4.5	4.8	5.6	4.5	8.1	6.6	6.8	6.6	5.0	6.7	5.5	7.2
0.8	0.7	0.1	1.5	0.9	0.8	1.2	1.8	1.0	-0.3	0.7	1.0	1.0
-0.4	-0.2	-0.2	0.5	0.5	-0.2	0.5	0.9	-0.1	-1.1	0.2	0.9	1.2