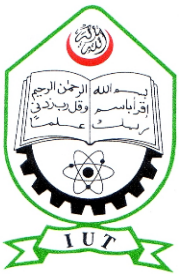

ISLAMIC UNIVERSITY OF TECHNOLOGY

A Subsidiary Organ of the Organization of Islamic Corporation
Dhaka, Bangladesh

Prediction of a gene regulatory network from gene expression Profiles with Linear Regression and Pearson Correlation Coefficient

Authors:

Mehedi Hasan (104407)

Shakhawat Ahmmed Nobin (104423)

Supervisor:

Tareque Mohmud Chowdhury
Assistant Professor
Department of Computer Science and Engineering

A thesis submitted to the Department of Computer Science & Engineering
In partial fulfilment of the requirements for the degree
B. Sc. Eng. in Computer Science &Engineering
Academic Year: 2013-2014

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the Analysis and investigation carried out by Mehedi Hasan and Shakhawat Ahmmed Nobin under the supervision of Tareque Mohmud Chowdhury in the Department of Computer Science and Engineering (CSE), IUT, Dhaka, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

_____

Mehedi Hasan
Student ID – 104407

_____

Shakhawat Ahmmed Nobin
Student ID – 104423

_____

Supervisor:
Tareque Mohmud Chowdhury
Assistant Professor

Department of Computer Science and Engineering
Islamic University of Technology (IUT)

Abstract

Reconstruction of gene regulatory networks is the process of identifying gene dependency from gene expression profile through some computation techniques. In our human body, all cells contain same genetic material but the same genes may or may not be active. This variation in the activation of genes assists researchers to understand more about the function of the cells. Microarray technology helps researchers to get insight about many different diseases such as various cancer disease, heart disease, mental illness, and infectious disease, etc. In this study, a cancer-specific gene regulatory network has been constructed using a simple and novel machine learning approach. First, significant genes differentially expressing them self in the disease condition has been identified using linear regression algorithm. Next, regulatory relationships between the identified genes has been computed using Pearson correlation coefficient. Finally The obtained results has been validated with the available databases and literatures. We can identify the hub genes and can be targeted for the cancer diagnosis.

Contents

Declaration of Authorship ii
Abstract iii
Contents iv
List of Figures vi
List of Tables vi
CHAPTER-1 1
Introduction 1
1.1 Overview 1
1.2 Problem Statement 1
1.3 Research Challenges 2
1.3.1 Dimension 2
1.3.2 Noise 2
1.3.3 Complex Network Relationship 2
1.3.4 Threshold Selection 2
1.4 Thesis Objective 2
1.5 Thesis Contributions 3
1.6 Organization of thesis 3
CHAPTER-2 4
Literature Review 4
2.1 Gene and Gene Expression 4
2.2 Microarray Data and Challenges 5
2.3 Gene Regulatory Network 5
2.4 Related Works 6
2.4.1 RELIEF-F 6
2.4.2 Wrapper and filter approach 8
2.4.3 Spearman's rank correlation coefficient 8
2.4.3 Bayesian network 8
2.4.4 (BOLS) algorithm 9
2.4.5 T test and fold change 9
2.5 Overall GRN construction Process 9
CHAPTER-3 10

Proposed Method 10
3.1 Skeleton of Proposed Method 10
3.2 Microarray Dataset 12
3.2.1 Preprocessing and Normalization 12
3.3 Most Significant gene extractions 12
3.3.1 Proposed Algorithm 12
3.3.2 Removing Redundant Gene 12
3.3.3 Linear Regression on Microarray Dataset 13
3.3.4 Working Principle 15
3.4 Identifying Regulatory Relationship 15
3.4.1 Pearson Correlation on significant genes 15
3.4.2 Working Principle 16
3.5 Construction of Gene Regulatory Network 16
3.6 Identifying Genes Responsible for Cancer 17
CHAPTER-4 18
Performance Analysis 18
4.2.1 Stability checking of Proposed Procedure 19
4.2.2 Validation and Accuracy 21
4.2.3 Comparison Analysis 25
CHAPTER-5 27
Conclusion 27
5.1 Summary of Contribution 27
5.2 Limitation and future work 27
Appendix A 28
Bibliography 38

List of Figures

Figure 1: Example of Gene Regulatory Network (GRN). 3
Figure 2: Gene 4
Figure 3 : Gene Expression 5
Figure 4: Gene Regulatory Network 6
Figure 5 : Steps of Proposed Methodology 10
Figure 6 : flowchart of our proposed method 11
Figure 7 : Linear Regression 14
Figure 8 : Constructing Gene Regulatory Network 16
Figure 9 : Identifying Genes Responsible for Cancer 17
Figure 10 : Comparison analysis Subgroup1 (30 genes) 19
Figure 11 : Comparison analysis Subgroup2 (30 genes) 20
Figure 12 : Final GRN with 100 genes from dataset-2 22

List of Tables

Table 1 : A typical gene expression matrix 5
Table 2 : Datasets used for our method. 18
Table 3 : Similarity between two subgroup 20
Table 4 : Genes Associated with a High Susceptibility of Colorectal Cancer [25-28]. 23
Table 5 : Highly connected genes involve in network construction for dataset 1 24
Table 6 : Highly connected genes involve in network construction dataset 2 24
Table 7 : Comparison among Literature Reviewed Result and obtained Genes 26

CHAPTER-1

Introduction

1.1 Overview

Sequencing the human genome is one of the important accomplishment in the history of System Biology. Cancer of various types have been the leading cause of death for recent years. According to WHO (World Health Organization) 8.2 million people died from cancer in 2012 and 20% of them could be cured if early detection would possible [1]. As no single genes decides how an organism grows therefore an understanding of gene regulatory network is the key that will open the door to those early detection of those diseases.

Healthy and cancer cells often share some common cancer responsible genes. Due to difference in GRNs some cells shows cancer some are not. By the virtue of Microarray technology a large simulation can easily be done with single experiment over thousand data's which helps researchers to detect those markers responsible for showing cancer [2].Research on gene expression profile based cancer detections is accepted by many researcher throughout the world [3]. Dimensional problem (X>>>Y) alone with noise problem disturb easy simulation on microarray data, where R is a matrix and X and Y are column and row, representing the genes and samples (sometimes environmental conditions and sometimes time series). Gene Regulatory Network has been widely investigated in literature. Examples are Boolean Network, Neural Network, Probabilistic Boolean Network , Support Vector Machine, Multi-layer Perceptron, Machine learning approach, Linear and Nonlinear ordinary differential equations are discussed here [5] [6] [7] [8] [9] [10].

In this work we used Machine learning approach (Linear Regression based feature selection) to reduce the dimension of microarray dataset and Modified Pearson Correlation to reconstruct the GRN of [prostate] cancer. Cause of this disease is the change in expression level of genes.

1.2 Problem Statement

It's really a tough job to find a specific gene or group of genes which is responsible for cancer in human body among a huge amount of genes. Some genes expressed in one sample in some amount and same genes expressed with another value to create a cancer. It is hardly believed that if some specific genes network could be identified then medicine could be applied on those

genes to connected mostly with others could reduce the cancer mortality. But a large dataset with thousands of redundant genes changing sample to sample, time to time make the work difficult. Therefore reducing the dataset to indicate the most significant genes and then correlate them to find their relations then representing them in network may give idea about the responsible genes with maximum connections with other genes in that network.

1.3 Research Challenges

1.3.1 Dimension

Total number of genes are very large in compare with the total number of sample [12]. It is very difficult to remove some genes since we could have lost the important genes.

1.3.2 Noise

Observed data contains a significant amount of noise [12]. In general, the changes in the measured transcript values between different experiments are caused by both biological variations and experimental noise. To correctly interpret the gene expression in microarray data, it is crucial to understand the sources of the experimental noise [13].

1.3.3 Complex Network Relationship

Finally the obtained network is much complex and large to represent and select the most significant genes [11].

1.3.4 Threshold Selection

Difficult to select Threshold for Pearson Correlation technique. Since significant genes could be out due to bad selection of threshold. The threshold for which Maximum validate node is selected is chosen as threshold.

1.4 Thesis Objective

We already stated that a large amount of data with noise is present in microarray dataset. We need to remove those genes that are not involve in any type of dependencies. In this study we will not consider those genes which do not show any divergences from normal expression. And by selecting a list of most significant genes we need to correlate them so that on the basis of those we can easily show a network with nodes and edges. Nodes represent the genes and edges representing the dependencies among those genes. Our main goals is to find those genes with large degree which can be treated as most significant genes responsible for early detection and prevention of cancer. Here we will emphasis on the expression level of mRNA and Protein.

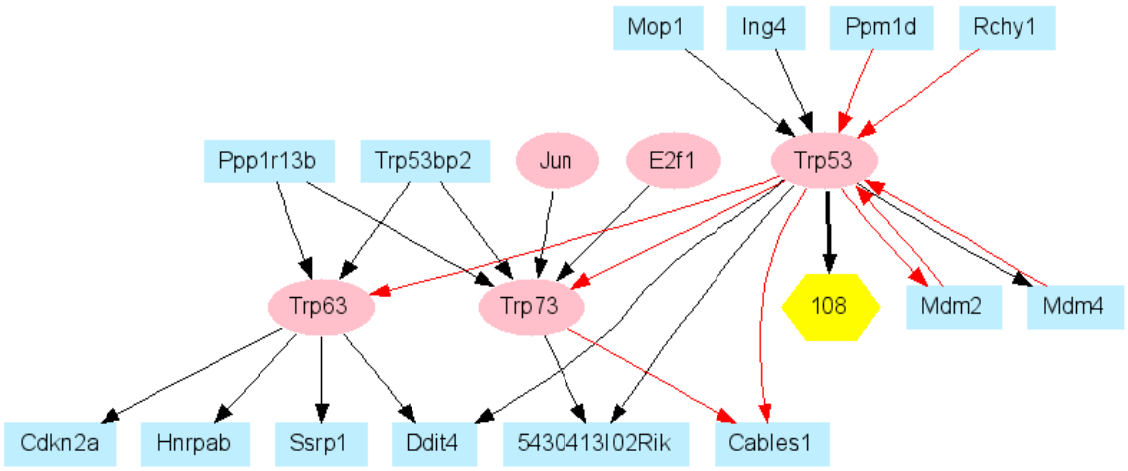

Figure 1: Example of Gene Regulatory Network (GRN).

1.5 Thesis Contributions

The main features of this thesis is to construct a gene regulatory network which will help biologist to identify the responsible cancer genes easily. We have proposed a combined approach which is robust comparing to others. The contribution of this thesis are:

- Handling the high-dimensionality problem by removing redundant genes without losing any significant information. These are done by measuring expression level of genes in different samples and get rid of those unwanted genes.
- We have reduced the complexity of analyzing the large microarray data.
- Our Proposed method Used Pearson Correlation technique alone with linear regression.
- We constructed Network with different parameter with different amount of genes in different levels to show the actual network.
- For experimental result we have implemented our proposed method and figured out the performance using graphical and statistical approaches
- Comparative analysis of our proposed method with different data set is given to show the strength of this combined technique.

1.6 Organization of thesis

The rest of the thesis will be organized as follows: In Chapter 2 we present the literature review of the existing methods and their performance as well as limitations for the detection process. In Chapter 3, we proposed our method for constructing gene regulatory network. There we discuss the overall idea of our proposed method and step by step implementation process. In Chapter 4, Data preprocessing, Experimental set up, Network construction, experimental result and performance analysis of our proposed method with various challenges are shown. Besides with other method a comparative analysis is also shown. Finally in Chapter 5, we conclude our thesis contribution and shown the future scopes for further developing the proposed method.

CHAPTER-2

Literature Review

2.1 Gene and Gene Expression

A gene is the basic physical and functional unit of heredity. Genes, which are made up of DNA, act as instructions to make molecules called proteins. In humans, genes vary in size from a few hundred DNA bases to more than 2 million bases. The Human Genome Project has estimated that humans have between 20,000 and 25,000 genes.

Every person has two copies of each gene, one inherited from each parent. Most genes are the same in all people, but a small number of genes (less than 1 percent of the total) are slightly different between people. Alleles are forms of the same gene with small differences in their sequence of DNA bases. These small differences contribute to each person's unique physical features.

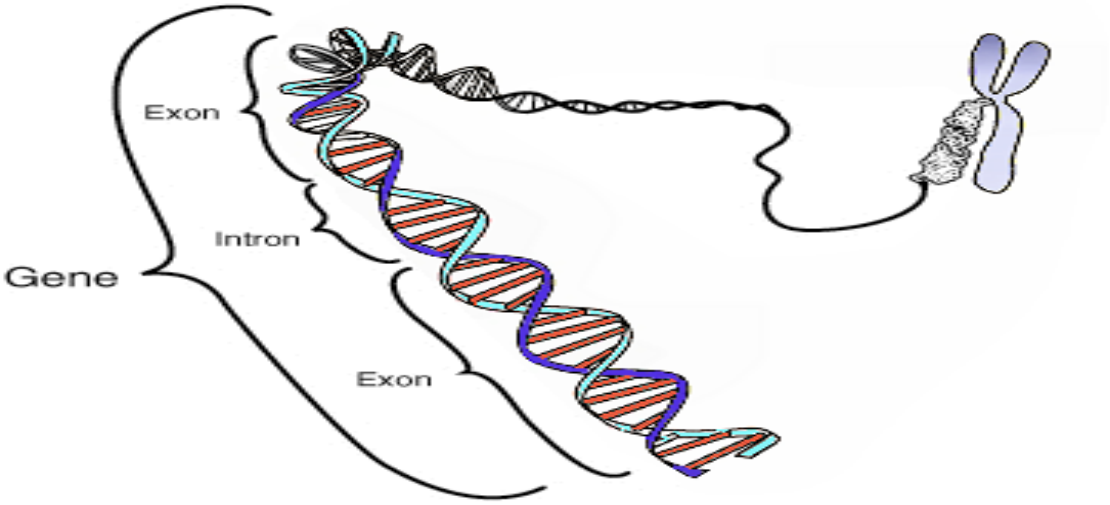

Figure 2: Gene

Information about functional relationship and interactions both between single genes and gene groups can be acquired from gene expression [14] [15].

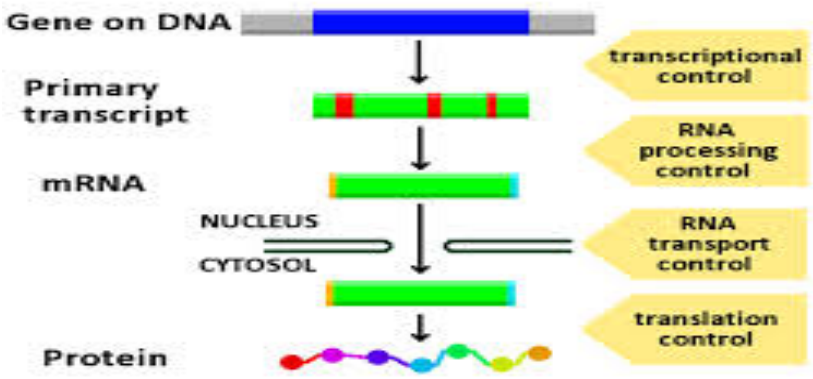

Figure 3 : Gene Expression

Gene expression is the way by which information from a gene is used in the synthesis of a functional gene product.

2.2 Microarray Data and Challenges

Most of the time we have seen to take gene expression value from microarray data. The representation is given bellow. There present sample in one axis and genes in other axis.

	Gene 1	Gene 2	Gene 3	Gene x	Class
Sample 1	F_{11}	F_{12}	F_{13}	F_{1x}	C_1
Sample 2	F_{21}	F_{22}	F_{2x}	F_{2x}	C_2
Sample 3	F_{y1}	F_{y2}	F_{yn}	F_{yx}	C_y

Table 1 : A typical gene expression matrix

We get this data after some steps where expression process sometime modulated, transcriptions including, spicing of RNA, then translation and post translational change of a protein.

2.3 Gene Regulatory Network

A gene regulatory network or genetic regulatory network (GRN) is a collection of DNA segments in a cell which interact with each other indirectly (through their RNA and protein expression products) and with other substances in the cell, thereby governing the expression levels of mRNA and proteins. In general, each mRNA molecule goes on to make a specific protein (or set of proteins). In some cases this protein will be structural, and will accumulate at the cell membrane or within the cell to give it particular structural properties.

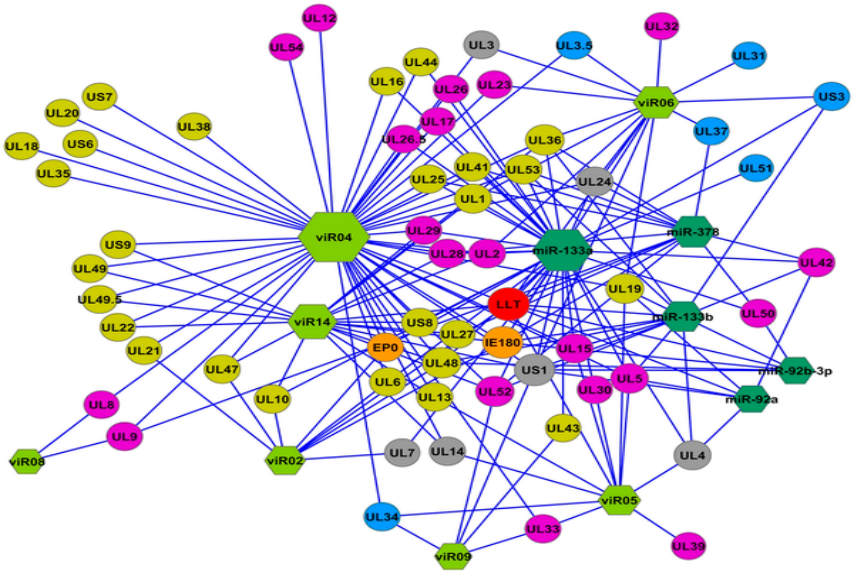

Figure 4: Gene Regulatory Network

In other cases the protein will be an enzyme, i.e., a micro-machine that catalysis a certain reaction, such as the breakdown of a food source or toxin. Some proteins though serve only to activate other genes, and these are the transcription factors that are the main players in regulatory networks or cascades. By binding to the promoter region at the start of other genes they turn them on, initiating the production of another protein, and so on. Some transcription factors are inhibitory.

2.4 Related Works

Although many papers have been published suggesting different GRN constructions techniques, the challenges which are faced have not been overcome yet. Every algorithm have some strong side and weak side or lacking and limitations. We also find some limitations on using one algorithm with other algorithm in some study. For this even today it is still a prominent research topic in bioinformatics fields.

Some procedure discussed below:

2.4.1 RELIEF-F

Relief-F is improved version of original Relief algorithm which has three important improvement and they are: less sensitivity to noise , better strategy for coping missing value and handling multiclass data[1].

Algorithm 1 RELIEF-F (The pseudo code):

Input: M learning instances X_k described by N features; C classes; m iterations; class probability p_w; number of n nearest instances from each class

Output: for each feature F_i weight within $1 \leq W[i] \leq 1$

1: **for** i = 0 to N **do**

2: W[i] = 0

3: **end for**

4: **for** l =1 **to** m **do**

5: randomly pick an instance X_k(with class y_k);

6: **for** y = 1 **to** C **do**

7: find n nearest instances x[j,y] from class y, where j = 1,……,n;

8: **for** i = 1 **to** N **do**

9: **for** j = 1 **to** n **do**

10: if y = y_k **then** {nearest hit}

11: W [i] = W [i] + diff(i; X_k,X[j, y])=(m × n);

12: **else** {nearest misses}

13: W [i] = W [i] + P_y/1 -p_{yk} ×diff(i; x_k, x[j; y])/(m × n);

14: **else if**

15: **end for**

16: **end for**

17: **end for**

18: **end for**

19: **return (W);**

2.4.2 Wrapper and filter approach

Wrapper approach is wrapped within a learning algorithm. The filter approach is based on their information on dependency. Wrapper approach is to predict the accuracy of a given feature subset. Some forward or backward selection algorithm is applied furthermore for better result. This process is repeated until the goal is achieved. But this method is expensive as it has the bigger run time. The filter approach uses properties of data. Complexity is less than the wrapper model. This approach is suitable for large datasets. In bioinformatics most dataset are very large and it is largely used and then any classifier can be used for evaluate the classification accuracy of the test data which is not possible in wrapper approach.

2.4.3 Spearman's rank correlation coefficient

Spearman's rank correlation coefficient is a nonparametric measure of statistical dependence between two variables. It describe the relationship between two variables using a monotonic function. Spearman's coefficient is appropriate for both continuous and discrete variables including ordinal variables. For a sample size n, the n raw scores X_i , Y_i are converted to ranks x_i , y_i and Þ is [2] :

$$Þ = 1 - \frac{.6 \sum d_i^2}{n(n^2 - 1)}$$

Where d_i = x_i - y_i (difference between ranks)

2.4.3 Bayesian network

BNs correspond to another graphical model structure known as a directed acyclic graph (DAG) that is popular in the statistics, the machine learning, and the artificial intelligence societies. BNs are both mathematically rigorous and intuitively understandable. They enable an effective representation and computation of the joint probability distribution (JPD) over a set of random variables [3]. The structure of a DAG is defined by two sets: the set of nodes (vertices) and the set of directed edges. The nodes represent random variables and are drawn as circles labeled by the variable names. The edges represent direct dependence among the variables are drawn by arrows between nodes [22].

2.4.4 (BOLS) algorithm

Development of efficient computational methods to find gene regulatory networks is one of the great challenges. Networks with large numbers of genes will likely require stronger optimization algorithms. Linear systems of ordinary differential equations are useful for modeling simple gene regulatory systems [4]. Reverse engineering algorithm for underdetermined and ill conditioned linear model, Bayesian Orthogonal Least Squares (BOLS) algorithm [22]. A system of a genetic regulation using differential equations is described:

$\frac{de_i(t)}{dt} = \sum_{j=1}^{k} w_{ij}\ e_j(t) + \text{€}(t)$ for i=1,2,……..,K

General overview of created networks by BOLS algorithm indicates that there are a number of hubs, which are key regulators for many genes. When visualizing the data by cell cycle stages, it becomes evident that the BOLS derived networks clearly define the sub networks in each stage.

2.4.5 T test and fold change

Computing a *t*-statistic can be problematic because the variance estimates can be skewed by genes having a very low variance. These genes are associated to a large *t*-statistic and falsely selected as differentially expressed [21]. Another drawback comes from its application on small sample sizes which implies low statistical power. Consequently, the efficacy of a *t*-test along with the importance of variance modeling have been seriously called into question.[9] The fact the test was introduced more than 100 years ago should mean it is limited to some degree. The tests are based on limited theoretical assumptions and do not take into account all we know about these days. They are not specific over one sample, though it has been suggested over large samples their accuracy is approximately correct.

2.5 Overall GRN construction Process

Studying works on GRN we mostly found that by following these steps ultimately Gene regulatory Network is being constructed. Though different approaches is being applied to different works which varying the performance of

1. Data preprocessing
2. Significant Feature selection
3. Correlation among genes
4. Construction of network
5. Detect responsible genes

CHAPTER-3

Proposed Method

3.1 Skeleton of Proposed Method

To find regulatory relationship between gene pairs using gene expression profile, many techniques have been used in the literature. In this work, for removing the redundant genes linear regression and for dependency among genes Pearson's correlation coefficient has been applied [16] [17]. The main steps of the proposed algorithm are outlined as follows.

(1) Preprocessing of the dataset
(2) Identification of most significant genes
(3) Finding regulatory relationship between gene pairs
(4) Representation of network
(5) Identify the genes responsible for cancer.

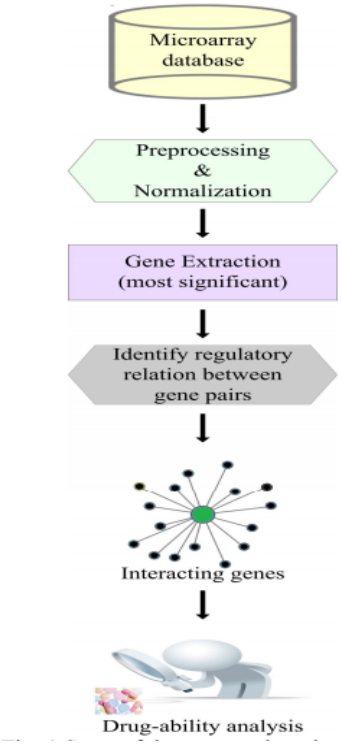

Figure 5 : Steps of Proposed Methodology

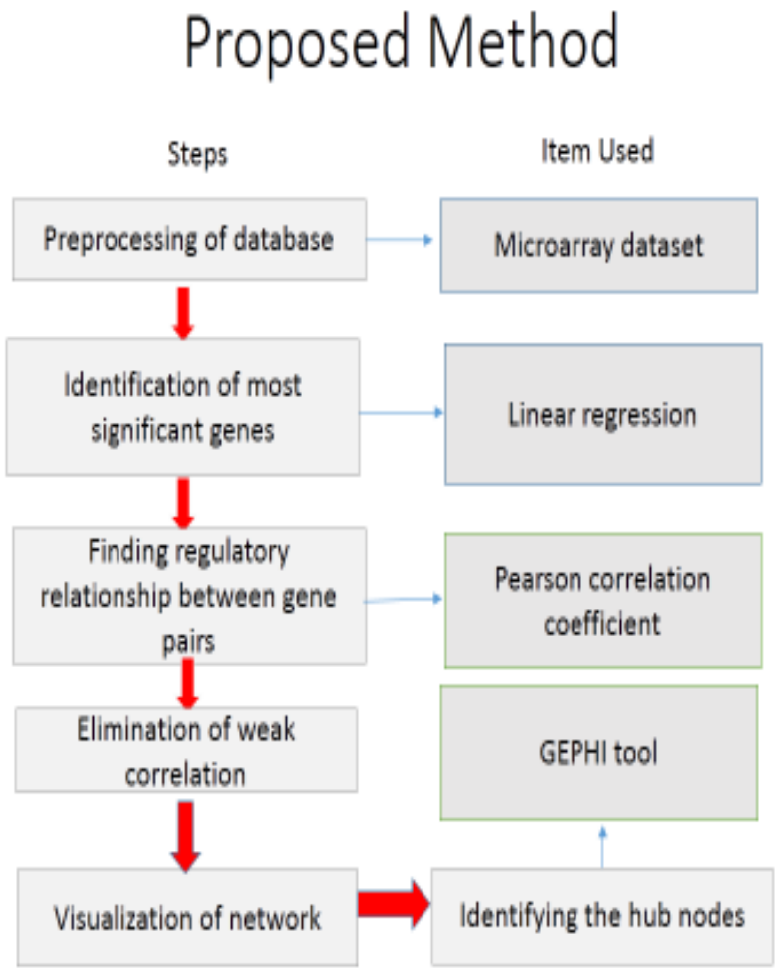

Figure 6 : flowchart of our proposed method

3.2 Microarray Dataset

3.2.1 Preprocessing and Normalization

Preprocessing steps is essential for the data since data are required to prepare them for subsequent inference of a GRN involve not only within samples but also between those microarray dataset samples. After normalization data become easier to handle to obtain gene-centric values of gene expression [18]. We did data preprocessing to handle missing values, duplicate and missing gene names, etc. in the datasets.

3.3 Most Significant gene extractions

In this step, those genes are identified that are differentially expressing themselves in diseased condition. A lots of method are presented in different publications for identifications of differentially expressed gene as for example fold-change , t-test statistics , ANOVA rank product, Random Variance Model, Limma , Linear regression [19] and many more. For efficiency and significant results we will use Linear Regression Analysis. The main steps of the Linear Regression algorithm are outlined as follows:

3.3.1 Proposed Algorithm

Our method of selection of most significant genes is applied on microarray datasets in which two kind of malignant or one normal and another kind of malignant data is present. At the very beginning of this based on the similarity in expression value we removes some genes then applying the linear regression to measure the divergence to select genes from microarray dataset.

3.3.2 Removing Redundant Gene

Microarray dataset contains a large amount of redundant data, huge noise with very closely expressed value. Our main goal is to remove those closely expressed value which ultimately gives no significance in gene selection through mean calculation of different genes expressions value. Our training dataset need to be divide into two subtypes D1 and D2 where we calculate he values of each genes in one subtypes and compare that with another data types for all samples average this procedure need to do for all genes of the previous subtypes. We can easily remove those genes which gives similar values as those will not give any discriminative information about gene expression value.

3.3.3 Linear Regression on Microarray Dataset

To find the genes with discriminative information is our main goal. Linear regression do not give much significant genes if we take a huge dataset with many genes present no discriminative values among them. As we have two set of data so applying regression model on microarray data can be consider as multi variable linear regression approach.

Defining an explanatory and a dependent variable is the key works in this regression analysis. We need to apply multiple target variable since our dataset contains multiple genes which is our target variable in this scenario. we need to compute G© for each of the genes considering as a target variable and all other genes as dependent variable in the base subtype of the dataset to detect the regression model.

$G_© = \beta_0 g_0 + \beta_2 g_2 + \beta_3 g_3 + \ldots\ldots + \beta_n g_n$

$G_© = \beta_0 g_0 + \beta_1 g_1 + \beta_3 g_3 + \ldots\ldots + \beta_n g_n$

$G_© = \beta_0 g_0 + \beta_1 g_1 + \beta_2 g_2 + \ldots\ldots + \beta_n g_n$

…… ……. …… ……..

$G_© = \beta_0 g_0 + \beta_1 g_1 + \beta_2 g_2 + \beta_3 g_3 + \ldots\ldots + \beta_{n-1} g_{n-1}$

Equations stated above is the equation for linear regression model in which $G_©(i)$ denote an explanatory variable and other g_j are the dependent variable without g_i. From gradient descent algorithm we can easily calculate parameter matrix (β) after considering all genes individually.

This β matrix represents the regression model for the subtype of dataset that has been considered for comparison with the other subtype, where each row of β (β_i) is the set of parameters for a particular $G_©(i)$. Using the transpose of this matrix we, statistically can predict the gene expression values by applying $\beta_{(i)}$ on the other subtype of the training dataset and compute the $G_©$'(i). This is done by equation where g_i' is the feature vector of the second subtype of dataset and is β_i^T transpose of parameter vector generated from the first subtype of dataset.

$$G_©\text{'}(i) = \beta_i^T g_i\text{'} \qquad (3.3.1)$$

Our model was designed in such way where we divide our actual dataset into two parts: training dataset and test dataset. From the training dataset we have separated the two different subtypes of data.

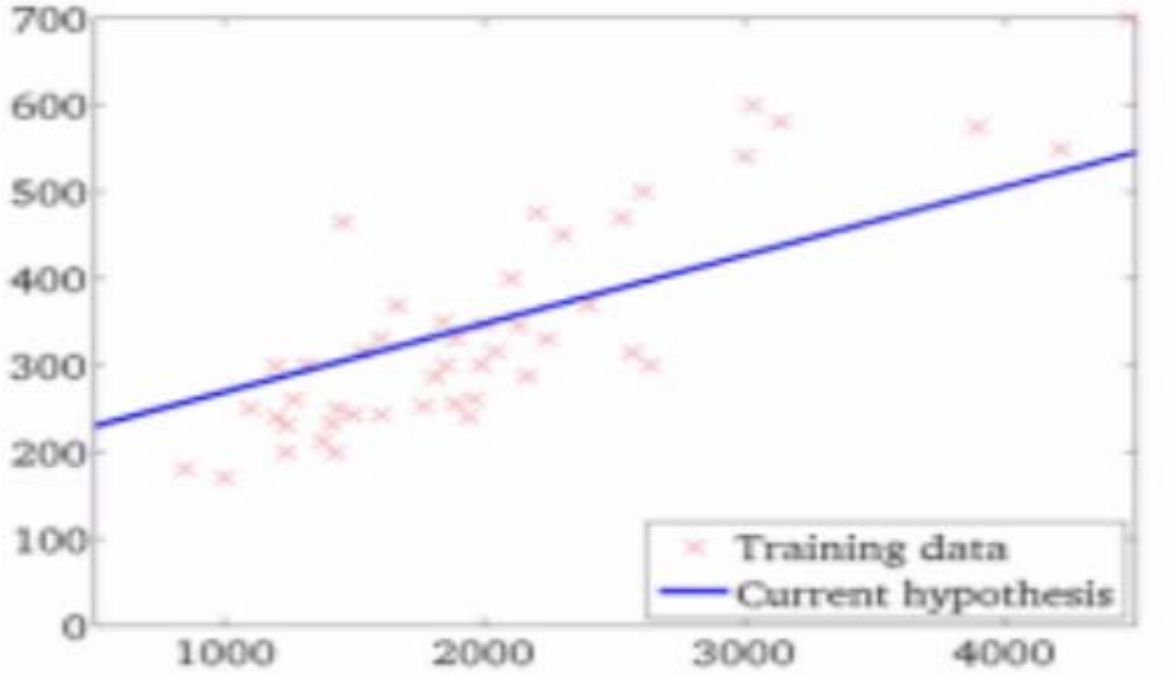

Figure 7 : Linear Regression

ALGORITHM-1

INPUT: D1 and D2 are two subtypes of training dataset. N is the number of features and the number of samples is M.

OUTPUT: A significant set of features.

1. *Mean values for D1 and D2 on every features need to calculate*
2. *Find the difference of mean values between D1 and D2 and sort features on them*
3. *Remove the features with smaller difference (no discriminant expression values)*
4. *Apply linear regression analysis where each features as predictive variable*
5. *Calculate Parameter β for each features*
6. *Create a parameter matrix (β) of size (n*n) for D_1*
7. *Calculate $G_{©}'(i)$ value from Equation 3.3.1*
8. *Calculate the divergence of expression values from standard and sort the features based on their divergence value*
9. *Choose the top-n most significant genes.*

3.3.4 Working Principle

The divergence of genes expression value means those features got some change from ideal value. We have ranked the genes; those are selected based on their divergence from the standard model of regression line. The most deviated gene from the regression line got the highest rank in the feature subset list. The more the expression value is diverged from the regression line, the better possibility the feature is a discriminative feature.

3.4 Identifying Regulatory Relationship

To create a regulatory relationship we must bring a relation among gene pairs. We proposed the Pearson correlation technique to detect how much related the most significant genes are that we get from our previous steps.

3.4.1 Pearson Correlation on significant genes

Correlation means sets of data measuring to detect how well they are related. This is a statistical technique where Pearson correlation is chosen for significant result outcome. It shows linear relationship between two data. We calculate the value 'r' to detect how strongly they are bonded. We follow this equation to calculate r:

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[\, n\Sigma x^2 - (\Sigma x)^2 \,][\, n\Sigma y^2 - (\Sigma y)^2 \,]}}$$

Equation 3.4.1

Where n is the total number of genes, and x and y are two genes between which genes we will calculate the r value.

ALGORITHM-2

INPUT: A set of significant genes

OUTPUT: Weighted Relation between all those genes.

1. *Take one gene and find r value with all other genes according to equation 3.4.1*
2. *Do 1 for all genes to calculate r for all genes*
3. *Calculate the absolute r value which is greater than .85*

3.4.2 Working Principle

The result of the Pearson Correlation is between -1 to 1 though it's rare to get -1, 0 or 1. Three main types of correlation is present those are change like this

HIGH Correlation: 0.5 to 1.0 or -0.5 to 1.0

Medium Correlation: 0.3 to 0.5 or -0.3 to 0.5

Weak Correlation: 0.1 to 0.3 or -0.1 to 0.3

We have selected highly correlated genes only in our work to get largely dependent gene pair.

3.5 Construction of Gene Regulatory Network

Now gene interaction network has been constructed, where nodes correspond to gene names and pairwise r value is allocated to the edge between genes. We use a software name gephi to construct network.

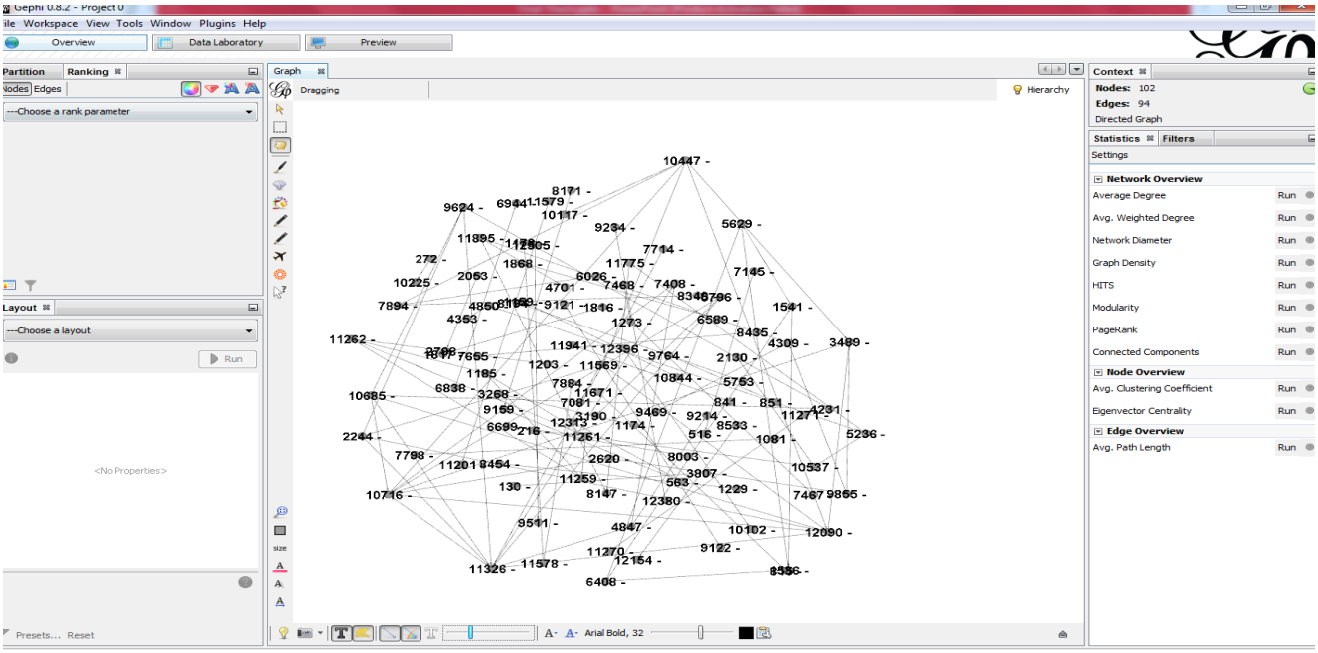

Figure 8 : Constructing Gene Regulatory Network

3.6 Identifying Genes Responsible for Cancer

Genes are treated as Node and relation between them as edge. Therefor the nodes with most degree is related with most genes. That means those are treated as hub node which can be taken into action for early detection or medicated action to take.

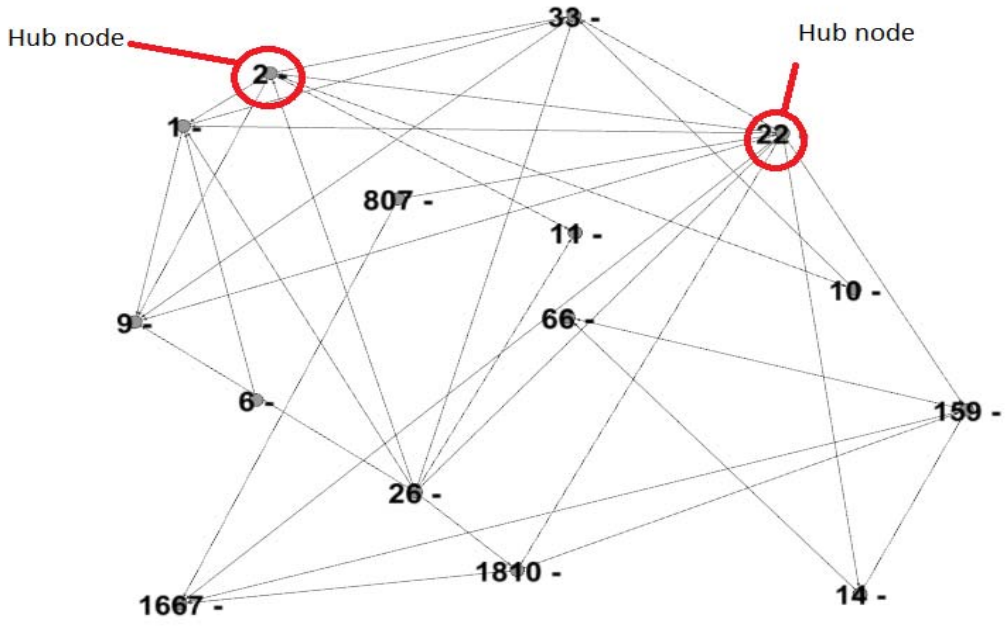

Figure 9 : Identifying Genes Responsible for Cancer

CHAPTER-4

Performance Analysis

4.1 Data Set and Experimental Setup

In our experiment, we have calculated the relationship among genes in 'Matlab 2013' and construction of Gene Regulatory Relationship (GRN) with the 'Gephi' graph design tool. All the simulation are performed on a personal computer of 2.13GHz processors with 2 GB main memory. Here we have used 3 data set. Among them two (one all and one colon) of them used for checking the strength of the proposed procedure and last dataset (colon) used for validation of our result with accuracy.

Data Type	Class	Samples	Genes	Purpose of Use
All Data	2 Class(one cancer positive & one cancer negative)	128	12625	Stability test
Colon Tumor Data 1	2 Class(one cancer positive & one cancer negative)	62	2000	Stability test
Colon Tumor Data 2	2 Class(one cancer positive & one cancer negative)	20	15552	Accuracy and Validation

Table 2 : Datasets used for our method.

4.2 Performance Analysis

In our study we checked the strength of the proposed procedure and then use to find genes responsible. After getting a set of responsible genes we matched those with validated gene set to get result and accuracy.

4.2.1 Stability checking of Proposed Procedure

In our dataset there were total 62 samples. Among them 19 sample was cancer positive and 41 was cancer negative. We divided those into two subgroups and apply our procedure parallels into two sub group to find the result obtained. We have seen that our method was detected almost same genes from subgroup 1 and subgroup 2, which proved stability of our method.

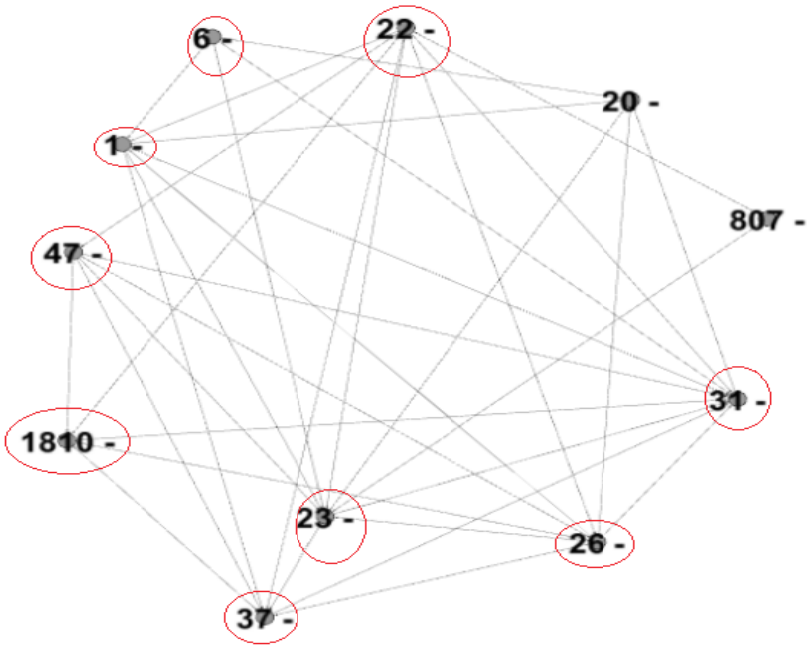

Figure 10 : Comparison analysis Subgroup1 (30 genes)

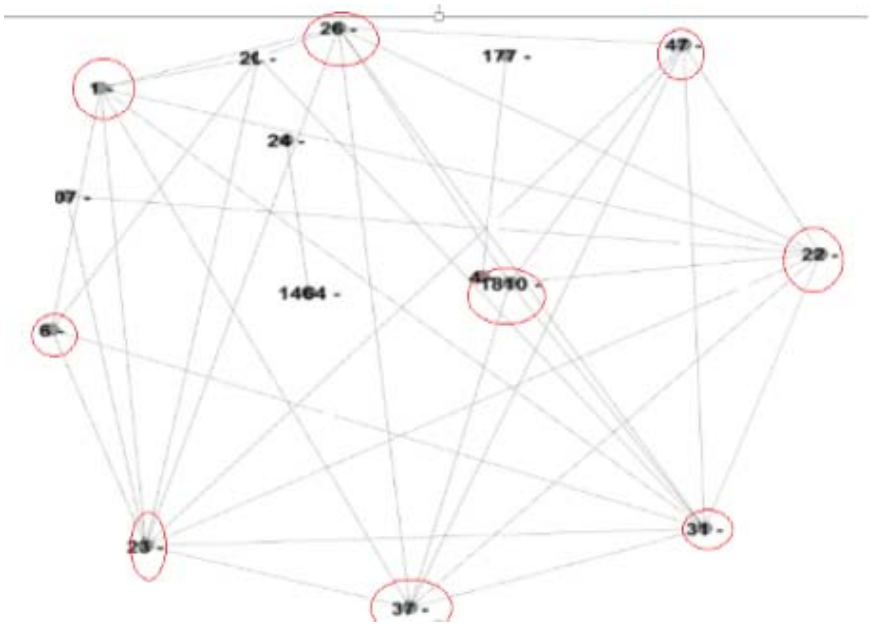

Figure 11 : Comparison analysis Subgroup2 (30 genes)

Gene ID	Group-1	Group-2
1	☑	☑
6	☑	☑
20		☑
22	☑	
24	☑	☑
37	☑	☑
97	☑	
1464	☑	
1810	☑	☑

Table 3 : Similarity between two subgroup

In our study, we divide the data set in two part. Individually we apply our method on both. Then we compare each and check if there is any common set of gene in two parts or not. We find a common set of genes having in both result. That satisfy the stability of our method. In table we will show the comparison.

4.2.2 Validation and Accuracy

From literature review and available dataset have got some genes like APC, MUTYH, TP, EPCAM, BMPR are responsible for cancer in a sample [25]. And 5 of them are present in our dataset we took 100 genes for our procedure and 3 of them were also present in our resultant constructed genes shown in table 4.2.

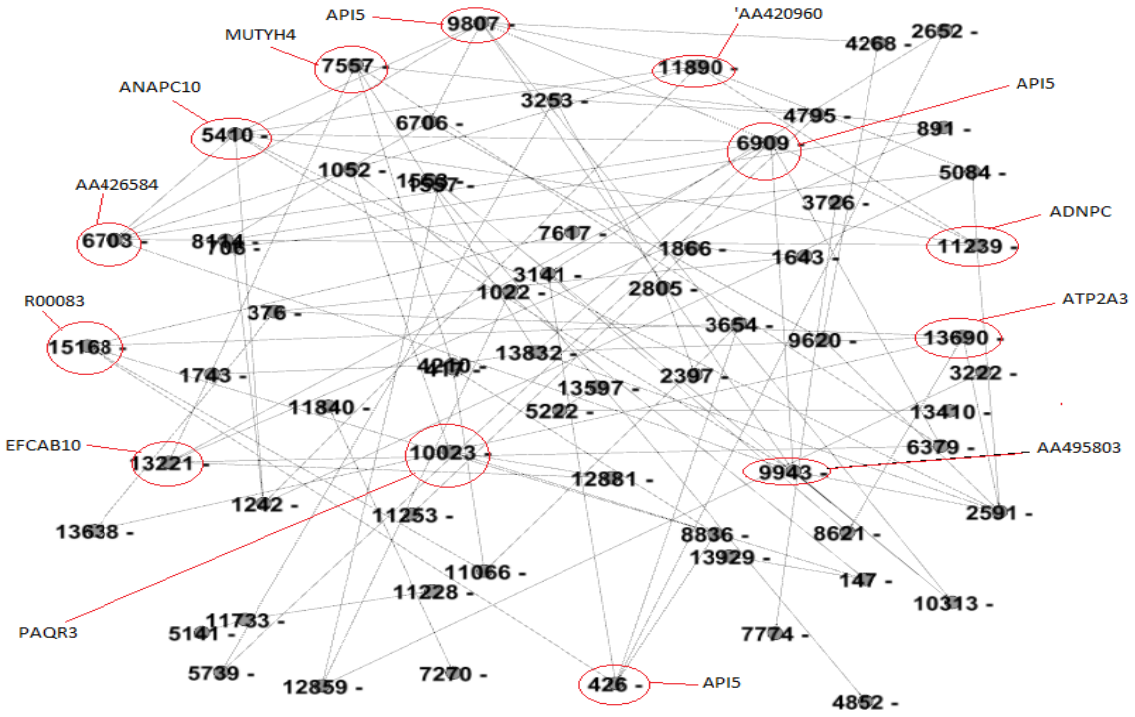

Figure 11: Final GRN with 100 genes from dataset-1

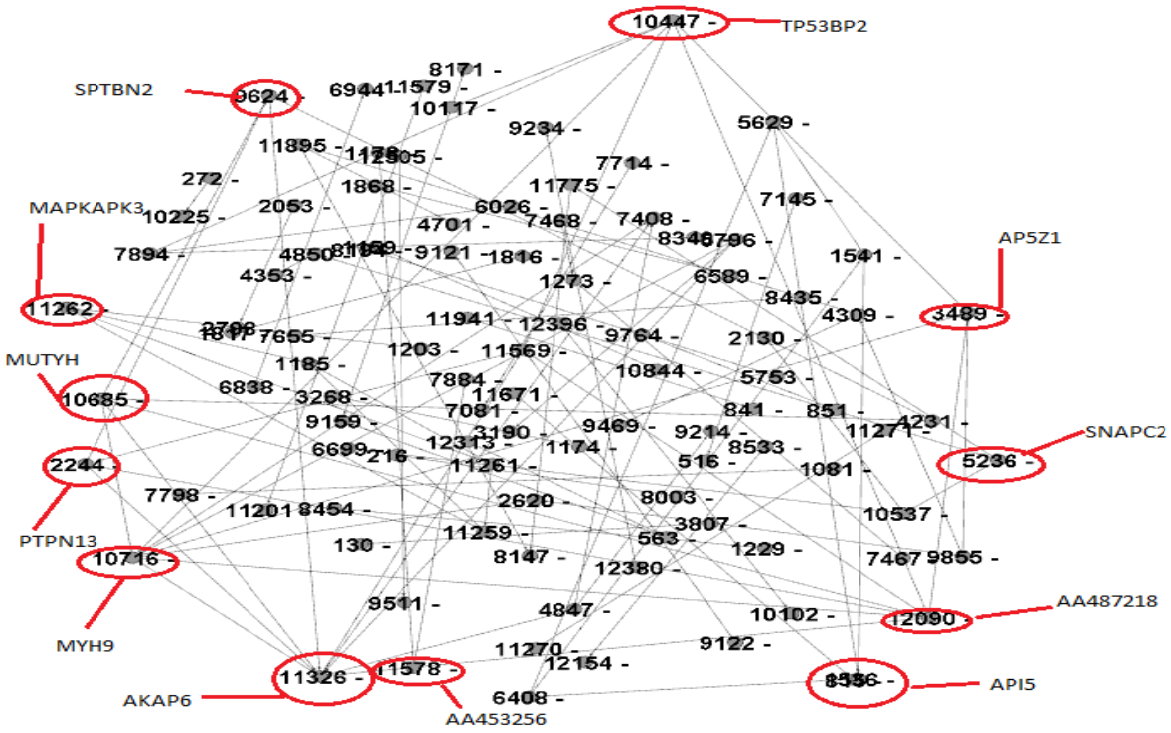

Figure 12 : Final GRN with 100 genes from dataset-2

Gene	Syndrome	Hereditary Pattern	Predominant Cancer
Tumor suppressor genes			
APC (OMIM)	FAP	Dominant	Colon, intestine, etc.
TP53 (*p53*) (OMIM)	Li-Fraumeni	Dominant	Multiple (including colon)
STK11 (*LKB1*) (OMIM)	PJS	Dominant	Multiple (including intestine)
PTEN (OMIM)	Cowden	Dominant	Multiple (including intestine)
BMPR1A (OMIM)	JPS	Dominant	Gastrointestinal
SMAD4 (*MADH/DPC4*) (OMIM)	JPS	Dominant	Gastrointestinal
Repair/stability genes			
MLH1 (OMIM), *MSH2* (OMIM), *MSH6* (OMIM), *PMS2* (OMIM)	LS	Dominant	Multiple (including colon, uterus, and others)
EPCAM (*TACSTD1*) (OMIM)	LS	Dominant	Multiple (including colon, uterus, and others)
MYH (*MUTYH*) (OMIM)	MYH-associated polyposis	Recessive	Colon
POLD1 (OMIM), *POLE* (OMIM)	Oligopolyposis	Dominant	Colon, endometrial
FAP = familial adenomatous polyposis; JPS = juvenile polyposis syndrome; LS = Lynch syndrome; OMIM = Online Mendelian Inheritance in Man database; PJS = Peutz–Jeghers syndrome.			

Table 4 : Genes Associated with a High Susceptibility of Colorectal Cancer [25-28].

From the above table we can see the genes which are mostly responsible and now we will mark those genes which are also available in our study. We applied the method in two dataset. Common genes are marked with red circle alone with their number of degree for which they are treating as responsible genes.

Serial No.	Gene ID	Genes name	Top genes with degree
1.	9807	API5	8
2.	6909	SIN3B	7
3.	13690	ATP2A3	7
4.	10023	PAQR3	7
5.	9943	AA495803	6
6.	6703	AA426584	6
7.	426	API53	6
8.	13221	EFCAB10	5
9.	15168	R0083	5
10.	7557	MUTYH4	5
11.	11890	AA420960	5
12.	13690	ATP2A3	5
13.	11239	ADNPC	5

Table 5 : Highly connected genes involve in network construction for dataset 1

Serial No.	Gene ID	Genes name	Top genes with degree
1.	11326	AkAP6	7
2.	10716	MYH9	7
3.	12090	AA487218	6
4.	10447	TP53BP2	6
5.	106685	MUTYH	6
6.	11578	AA53256	5
7.	9624	SPTBN2	4
8.	13221	PYPN13	4
9.	3489	AP5Z1	4
10.	8356	MUTYH4	4
11.	11890	AA487218	4
12.	11262	MAPKAPK3	4
13.	2244	PTPN	4

Table 6 : Highly connected genes involve in network construction dataset 2

$$\text{Accuracy} = \frac{\text{Detected Genes by our Procedure}}{\text{Actual Genes Detected from literature}}$$

By using this equation we calculate our estimated accuracy where. From literature we APC, MUTYH, TP, EPCAM, BMPR got genes responsible for cancer in our dataset and among them APC, MUTHY, TP Were present .Finally we got 3 Genes which match with the genes from those literature and publications. Thus lastly we got Accuracy 64% on a particular dataset.

4.2.3 Comparison Analysis

A number of approaches has been followed to identify the genes responsible for cancer . Here this study we will compare only those approaches which is similar with our technique. In the table below we have shown the final genes identified by other method and our method . Here we compared the result with the names found in the different literature and website (biological database) . We have shown result of two dataset.

Serial No	Name of the Genes Found in literature and Biological Database	T-test+Fold change with Pearson Correlation	T-test+Fold change with Mutual Information	Linear Regression with Pearson Correlation (Dataset 1)	Linear Regression with Pearson Correlation (Dataset 2)
1	APC (Colon)			☑	☑
2	TP53 (Multiple)		☑	☑	☑
3	MLH1(Multiple)				
4	MUTYH (Colon)			☑	☑
5	POLD1 (Colon)	☑	☑	☑	
6	ACAT2(Multiple)	☑	☑		☑

Table 7 : Comparison among Literature Reviewed Result and obtained Genes

CHAPTER-5

Conclusion

5.1 Summary of Contribution

The complex molecular interaction is for perturbation in the GRN. So, detecting he cancerous genes is a key step for cancer diagnosis. A regulatory network give idea among genes interactions and dependency. In our procedure we took a machine learning approach to find the most significant genes, Pearson correlation between gene-pairs to reconstruction of gene regulatory network. Where we took a sample of 15552 genes and from there after linear regression analysis we got 100 most significant genes after that we apply Pearson correlation model on those 100 Genes to get Pearson factor 'r' value. From those genes we only consider genes which got r>.85 and get 56 correlated genes. After reconstruction of those gens we found 12 as a hub nodes and from literature we have found 3 are similar with our procedural result output.

5.2 Limitation and future work

Due to difficulty in dataset availability, the construction of gene regulatory networks and their validation in a realistic manner is really a difficult task. Our study need more experimental validation to get the maximum utility. Our propose approach can help to identify common molecular interaction in the cancer study not only in colon cancer but other cancer like lung, breast, etc. In future we will try to implement with other dataset for construction of those types of cancer. Since microarray data is noisy in future we will try to apply some more approach to find more relevant genes as well as a strong correlation techniques my apply to find better dependency network.

Appendix A

Matlab Simulation Code of Proposed Method

ReadData.m

```
data=geosoftread('GDS2918.soft');
data.Data;
d = size(data.Data);
fid = fopen('Mymatrix1.txt','wt');

for ii = 1:size(data.Data,1)
   fprintf(fid,'%20.18f \t',data.Data(ii,:));
   fprintf(fid,'\n');
end
fclose(fid);
```

NormalizeData.m

```
data = importdata('Mymatrix1.txt');
data= data';
[ r, c ] = size(data);

datanorm = zeros( r, c);

for col = 1 : c
   maximum = max( data( :, col ));
   minimum = min( data( :, col ));
   for row = 1 : r
      value = data( row, col );
      datanorm( row, col ) = ( value - minimum ) / ( maximum - minimum ) * 10;
   end
end

testdata1 = zeros( 10, c);
testdata2 = zeros( 10, c);
```

```
i = 1;
for j = 1 : 10
   for k = 1: c
   testdata1( i, k ) = datanorm( j, k );
   end
    i = i + 1;
end

i = 1;

for j = 11 : 20
   for k = 1: c
   testdata2( i, k ) = datanorm( j, k );
   end
   i = i + 1;
end

i = 1;

dlmwrite( 'datanorm.txt', datanorm, 'delimiter', '\t', 'precision', 5, 'newline', 'pc' );
dlmwrite( 'testdata1.txt', testdata1, 'delimiter', '\t', 'precision', 5, 'newline', 'pc' );
dlmwrite( 'testdata2.txt', testdata2, 'delimiter', '\t', 'precision', 5, 'newline', 'pc' );
```

A_Removing_Genes_Based_on_Similar_expression_level.m

```
%% Reading the normal input dataset
input1 = importdata('Training Data Type 1.txt');
input2 = importdata('Training Data Type 2.txt');
ffid = fopen('selected.txt','wt');

[rows1,column1] = size(input1);
[rows2,column2] = size(input2);
rows1
column1

for i = 1:column1
  in1 = mean(input1(1:rows1,i));
  in2 = mean(input2(1:rows2,i));

  mean_difference(i,1) = abs(in1 - in2);
  index(i,1) = i;
  %fprintf(fid, '%d\n', mean_difference(i,1));
end
```

```
% Sorting mean_difference in descending order to get the greatest
% differences

for i = 1:column1
   for j = i:column1
     if(mean_difference(i,1)<= mean_difference(j,1))
      temp1 = mean_difference(i,1);
      mean_difference(i,1) = mean_difference(j,1);
      mean_difference(j,1) = temp1;

      temp2 = index(i,1);
      index(i,1) = index(j,1);
      index(j,1) = temp2;
     end
   end
end

% selectin the features for finding the coefficients.
% 10% data will be considered for this step

number_of_selected = (column1*10)/100;
for i = 1:number_of_selected
  fprintf(ffid,'%d\n',index(i,1));
end
fclose(ffid);
```

Constructing coefficient matrix.m

```
%% Reading the normal patients gene expression file
input = importdata('Training Data Type 2.txt');
[rows_input, columns_input] = size(input);

%% Importing the selected Attributes
selected_attribute = importdata('selected.txt');
[rows_attr,columns_attr] = size(selected_attribute);

%% Making the X's (predictors) and Y's (obsevers)

coef = 1;
for iteration = 1:rows_attr
   clear b;
   clear Y;
```

```
  for y = 1:rows_input
    Y(y,1) = input(y,selected_attribute(iteration));
  end

  k = 1;
  clear X;

  % Conatructing X excluding the observer feature column (Y)
  for j = 1:rows_attr
     if(j ~= iteration) % If the column is not observer column
         for i = 1:rows_input
            X(i,k) = input(i,selected_attribute(j));
         end
         k = k + 1;
     end
  end
  b = regress(Y,X); % Multiple linear regression

  % Constructing the matrix of coefficients
  for c = 1:rows_attr - 1
      coefficient(c,coef) = b(c);
  end
  coef = coef + 1;
end

%% Creating File
fid = fopen('coefficient.txt','w');

for cf = 1:rows_attr-1
   for c = 1:coef-1
      fprintf(fid, '%d\t', coefficient(cf,c));
   end
   fprintf(fid,'\n');
end
fclose(fid);
```

Observe values using feature with expression.m

```
%% Reading the coefficient file
coefficients = importdata('coefficient.txt');
[rows_coef, columns_coef] = size(coefficients);

%% Reading the patients' dataset
input = importdata('Training Data Type 1.txt');
[rows_input, columns_input] = size(input);
```

```
%% Reading Selected genes
selected_genes = importdata('selected.txt');
[rows_attr, columns_attr] = size(selected_genes);

for i = 1:rows_attr
  index(i,1) = selected_genes(i,1);
end

% Creating File
fid = fopen('features.txt','w');
%% Calculating the observer (Y) values with the predictor values
for iteration = 1:rows_attr
   % The actual observers' values (actual Ys')
   clear AY;
   clear PY;
   for y = 1:rows_input
     AY(y,1) = input(y,selected_genes(iteration));
   end

   k = 1;
   clear X;
   % Making Xs' for current Y (feature)
   for j = 1:rows_attr
      % Not considering the Y (observer's) column
      if(j ~= iteration)
        for i = 1:rows_input
          X(i,k) = input(i,selected_genes(j));
        end
        k = k + 1;
     end
   end

   % Calculating predicted Ys'
   for p = 1:rows_input
      sum = 0;
     for q = 1:rows_coef
        sum = sum + (X(p,q) * coefficients(q,iteration));
     end
     % Predicted Y values
     PY(p) = sum;
   end

   dsum = 0;
   for diff = 1:rows_input
     divergence(diff) = abs(AY(diff) - PY(diff));
```

```
      dsum = dsum + divergence(diff);
    end
    %fprintf(fid, '%d\n', dsum);
    div(iteration,1) = dsum;
end

% sorting the divergenece in descending order as greatese divergence comes
% first

for i = 1:rows_attr
  for j = i:rows_attr
    if(div(i,1) <= div(j,1))
      temp1 = div(i,1);
      div(i,1) = div(j,1);
      div(j,1) = temp1;

      temp2 = index(i,1);
      index(i,1) = index(j,1);
      index(j,1) = temp2;
    end
  end
end

number_of_features = (rows_attr*50)/100;
for i = 1:number_of_features
  fprintf(fid,'%d\n',index(i,1));
end

fclose(fid);
```

Creating_different_number_of_feature.m

```
%% Creating feature set of different sizes

% Reading the file contianing features index
feature_index = importdata('features.txt');
[rows_f, col_f] = size(feature_index);

% cerating file
fid10 = fopen('10 features3.txt','w');
fid20 = fopen('20 features3.txt','w');
fid30 = fopen('30 features3.txt','w');
fid40 = fopen('40 features3.txt','w');
fid50 = fopen('50 features3.txt','w');
fid60 = fopen('60 features3.txt','w');
```

```
fid70 = fopen('70 features3.txt','w');
fid80 = fopen('80 features3.txt','w');
fid90 = fopen('90 features3.txt','w');
fid100 = fopen('100 features3.txt','w');
fid150 = fopen('150 features3.txt','w');
fid200 = fopen('200 features3.txt','w');
fid300 = fopen('300 features3.txt','w');
fid400 = fopen('400 features3.txt','w');
fid500 = fopen('500 features3.txt','w');


% 10 features file
for j = 1:10
   fprintf(fid10,'%d\n',feature_index(j,1));
end
fclose(fid10);

% 20 features file
for j = 1:20
   fprintf(fid20,'%d\n',feature_index(j,1));
end
fclose(fid20);

% 30 features file
for j = 1:30
   fprintf(fid30,'%d\n',feature_index(j,1));
end
fclose(fid30);

% 40 features file
for j = 1:40
   fprintf(fid40,'%d\n',feature_index(j,1));
end
fclose(fid40);

% 50 features file
for j = 1:50
   fprintf(fid50,'%d\n',feature_index(j,1));
end
fclose(fid50);

% 60 features file
for j = 1:60
   fprintf(fid60,'%d\n',feature_index(j,1));
end
fclose(fid60);
```

```
% 70 features file
for j = 1:70
   fprintf(fid70,'%d\n',feature_index(j,1));
end
fclose(fid70);

% 80 features file
for j = 1:80
   fprintf(fid80,'%d\n',feature_index(j,1));
end
fclose(fid80);

% 90 features file
for j = 1:90
   fprintf(fid90,'%d\n',feature_index(j,1));
end
fclose(fid90);

% 100 features file
for j = 1:100
   fprintf(fid100,'%d\n',feature_index(j,1));
end
fclose(fid100);

% 150 features file
for j = 1:150
  fprintf(fid150,'%d\n',feature_index(j,1));
end
fclose(fid150);

%200 features file
for j = 1:200
  fprintf(fid200,'%d\n',feature_index(j,1));
end
fclose(fid200);


%300 features file
for j = 1:300
  fprintf(fid300,'%d\n',feature_index(j,1));
end
fclose(fid300);


%400 features file
```

```
for j = 1:400
   fprintf(fid400,'%d\n',feature_index(j,1));
end
fclose(fid400);

%500 features file
for j = 1:500
   fprintf(fid500,'%d\n',feature_index(j,1));
end
fclose(fid500);
```

Pearson.m

```
X = load('500 features3.txt');
T = load('Training Data Type 1.txt');


[row col] = size(X);
 n = size(T);
 n = n(1);


 fid = fopen('out3.txt','wt');

for i=1:row
   for j=i+1:row
      sum_x = sum(T(:,X(i)));
      sum_y = sum(T(:,X(j)));

      mul_xy = T(:,X(i)).* T(:,X(j));
      sum_mul_xy = sum(mul_xy);

      sq_x = T(:,X(i)).*T(:,X(i));
      sq_y = T(:,X(j)).*T(:,X(j));

      sum_sq_x = sum(sq_x);
      sum_sq_y = sum(sq_y);


      r = (    n*(sum_mul_xy) - (sum_x*sum_y)              )/sqrt((n*sum_sq_x -
sum_x*sum_x)*(n*sum_sq_y - sum_y*sum_y));

      fprintf(fid,'%5d %5d %20.18f',X(i), X(j), r);
      fprintf(fid,'\n');
```

```
  end
end
```

threshold.m

```
thres = load('out3.txt');

fid = fopen('thres3.txt','wt');

[row col] = size(thres);


thresValue = 0.85;

for i=1:row
   if( abs(thres(i,3)) >= thresValue)
      fprintf(fid,'%5d %5d %20.18f',thres(i,1), thres(i,2), thres(i,3));
      fprintf(fid,'\n');
   end

end
```

output.m

```
thres = load('thres3.txt');

fid = fopen('final3.txt','wt');

[row col] = size(thres);

for i=1:row
   fprintf(fid,'%5d \t %5d',thres(i,1), thres(i,2));
      fprintf(fid,'\n');


end
```

Bibliography

1. URL for WHO(World Health Organization) http://www.who.int/cancer/en/
2. X. Wang and O. Gotoh, "Microarray-based cancer prediction using soft computing approach.," Cancer informatics, vol. 7, pp. 123–39, Jan. 2009
3. R.Xu and D. Wunsch II, "Survey of Clustering Algorithm," *IEEE Transactions on Neural Networks,* vol 16, no.3,pp645-678,2005
4. Shmulevich,I, Dougherty,E., R., Kim, S., K., and Zhang,W., 2002, Probabilistic Boolean networks: a rulebased uncertainty model for gene regulatory networks, Bioinformatics Vol. 18, Pages 261-274, Oxford University Press.
5. R. D. Smet and K. Marchal, "Advantages and limitations of current network inference methods," Nat Rev Microbiol, vol. 8, pp. 717–729, 2010.
6. Teschendorff, A.E., M. Journee, P. A. Absil, et al, Elucidating the altered transcriptional programs in breast cancer using independent component analysis, PLoS Comput Biol., 2007, 3(8), e161.
7. J. Khan, J. Wei, M. Ringner, L. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. Antonescu, C. Peterson, and P. Meltzer, "*Classification and diagnostic prediction of cancer using gene expression profiling and artificial neural networks*" *Nature Medicine ,* vol. 7, pp. 973-979,2001.
8. A. M. Lesk, Introduction to Bioinformatics, 2nd ed. New York, NY,USA: Oxford Univ. Press, 2005, ch. 6 .
9. S. R. Maetschke, P. B. Madhamshettiwar, M. J. Davis, and M. A. Ragan, "Supervised, semi-supervised and unsupervised inference of gene regulatory networks," arXiv:1301.1083v1, 2013.
10. Martoglio, A.-M., J. W. Miskin et al, A decomposition model to track gene expression signatures: preview on observer-independent classification of ovarian cancer. Bioinformatics 2002,18(12):1617-1624
11. H. W. Ressom, et al., (2006). "Inference of gene regulatory networks from time course gene expression data using neural networks and swarm intelligence", In Proceeding of IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology, pp. 1-8.
12. J. Quackenbush, "Computational analysis of microarray data." Nat Rev Genet, vol. 2, pp. 418–417, 2001
13. *Lee* M.-L. T.*, Kuo, F. C., Whitemore, G. A. & Sklar, I. (2000) Proc. Natl. Acad. Sci. USA*97*,9834.*

14. J. Quackenbush, “Computational analysis of microarray data.” Nat Rev Genet, vol. 2, pp. 418–417, 2001.
15. Z. .Wang, M. Gerstein, and M. Snyder, “Rna-seq: a revolutionary tool for transcriptomics,” Nat Rev Genet, vol. 10, pp. 57–63, 2009.
16. X. H. W. Pan, “Linear regression and two class classification with gene expression data,” Bioinformatics, vol. 19(16), pp. 2072–2078, 2003.
17. K. Raza & R. Parveen, (2012). "Soft computing approach for modeling genetic regulatory networks", *Advances in Computing and Information Technology*, vol. 178, pp. 1-12.
18. R. de Matos Simoes and F. Emmert-Streib, "Influence of Statistical Estimators of Mutual Information and Data Heterogeneity on the Inference of Gene Regulatory Networks," PLoS ONE, vol. 6, issue 12, pp. e29279, 2011
19. A. J. Scott, “Illusions in regression analysis,” International Journal of Forecasting (forthcoming), 2012
20. K. Kira and L. A. Rendell, “The feature selection problem: Traditional methods and a new algorithm,” in Proceedings of the AAAI-92, AAAI press,1992, pp. 129–134.
21. Myers, Jerome L.; Well, Arnold D. (2003). *Research Design and Statistical Analysis* (2nd ss
22. Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems, Morgan Kaufmann, San Francisco.
23. D’Haeseleer,P., Liang,S. & Somogyi,R., “Genetic Network Inference: from Co-Expression Clustering to Reverse Engineering”, Bioinformatics, vol. 16, pp. 707-726, 2000.
24. T. Martin, et al., (2010). "Comparative study of three commonly used continuous deterministic methods for modeling gene regulation networks", BMC Bioinformatics, vol. 11, pp. 459.
25. Laken SJ, Petersen GM, Gruber SB, et al.: Familial colorectal cancer in Ashkenazim due to a hypermutable tract in APC. Nat Genet 17 (1): 79-83, 1997
26. Daley D, Lewis S, Platzer P, et al.: Identification of susceptibility genes for cancer in a genome-wide scan: results from the colon neoplasia sibling study. Am J Hum Genet 82 (3): 723-36, 2008.
27. Win AK, Jenkins MA, Buchanan DD, et al.: Determining the frequency of de novo germline mutations in DNA mismatch repair genes. J Med Genet 48 (8): 530-4, 2011.
28. URL for Colon Cancer http://www.cancer.gov/cancertopics/pdq/genetics/colorectal/HealthProfessional/page2#Section_2579