Article

Classification and prediction of dengue fever from microarray samples by LDA based on PPI network

Nahida Habib[1], **Kawsar Ahmed**[2,3], **Md. Binyamin**[4], **M. Mesbahuddin Sarker**[5], **K. M. Akkas Ali**[5]

[1]Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902, Bangladesh

[2]Group of Bio-photomatiχ, Santosh, Tangail-1902, Bangladesh

[3]Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902, Bangladesh

[4]Department of Statistics, Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902, Bangladesh

[5]Institute of Information Technology (IIT), Jahangirnagar University, Dhaka, Bangladesh

E-mail: kawsar.ict@mbstu.ac.bd, k.ahmed.bd@ieee.org, nahidahabib164@yahoo.com

Received 6 December 2017; *Accepted 10 January 2018*; *Published 1 June 2018*

Abstract

Modern Bioinformatics tools have a tremendous contribution in gene analysis, Protein-Protein Interaction (PPI) Network creation and Drug design. It’s been a big challenge to pick out a small subset of informative data from a large microarray dataset and reach on an accurate classification. A successful and precise classification of any disease into its subtype is necessary for successful diagnosis and treatment of the disease. The NCBI Gene Expression Omnibus (GEO) is the extensive storage containing experimental microarray data. In this research, PPI networks and a common drug is designed for the unique DENGUE samples and Linear Discriminant Analysis (LDA) and Principle Component Analysis (PCA) techniques are applied for the classification of Dengue fever genes into its unique samples. Comparing to PCA, in LDA, LD1 classifies 96.2% while PC1 Classifies 46%. Using LDA, also a prediction is made to predict samples from gene variance. Moreover, LDA predicts approximately 73.21% accurate results. All of the calculation, comparison and gene analysis is performed using R tool and UniHi tool is used for the creation of PPI network and Drug design. Here, a common drug is designed which can be used for all of the sample type of the Dengue fever but in different proportion.

Keywords protein-protein interaction; drug design; gene expression omnibus; microarray data; Linear Discriminant Analysis; Principle Component Analysis.

Network Biology
ISSN 2220-8879
URL: http://www.iaees.org/publications/journals/nb/online-version.asp
RSS: http://www.iaees.org/publications/journals/nb/rss.xml
E-mail: networkbiology@iaees.org
Editor-in-Chief: WenJun Zhang
Publisher: International Academy of Ecology and Environmental Sciences

1 Introduction

Bioinformatics can be defined as a combination of biology and computer Science using mathematical and statistical methods or it is a science of analyzing and processing biological data using computer science techniques. It is an interdisciplinary field that develops methods and software tools for understanding biological data (Bioinformatics). Bioinformatics is used in analyzing genomes, proteomes (protein sequences), three-dimensional modeling of biomolecules and biologic systems, etc (Medicinenet). Bioinformatics employs a wide range of computational techniques including sequence and structural alignment, database design and data mining, macromolecular geometry, phylogenetic tree construction, prediction of protein structure and function, gene finding, and expression data clustering (Luscombe, 2001).

The R programming language is an open source scripting language for predictive, analytics and data visualization (TechTarget). R provides a wide variety of statistical (linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering) and graphical techniques, and is highly extensible (R[Home]). It has come with excessive number of libraries and packages (almost 5000). R is one of the most popular languages used by statisticians, data analysts, researchers and marketers to retrieve, clean, analyze, visualize and present data (PROGRAMIZ). It has been seen that, R language is considered responsible for most powerful analytics, statistics, and visualizations tasks.

The Gene Expression Omnibus (GEO) is an international public repository that archives and freely distributes microarray, next-generation sequencing and other forms of high-throughput functional genomic data sets (Barrett, 2013). The database has a flexible design that can handle diverse styles of both unprocessed and processed data in a MIAME- (Minimum Information About a Microarray Experiment) supportive infrastructure that promotes fully annotated submissions (Barrett, 2006). GEO contains Platforms, Series, Samples, Datasets, and Profiles to represent data. R-based analysis of GEO data can be performed using GEO2R.

A microarray database is a repository containing microarray gene expression data and the key uses of a microarray database are to store the measurement data, manage a searchable index, and make the data available to other applications for analysis and interpretation (either directly, or via user downloads) (Microarray [Wikipedia]). Microarrays are one of the latest breakthroughs in experimental molecular biology that allow monitoring the expression levels of tens of thousands of genes simultaneously (Saravanakumar and Natarajan, 2011). Identification of differential gene expression is the first task of an in depth microarray analysis (Mutch, et al., 2001).

Linear Discriminant Analysis (LDA) is a classification method which is used to find the linear combination of feature that separates two or more classes of objects or examples. LDA is used in statistics, mathematics, bio-informatics, pattern recognition and machine learning. LDA is also closely related to principal component analysis (PCA) (LDA [Wikipedia]). Linear Discriminant Analysis (LDA) is most commonly used as dimensionality reduction technique in the pre-processing step for pattern-classification and machine learning applications (Sebastian Raschka). Unlike Logistic regression, LDA is not limited by two-class classification problems instead it can be appropriately used for both two class and more than two class classification problems.

Principal component analysis (PCA) involves a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components (Principal Component Analysis). In simple words, principal component analysis is a method of extracting important variables (in form of components) from a large set of variables available in a data set (Analyticsvidhya). PCA is mostly used as a tool in exploratory data analysis and for making predictive models (PCA [Wikipedia]).

This research analyzes DENGUE microarray GEO datasets from NCBI. Based on this analysis on 56 affected patient's genes a PPI network is designed using UniHi tool which shows the directly connected genes. These directly connected genes are then used to classify DENGUE fever into four unique classes from 56 samples using LDA and PCA. A prediction showing the probability of occurring each unique class is also shown using LDA in R. And then a common drug is designed which can be used for all of the DNGUE fever classes but in different proportion. The paper is organized in 5 sections namely: Introduction, Backgrounds, Proposed Methodology, Result and Discussion and Conclusion. Section 2 discusses about the backgrounds or previous works related to the research, section 3 describes the proposed methodology and working principle, section 4 discusses and analyzes the result and last but not least section 5 includes conclusion and future work.

2 Background

Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) techniques are mostly used for data classification, feature extraction and dimensionality reduction. They are closely associated with each other. LDA ensures higher separability of classes by maximizing the between-class variance and minimizing the within-class variance. Thus LDA increases the ratio of between-class variance to the within-class variance of the given data. LDA can be also used for microarray data classification. According to (Jieping, 2004) Classical LDA projects the data onto a lower-dimensional vector space such that the ratio of the between-class distance to the within-class distance is maximized, thus achieving maximum discrimination. The prime difference between LDA and PCA is that PCA does more of feature classification and LDA does data classification (Balakrishnama and Ganapathiraju, 1995).

Principal component analysis (PCA) is often a first step in the analysis process; however, the standard approach is not tolerant to missing data, because it is based on eigen value decomposition of the covariance matrix (Wolfram S). PCA (Jackson, 1991) has been applied to microarray data in several publications as a data exploration tool (Holter et al., 2000; Raychaudhuri et al., 2000). In PCA, the shape and location of the original data sets changes when transformed to a different space whereas LDA doesn't change the location but only tries to provide more class separability and draw a decision region between the given classes (Balakrishnama and Ganapathiraju, 1995).

The Gene Expression Omnibus (GEO) repository at the National Center for Biotechnology Information (NCBI) archives and freely distributes high-throughput molecular abundance data, predominantly gene expression data generated by DNA microarray technology (Tanya Barrett, 2006). The primary role of GEO is data archiving, functioning as a hub for data deposit and retrieval (Edgar et al., 2002; Barrett et al., 2005). Although GEO represents a huge reservoir of gene expression data that is widely used by the scientific community, it was recognized that the full potential of the repository could only be achieved by making these data easy to search and analyze, even by individuals having little experience in the field, without the need of massive data downloads (Barrett et al., 2005).

Information about the Bio-informatics tool that can be used to show interaction, finding common genes and represent the PPI network among genes as well as proteins is discussed in the article (Klingstrom and Plwczynski, 2010). A common disease regulatory network for metabolic disorders is designed in the paper (Jesmin et al., 2016) by investigating on associated diseases. The UNIHI tool is used to predict PPI network, Common metabolic pathway and Drug design. Zhang and Feng (2017) used network analysis to analyze metabolic pathway of non-alcoholic fatty liver disease. A common pathway shared (Habib, 2016), by bipolar disorder, schizophrenia, coronary heart diseases and stroke can be designed to show the association and interaction among the proteins of these diseases which leads to the way to drug design. In addition, a common drug is designed for bipolar disorder and associated diseases in the article (Habib et al., 2017).

In this research, for the first time we have tried to analyze the GEO microarray DENGUE fevers data. Then PPI networks and Regulatory Interaction Networks are designed. After that, using both LDA and PCA techniques DENGUE fever genes are classified into unique fever sample types (DF, DHF, CP, HC) based on their variance value. A comparison between LDA and PCA results are shown to show the better classification model followed by prediction of samples of the test data based on the training data using LDA. A common drug is also designed for all of the unique DENGUE fever samples which may be used to remedy all the unique fever samples by absorbing in different proportions. All of these operations are performed using R programming language and UniHi tool is used for networks and drug design.

3 Proposed Methodology

Some necessary steps are accomplished to reach the desire goal. All the steps are described below through subsections 3.1 to 3.10 respectively. Fig. 3.1 shows the step by step graphical representation of the research methodology.

3.1 Data source

For the purpose of this research GEO data associated with DENGUE fever are collected from NCBI GEO dataset using R. GEO database stores curated gene expression DataSets, as well as original Series and Platform records in the Gene Expression Omnibus (GEO) repository (Ncbi) and The NCBI (National Center for Biotechnology Information) is freely accessible and downloadable on-line gene database.

3.2 Data collection and preprocessing

At first install and load the required R packages to download GEO data using R. Then using R query download and save GEO Reference Series GSE51808 data. From multiple platforms select Platform GPL13158 and create expression matrix. There are total 56 samples for DENGUE in GEO which includes 4 unique samples or classes. ‘Dengue Fever’ class is characterized as acute dengue fever (DF), ‘Dengue Haemorrhagic Fever’ class is characterized as life-threatening fever (DHF), the Convalescent class (CP) is the 3rd phase of DHF and ‘Healthy Control’ class is the sample meaning not infected by DENV. Collect gene_id and gene_symbol from GEO Reference Series and Platform. Match the gene_symbol with expression matrix and collect them. In the expression matrix the row names are the gene_ids, column names are the sample_ids and matrix values are the gene appearance value. Replace the row names of expression matrix which are the gene_ids by their corresponding matched gene symbol.

3.3 Use sample names as column names

Find Sample_ids with their corresponding sample names in the NCBI GEO Reference Series. In the expression matrix replace column’s name that is sample_ids by their corresponding sample names using R.

3.4 Variance and standard deviation

Write an R function to save the GEO Reference Series GSE51808 data into a text file. Read the text file into a variable g in R using read.table() query. From g calculate variance and standard deviation for all of the genes of all the samples. Sort the genes into decreasing order according to their variance value. Take top 100 variance genes and give a rank to them as highest variance gene is ranked 1, 2nd highest variance gene is ranked 2 etc. and organize them in a table called gene_rank. These result 89 unique genes. The highest variance genes will be needed, so lowest variance genes can be discarded. Edit the table and take top 50 ranked genes from the unique 89 genes.

3.5 Find interacted genes and create PPI

To design a common drug for the four unique samples, interacted or cross linkage genes need to be identified. A protein-protein interaction network is used to show the protein interaction, co-adjustment and co-regulations among genes. For the visualization of PPI network, Unified Human Interactome or UniHI tool is used here. From the PPI networks 3 directly interacted genes are found whose can be used for drug design and also in sample classification.

3.6 Data processing before classification

In this step, the table created in step 3.4 is again modified and a dataframe is created containing the resulting three genes and there variance value.

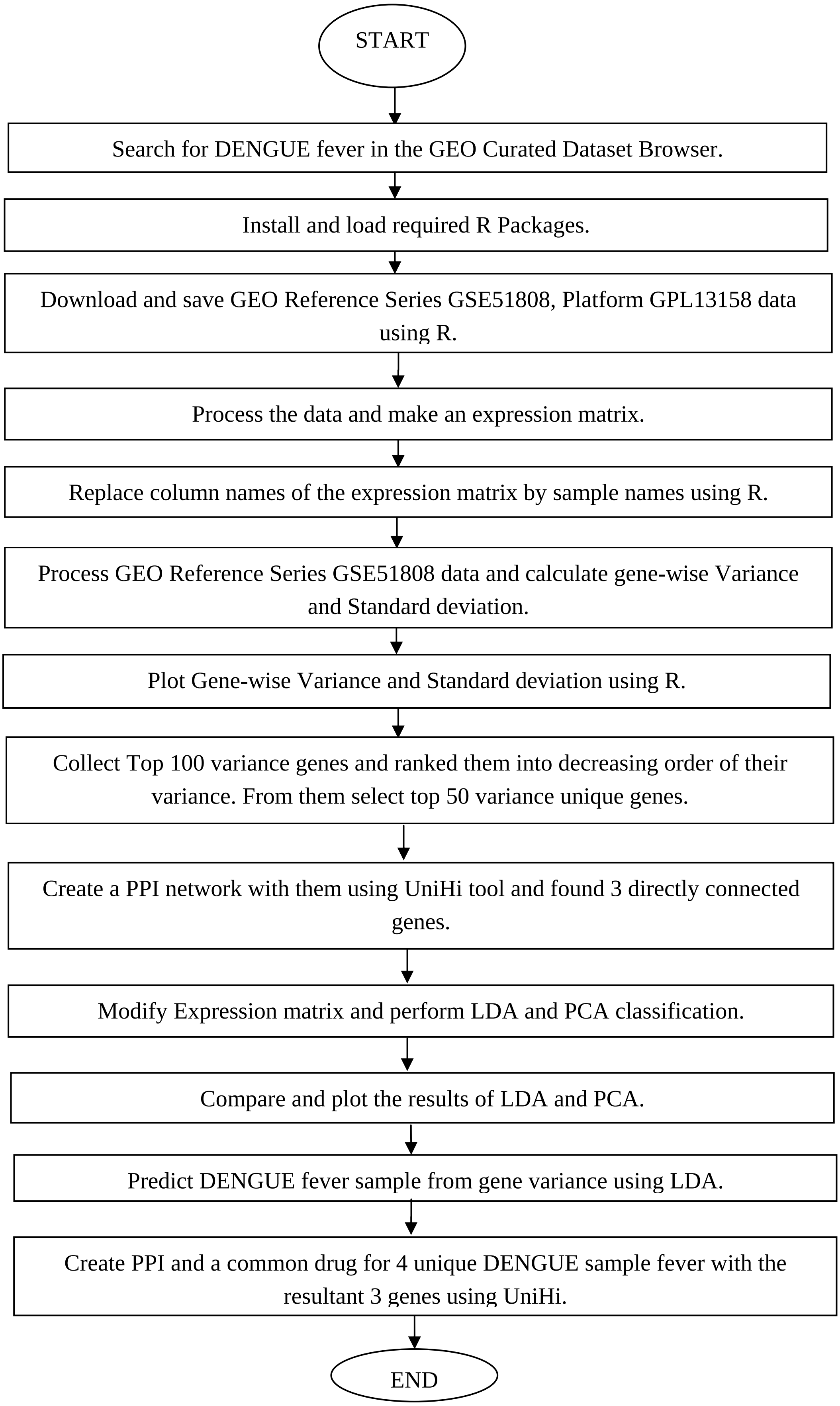

START
Search for DENGUE fever in the GEO Curated Dataset Browser.
Install and load required R Packages.
Download and save GEO Reference Series GSE51808, Platform GPL13158 data using R.
Process the data and make an expression matrix.
Replace column names of the expression matrix by sample names using R.
Process GEO Reference Series GSE51808 data and calculate gene-wise Variance and Standard deviation.
Plot Gene-wise Variance and Standard deviation using R.
Collect Top 100 variance genes and ranked them into decreasing order of their variance. From them select top 50 variance unique genes.
Create a PPI network with them using UniHi tool and found 3 directly connected genes.
Modify Expression matrix and perform LDA and PCA classification.
Compare and plot the results of LDA and PCA.
Predict DENGUE fever sample from gene variance using LDA.
Create PPI and a common drug for 4 unique DENGUE sample fever with the resultant 3 genes using UniHi.
END

Fig. 1 Flowchart of proposed methodology.

3.7 Classification using LDA and PCA

This is the vital step of the research project. In the GEO dataset, there are total 56 samples containing 4 unique samples for DENGUE fever. The classification methods will classify the genes into the unique samples based on their variance value by analyzing 56 samples. To perform gene classification at first install and load required R packages in R. Apply Linear Discriminant Analysis (LDA) on the dataframe created in the previous step and save the result. Similarly Apply Principal Component Analysis (PCA) and save the result.

3.8 Comparison of LDA and PCA result

Plot the result of both the LDA and PCA classification methods. Compare the results and chose the best results between them. As LDA gives better classification result than PCA, it is also used here as a prediction method to predict the unique DENGUE class.

3.9 Prediction of unique samples using LDA in R

During classification method using LDA a dataset is trained. Using that trained dataset other test dataset can be used for prediction with less error. Any other dataset can also be trained and further used to test other dataset using LDA in R.

3.10 Drug design

A drug can be defined as a substance used to treat, cure and prevent an illness, relieve from a pain, or modify some specific process in the body for some specific cure (Habib, 2017). From the directly interacted 3 genes another PPI can be created. Now we can design a common drug for the unique samples which are "Dengue Fever", "Dengue Hemorrhagic Fever", "Convalescent Patient" and "Healthy Control" using 50 genes or using directly connected 3 genes in UniHi tool.

4 Results and Discussion

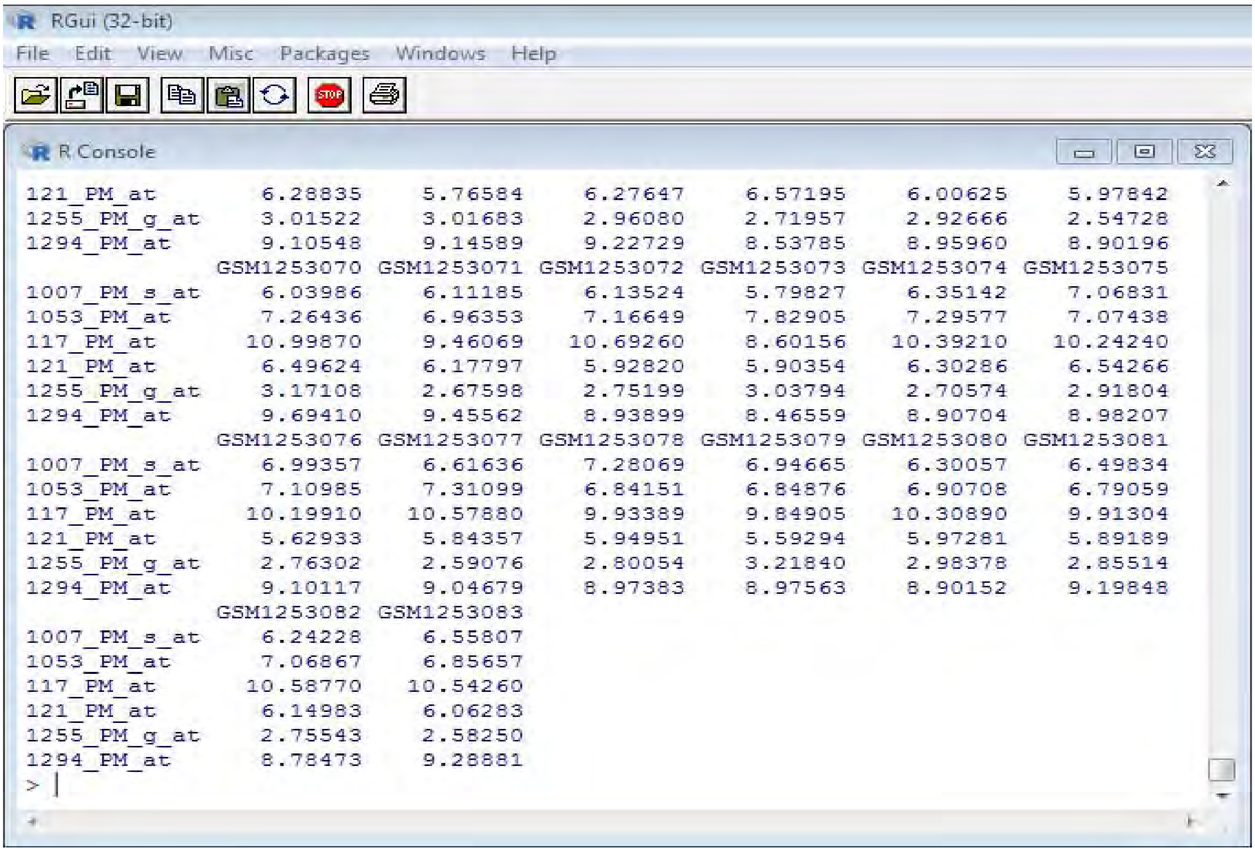
This section provides the result and discusses the output of each step of the proposed methodology of the current research project. The results are described in the following subsections through 4.1 to 4.8.

4.1 Data source

From the Curated Dataset Browser, searching for DENGUE fever results DataSet Record GDS5093, Platform GPL13158, Reference Series GSE51808 and Sample count 56 where there are 4 unique samples.

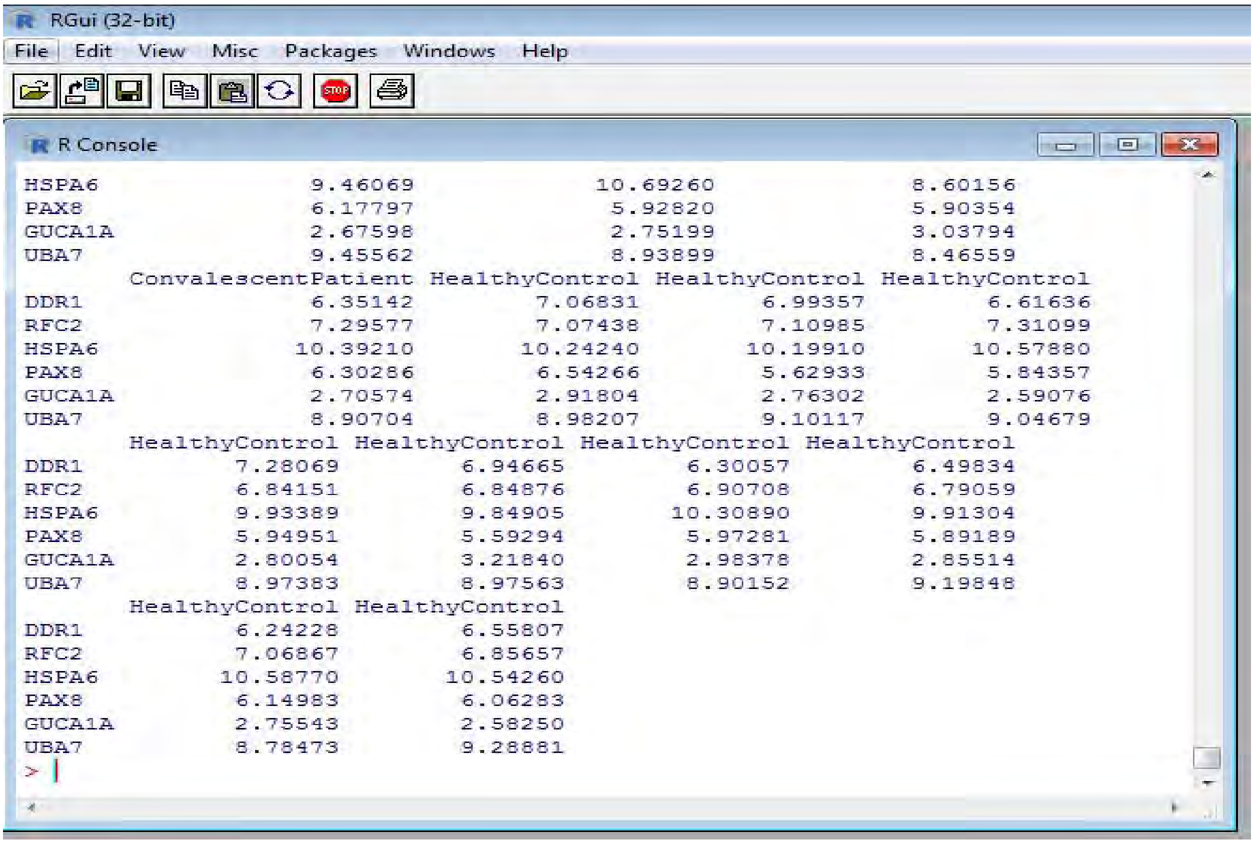
4.2 Data collection and preprocessing

The head of the expression matrix containing gene_id as row names and sample_id as column names are displayed in the Fig. 2. The Fig also shows the corresponding gene appearance value in the expression matrix. Now, the row’s names of the expression matrix are replaced by their corresponding matched gene symbol. There are 56 DENGUE fever samples with four unique samples type such as-"DengueFever", "DengueHemorrhagicFever", "ConvalescentPatient" and "HealthyControl". "Dengue Fever" or “DF” indicates Dengue infected patient at acute infection time point, "Dengue Hemorrhagic Fever", or “DHF” indicates severe Dengue infected patient at acute infection time point, "Convalescent Patient"or “CP” is the third phase of DHF that begins when the Critical Phase ends and is characterized when plasma leak stops and reabsorption begins and "HealthyControl" or “HC” is the healthy or control state indicating not infected by DENV. Replace all of the column’s names by their analogous sample names. After replacement of all the row’s names and column’s names by their corresponding gene symbol and sample name, the head of the expression matrix will look like Fig. 3.



```
121_PM_at        6.28835    5.76584    6.27647    6.57195    6.00625    5.97842
1255_PM_g_at     3.01522    3.01683    2.96080    2.71957    2.92666    2.54728
1294_PM_at       9.10548    9.14589    9.22729    8.53785    8.95960    8.90196
             GSM1253070 GSM1253071 GSM1253072 GSM1253073 GSM1253074 GSM1253075
1007_PM_s_at     6.03986    6.11185    6.13524    5.79827    6.35142    7.06831
1053_PM_at       7.26436    6.96353    7.16649    7.82905    7.29577    7.07438
117_PM_at       10.99870    9.46069   10.69260    8.60156   10.39210   10.24240
121_PM_at        6.49624    6.17797    5.92820    5.90354    6.30286    6.54266
1255_PM_g_at     3.17108    2.67598    2.75199    3.03794    2.70574    2.91804
1294_PM_at       9.69410    9.45562    8.93899    8.46559    8.90704    8.98207
             GSM1253076 GSM1253077 GSM1253078 GSM1253079 GSM1253080 GSM1253081
1007_PM_s_at     6.99357    6.61636    7.28069    6.94665    6.30057    6.49834
1053_PM_at       7.10985    7.31099    6.84151    6.84876    6.90708    6.79059
117_PM_at       10.19910   10.57880    9.93389    9.84905   10.30890    9.91304
121_PM_at        5.62933    5.84357    5.94951    5.59294    5.97281    5.89189
1255_PM_g_at     2.76302    2.59076    2.80054    3.21840    2.98378    2.85514
1294_PM_at       9.10117    9.04679    8.97383    8.97563    8.90152    9.19848
             GSM1253082 GSM1253083
1007_PM_s_at     6.24228    6.55807
1053_PM_at       7.06867    6.85657
117_PM_at       10.58770   10.54260
121_PM_at        6.14983    6.06283
1255_PM_g_at     2.75543    2.58250
1294_PM_at       8.78473    9.28881
> |
```

Fig. 2 Head of the Expression matrix with gene_id and sample_id as row's names and column's names.



```
HSPA6              9.46069             10.69260              8.60156
PAX8               6.17797              5.92820              5.90354
GUCA1A             2.67598              2.75199              3.03794
UBA7               9.45562              8.93899              8.46559
       ConvalescentPatient HealthyControl HealthyControl HealthyControl
DDR1               6.35142        7.06831        6.99357        6.61636
RFC2               7.29577        7.07438        7.10985        7.31099
HSPA6             10.39210       10.24240       10.19910       10.57880
PAX8               6.30286        6.54266        5.62933        5.84357
GUCA1A             2.70574        2.91804        2.76302        2.59076
UBA7               8.90704        8.98207        9.10117        9.04679
       HealthyControl HealthyControl HealthyControl HealthyControl
DDR1          7.28069        6.94665        6.30057        6.49834
RFC2          6.84151        6.84876        6.90708        6.79059
HSPA6         9.93389        9.84905       10.30890        9.91304
PAX8          5.94951        5.59294        5.97281        5.89189
GUCA1A        2.80054        3.21840        2.98378        2.85514
UBA7          8.97383        8.97563        8.90152        9.19848
       HealthyControl HealthyControl
DDR1          6.24228        6.55807
RFC2          7.06867        6.85657
HSPA6        10.58770       10.54260
PAX8          6.14983        6.06283
GUCA1A        2.75543        2.58250
UBA7          8.78473        9.28881
> |
```

Fig. 3 Head of the Expression matrix with gene_symbol and sample_name as row's names and column's names.

4.3 Variance and standard deviation

Calculate the variance and standard deviation for all of the genes of all the samples and plot them using R, which is shown in Fig. 4.

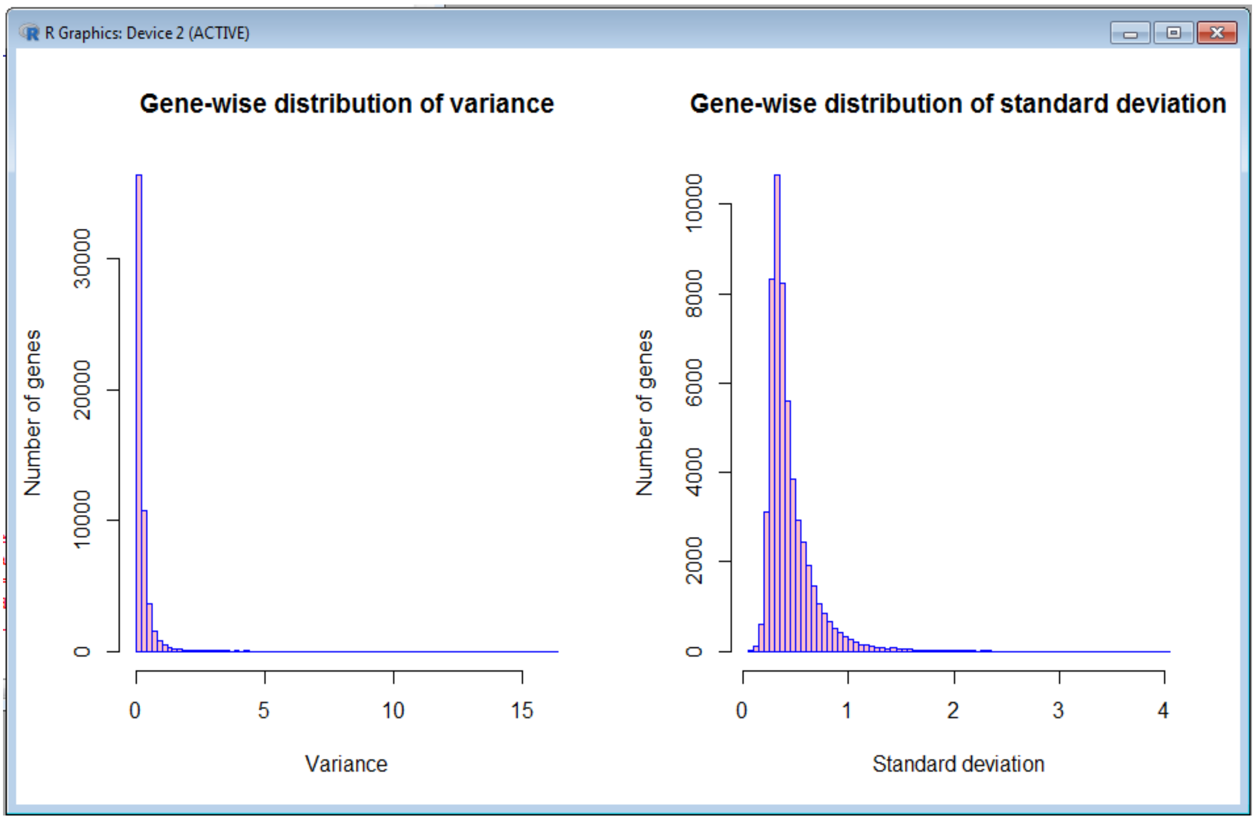


Fig. 4 Gene-wise distribution of variance and standard deviation plot.

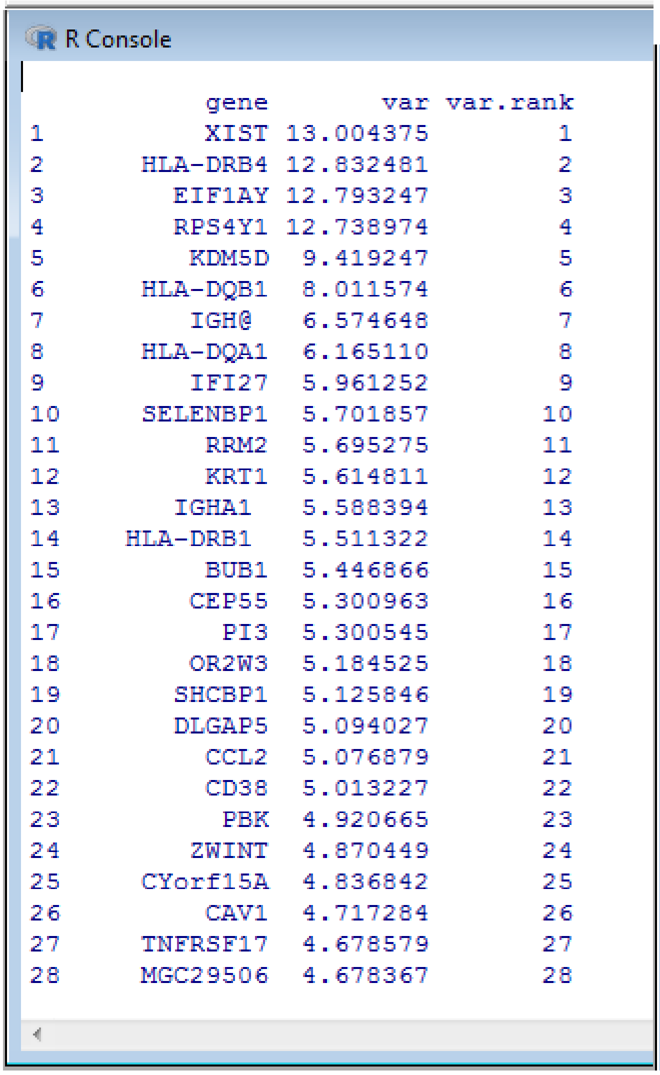
Collect top 100 ranked variance genes which contain 89 unique genes and put them in a table. Select top 50 ranked genes by modifying the table from the unique 89 genes and exposed them using Fig. 5 and 6. Genes are ranked according to their variance value such that highest variance gene is ranked as 1 and the lowest variance gene is ranked as 50. In the following figures, Fig. 5 shows the top 28 genes where Fig. 6 displays the bottom 22 genes from the unique 50 top ranked genes. These 50 ranked genes are further used to create PPI and Regulatory Interaction Networks.

4.4 Find interacted genes and create PPI network

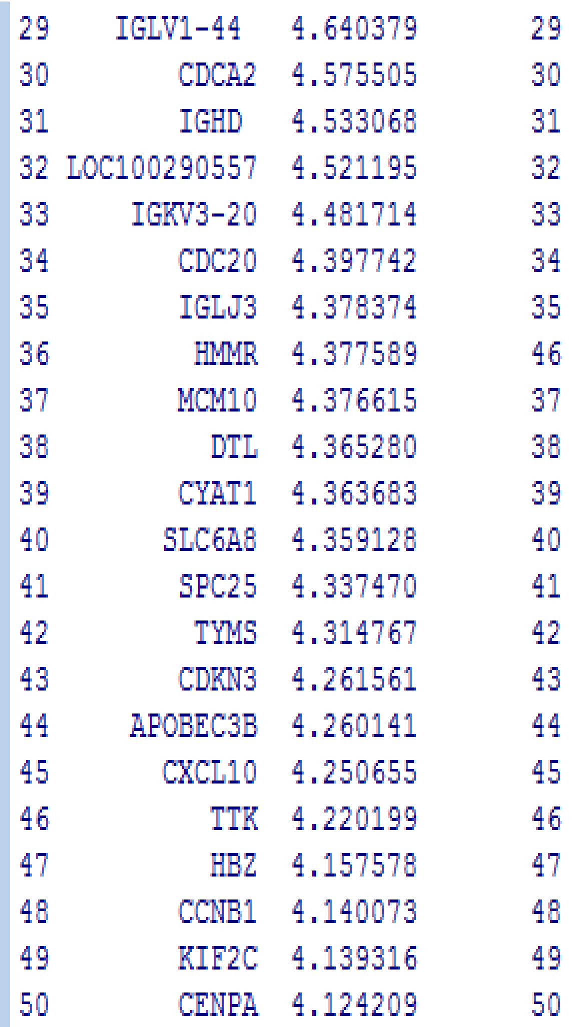
From the 50 ranked genes, using UniHi tool PPI network is created which can be demonstrated as in Fig. 7. In the network some of the genes are directly connected where other has indirect connection with each other. In Fig. 7, large circle containing gene name indicates genes and small yellow circle indicates protein. Regulatory Interaction Network is used to show the directly interacted genes which are linked by red line as shown in Fig. 8. From Fig. 8, three directly interacted genes are found. They are CDK1, PBK, CCL2 and connected by red line to the proteins or itself. These 3 genes are the main assets and will be used in further analysis.

4.5 Data processing and classification

Using the resultant 3 genes a dataframe is created from the table of step 3.4. Fig. 9 shows head of the dataframe. In the dataframe, there are 3 rows which are the resultant genes and 56 columns which are the 56 samples name. To perform classification on the dataframe, it needs to be transposed.



```
            gene       var var.rank
1           XIST 13.004375        1
2       HLA-DRB4 12.832481        2
3         EIF1AY 12.793247        3
4         RPS4Y1 12.738974        4
5          KDM5D  9.419247        5
6       HLA-DQB1  8.011574        6
7           IGH@  6.574648        7
8       HLA-DQA1  6.165110        8
9          IFI27  5.961252        9
10      SELENBP1  5.701857       10
11          RRM2  5.695275       11
12          KRT1  5.614811       12
13        IGHA1   5.588394       13
14      HLA-DRB1  5.511322       14
15          BUB1  5.446866       15
16         CEP55  5.300963       16
17           PI3  5.300545       17
18         OR2W3  5.184525       18
19        SHCBP1  5.125846       19
20        DLGAP5  5.094027       20
21          CCL2  5.076879       21
22          CD38  5.013227       22
23           PBK  4.920665       23
24         ZWINT  4.870449       24
25      CYorf15A  4.836842       25
26          CAV1  4.717284       26
27      TNFRSF17  4.678579       27
28      MGC29506  4.678367       28
```



```
29    IGLV1-44  4.640379       29
30       CDCA2  4.575505       30
31        IGHD  4.533068       31
32 LOC100290557 4.521195       32
33    IGKV3-20  4.481714       33
34       CDC20  4.397742       34
35       IGLJ3  4.378374       35
36        HMMR  4.377589       46
37       MCM10  4.376615       37
38         DTL  4.365280       38
39       CYAT1  4.363683       39
40      SLC6A8  4.359128       40
41       SPC25  4.337470       41
42        TYMS  4.314767       42
43       CDKN3  4.261561       43
44    APOBEC3B  4.260141       44
45      CXCL10  4.250655       45
46         TTK  4.220199       46
47         HBZ  4.157578       47
48       CCNB1  4.140073       48
49       KIF2C  4.139316       49
50       CENPA  4.124209       50
```

Fig. 5 Top 28 ranked genes of top 50 unique genes.

Fig. 6 28 to 50 ranked genes of top 50 unique genes.

The transposed dataframe contains 56 rows and 4 columns. The head of the transposed dataframe is displayed in Fig. 10. To perform classification using LDA, at least one column should be categorical. From Fig. 10, it is noticed that column ‘type’ contains categorical values; hence it is the categorical column of the dataframe. Now, LDA technique is performed on the dataframe using R. LDA techniques classifies the DENGUE fever samples into 4 unique samples, in that LD1 classifies 96.16% and LD2 classifies 3.62% data. Applying PCA on the same dataframe using R, it is found that PC1 classifies 45.98% and PC2 classifies 30.49% data. The classification using these 2 methods are shown in Fig. 11 and 12 respectively.

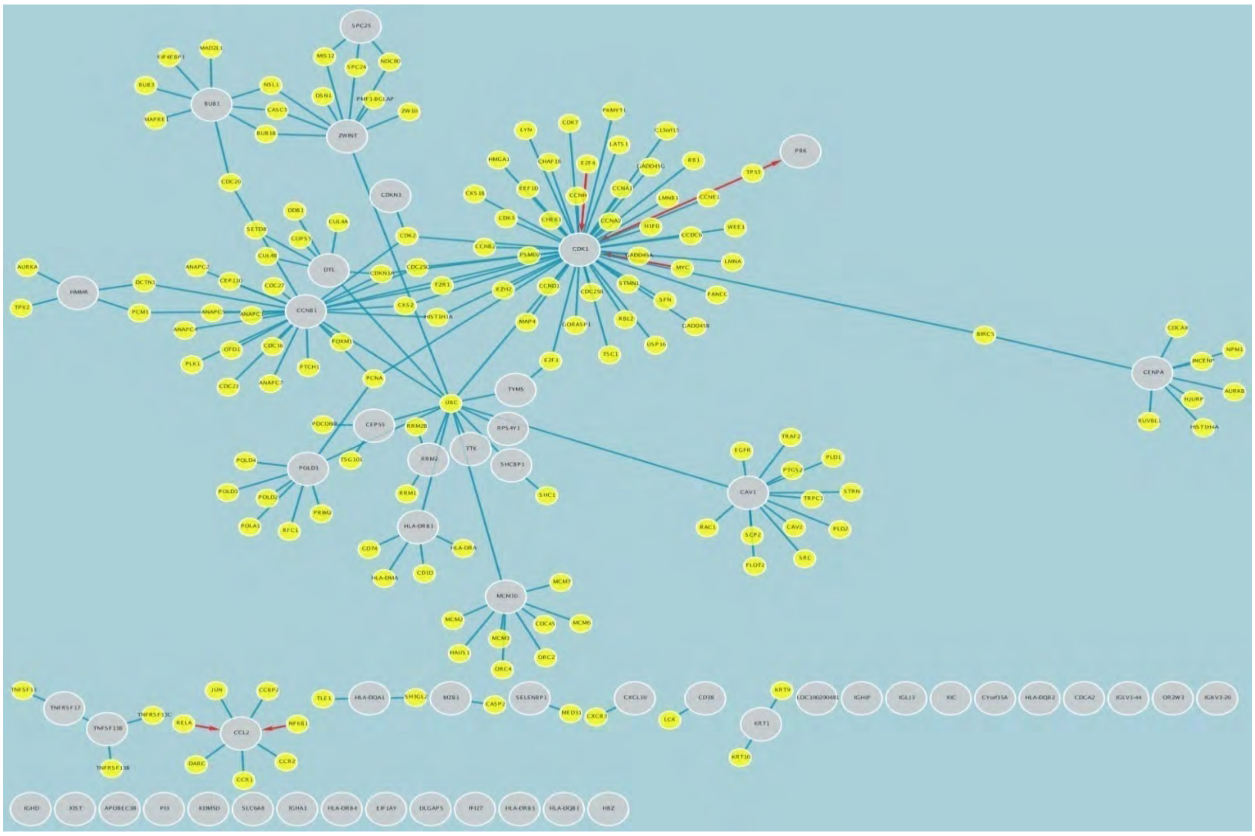


Fig. 7 PPI with top 50 genes.

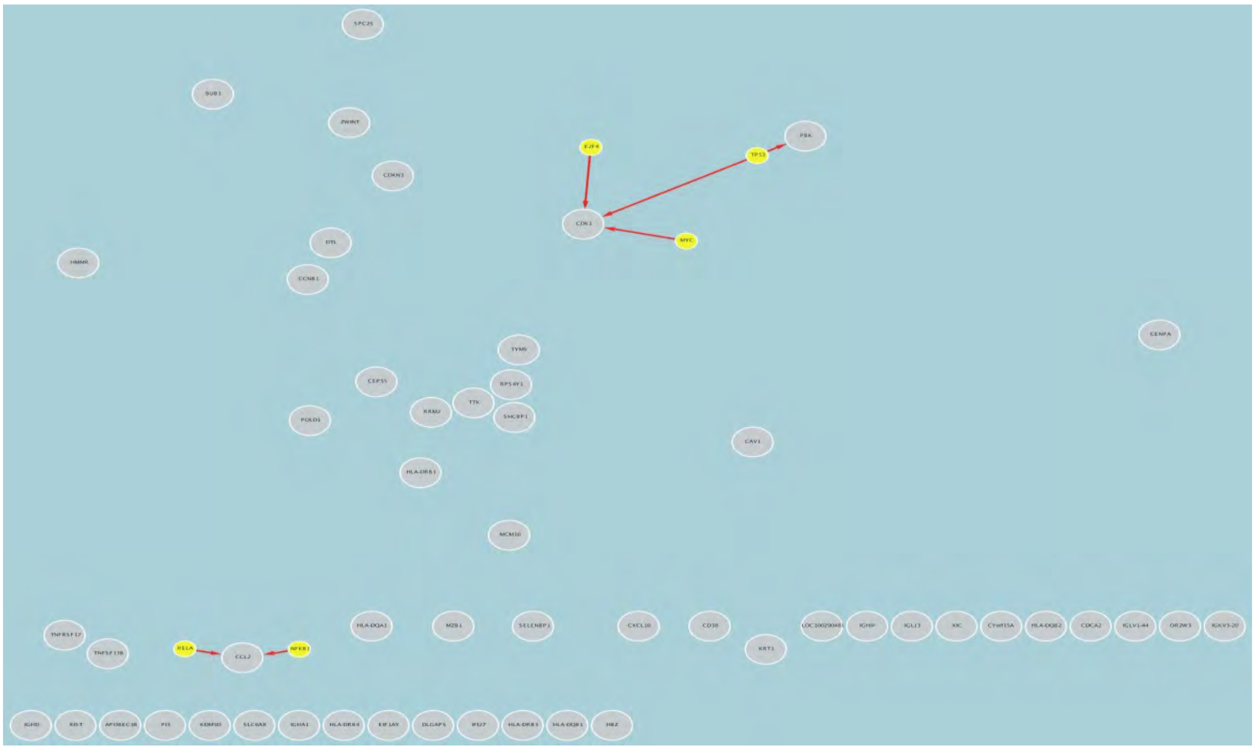
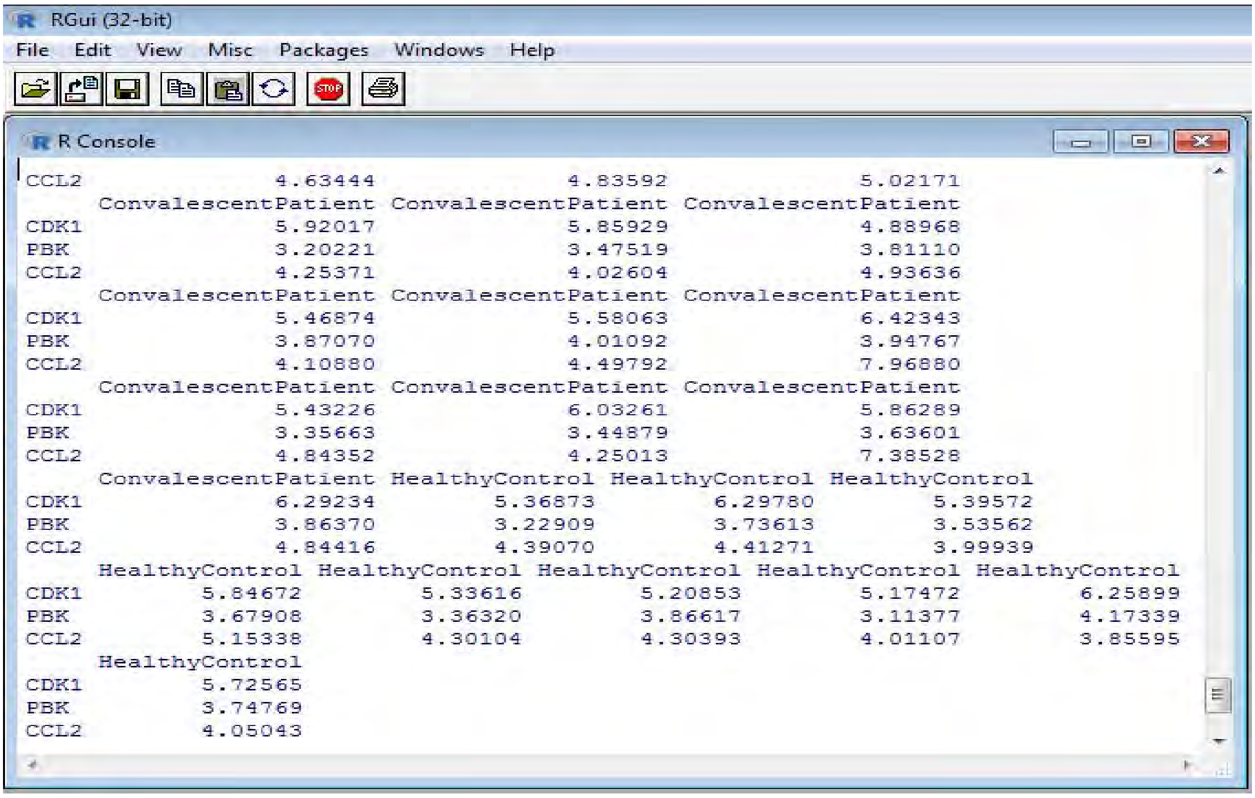
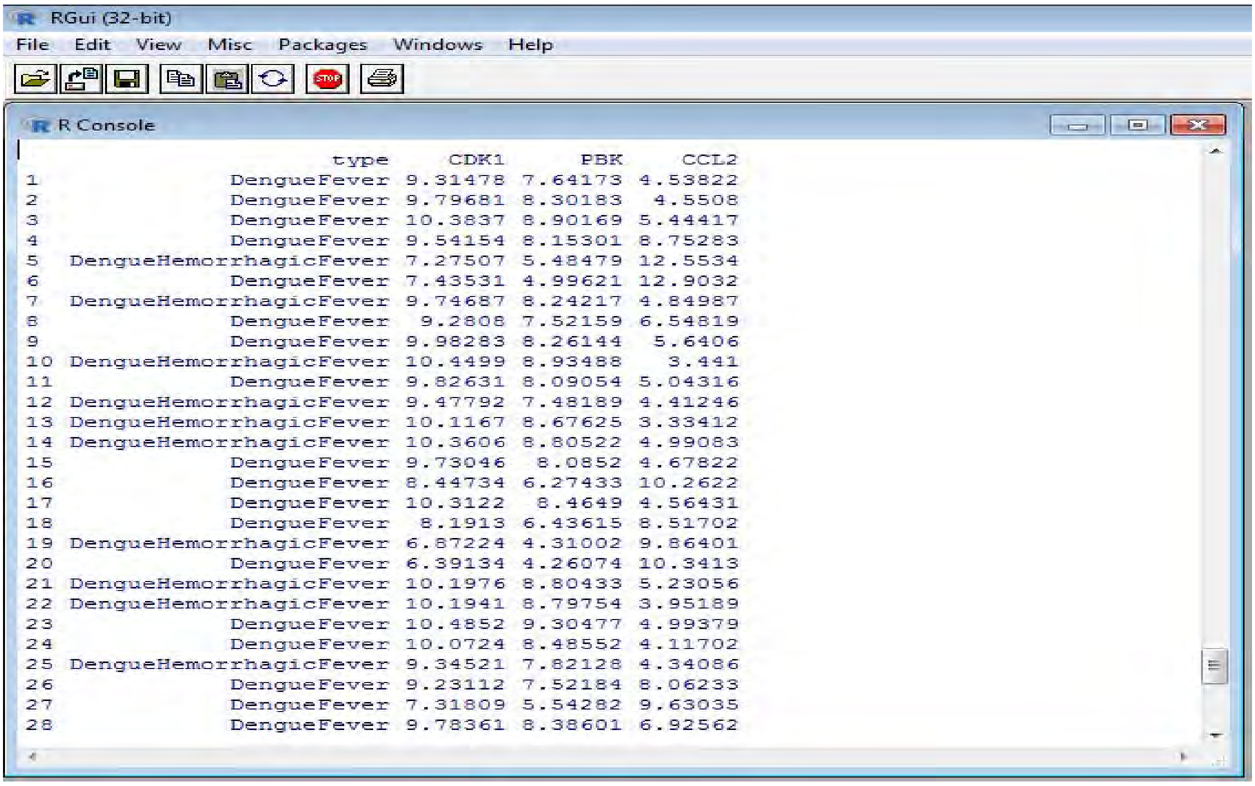


Fig. 8 Regulatory Interaction Network with top 50 genes.



```
CCL2             4.63444             4.83592             5.02171
     ConvalescentPatient ConvalescentPatient ConvalescentPatient
CDK1             5.92017             5.85929             4.88968
PBK              3.20221             3.47519             3.81110
CCL2             4.25371             4.02604             4.93636
     ConvalescentPatient ConvalescentPatient ConvalescentPatient
CDK1             5.46874             5.58063             6.42343
PBK              3.87070             4.01092             3.94767
CCL2             4.10880             4.49792             7.96880
     ConvalescentPatient ConvalescentPatient ConvalescentPatient
CDK1             5.43226             6.03261             5.86289
PBK              3.35663             3.44879             3.63601
CCL2             4.84352             4.25013             7.38528
     ConvalescentPatient HealthyControl HealthyControl HealthyControl
CDK1             6.29234        5.36873        6.29780        5.39572
PBK              3.86370        3.22909        3.73613        3.53562
CCL2             4.84416        4.39070        4.41271        3.99939
     HealthyControl HealthyControl HealthyControl HealthyControl HealthyControl
CDK1        5.84672        5.33616        5.20853        5.17472        6.25899
PBK         3.67908        3.36320        3.86617        3.11377        4.17339
CCL2        5.15338        4.30104        4.30393        4.01107        3.85595
     HealthyControl
CDK1        5.72565
PBK         3.74769
CCL2        4.05043
```

Fig. 9 Head of dataframe with resultant genes as row and samples name as column.



```
                     type    CDK1     PBK    CCL2
1             DengueFever 9.31478 7.64173 4.53822
2             DengueFever 9.79681 8.30183  4.5508
3             DengueFever 10.3837 8.90169 5.44417
4             DengueFever 9.54154 8.15301 8.75283
5  DengueHemorrhagicFever 7.27507 5.48479 12.5534
6             DengueFever 7.43531 4.99621 12.9032
7  DengueHemorrhagicFever 9.74687 8.24217 4.84987
8             DengueFever  9.2808 7.52159 6.54819
9             DengueFever 9.98283 8.26144  5.6406
10 DengueHemorrhagicFever 10.4499 8.93488   3.441
11            DengueFever 9.82631 8.09054 5.04316
12 DengueHemorrhagicFever 9.47792 7.48189 4.41246
13 DengueHemorrhagicFever 10.1167 8.67625 3.33412
14 DengueHemorrhagicFever 10.3606 8.80522 4.99083
15            DengueFever 9.73046  8.0852 4.67822
16            DengueFever 8.44734 6.27433 10.2622
17            DengueFever 10.3122  8.4649 4.56431
18            DengueFever  8.1913 6.43615 8.51702
19 DengueHemorrhagicFever 6.87224 4.31002 9.86401
20            DengueFever 6.39134 4.26074 10.3413
21 DengueHemorrhagicFever 10.1976 8.80433 5.23056
22 DengueHemorrhagicFever 10.1941 8.79754 3.95189
23            DengueFever 10.4852 9.30477 4.99379
24            DengueFever 10.0724 8.48552 4.11702
25 DengueHemorrhagicFever 9.34521 7.82128 4.34086
26            DengueFever 9.23112 7.52184 8.06233
27            DengueFever 7.31809 5.54282 9.63035
28            DengueFever 9.78361 8.38601 6.92562
```

Fig. 10 Head of the transposed dataframe.

In Fig. 11 and 12 displayed below, LDA and PCA results are represented seperately. Both of the figures use four separate color for four unique samples. Based on the gene variance, LDA technique classifies much more data than PCA technique. So the figures explicit the fact that, LDA gives better result and mostly preferable here than PCA.

4.6 Comparison of LDA and PCA result

In the previous step, LDA and PCA results are plotted using separate figures and also their comparison are discussed. This step shows the comparison on a single figure. Fig. 13 gives an expressive view of the LDA and PCA result. The figure uses four sign of four colors to indicate the classified unique samples.

4.7 Prediction of unique samples using LDA in R

In the classification method, at first a dataset is trained and then the trained dataset is used to predict classes from other test dataset. So, during classification method using LDA the dataframe created in step 3.7 is trained. Then to predict sample from this dataframe, it is now used as test data against the previously trained dataset. As there are 56 rows in the dataframe LDA predicts 56 samples for them as shown in Fig. 14. The prediction technique should predict the sample with no error or less error. Using LDA, it is also tried to predict with negligible error. In Fig. 15, a table is used to show the actual versus predicted results. Here, the average error is 0.27.

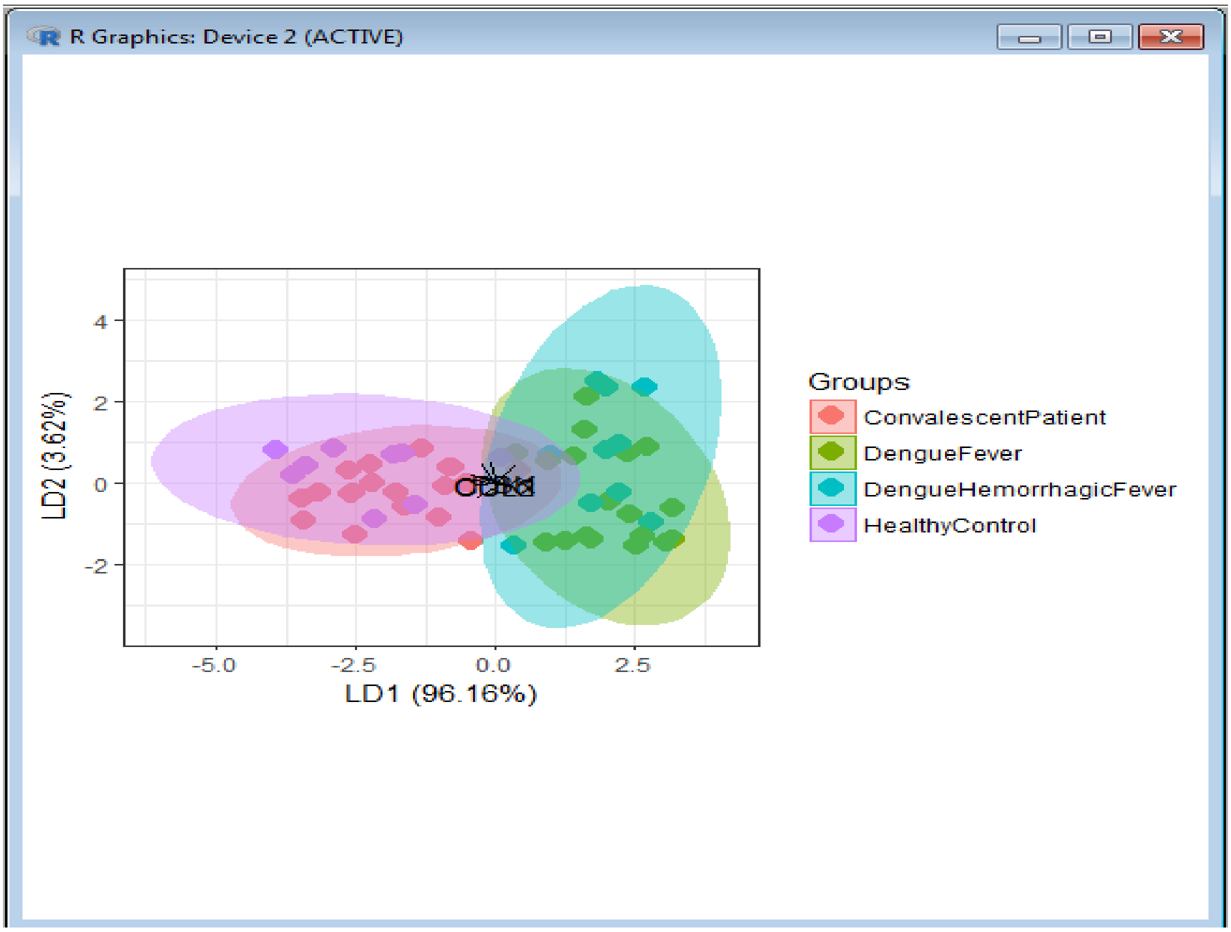


Fig. 11 Gene Classification into 4 Unique Samples using LDA in R.

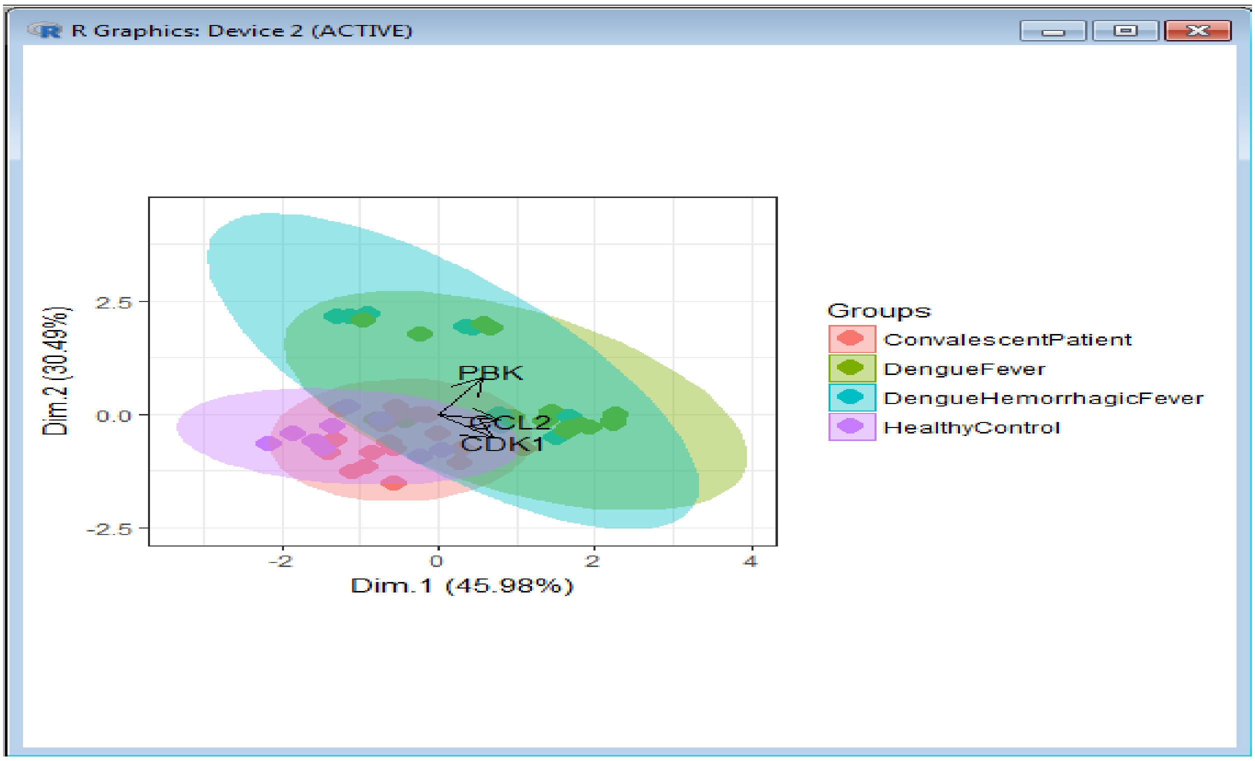


Fig. 12 Gene Classification into 4 Unique Samples using PCA in R.

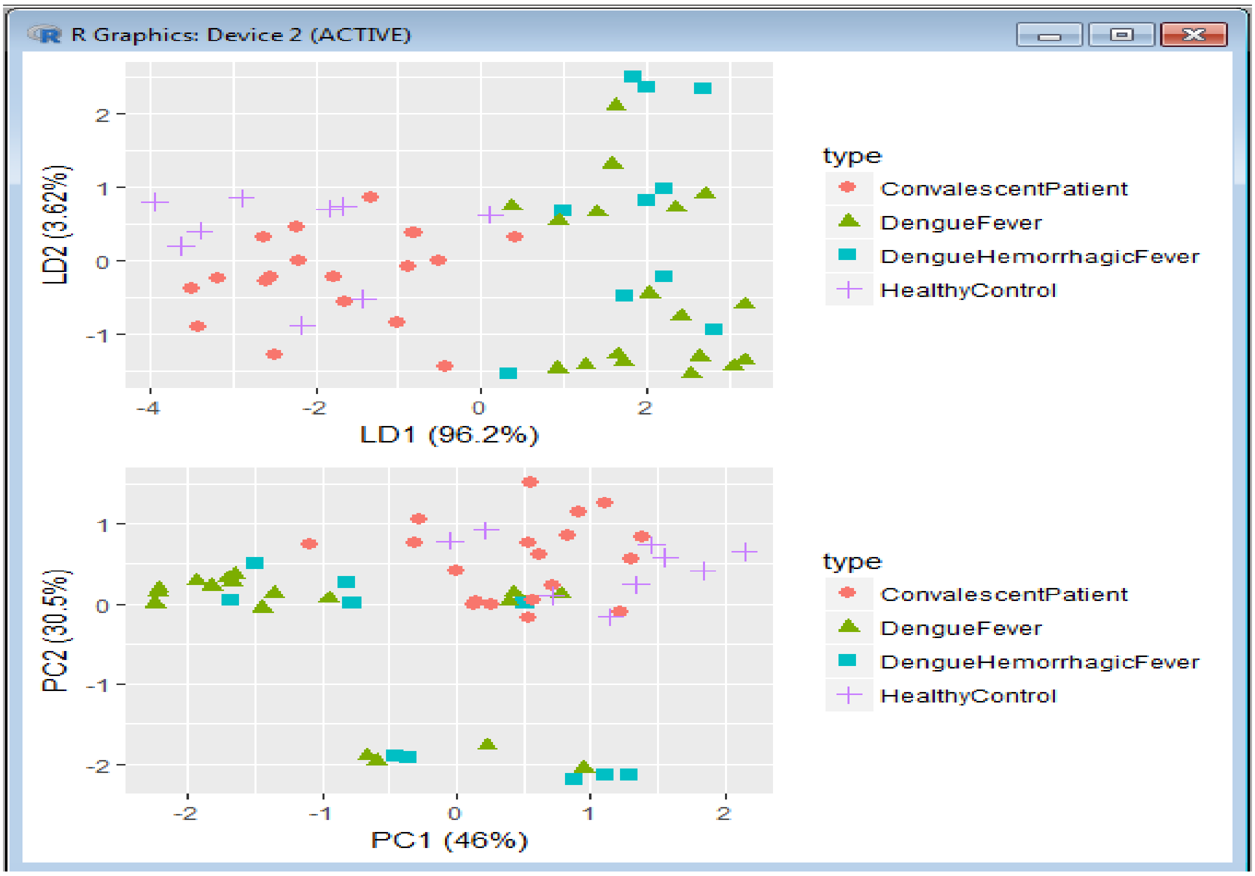
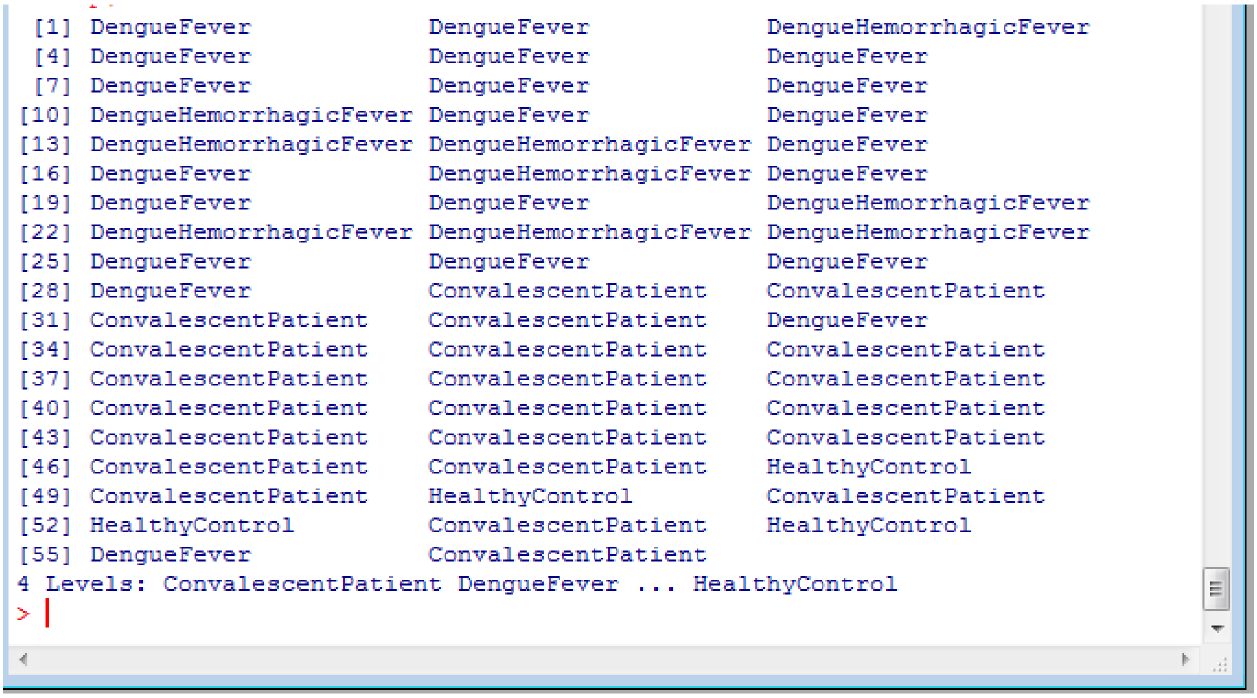
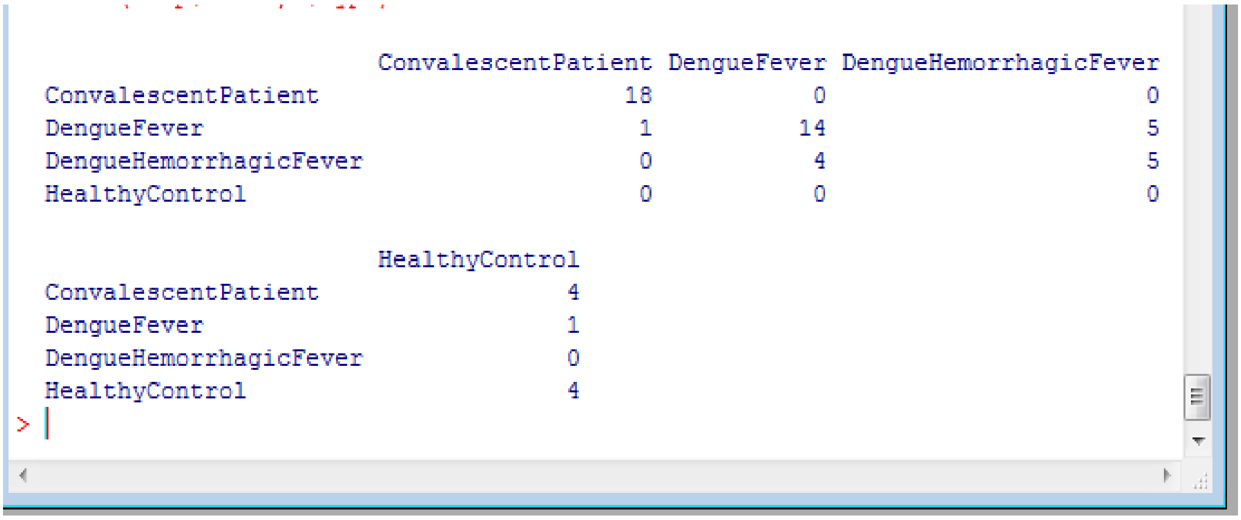


Fig. 13 Comparison plot of LDA and PCA results.



```
 [1] DengueFever               DengueFever               DengueHemorrhagicFever
 [4] DengueFever               DengueFever               DengueFever
 [7] DengueFever               DengueFever               DengueFever
[10] DengueHemorrhagicFever DengueFever               DengueFever
[13] DengueHemorrhagicFever DengueHemorrhagicFever DengueFever
[16] DengueFever               DengueHemorrhagicFever DengueFever
[19] DengueFever               DengueFever               DengueHemorrhagicFever
[22] DengueHemorrhagicFever DengueHemorrhagicFever DengueHemorrhagicFever
[25] DengueFever               DengueFever               DengueFever
[28] DengueFever               ConvalescentPatient    ConvalescentPatient
[31] ConvalescentPatient    ConvalescentPatient    DengueFever
[34] ConvalescentPatient    ConvalescentPatient    ConvalescentPatient
[37] ConvalescentPatient    ConvalescentPatient    ConvalescentPatient
[40] ConvalescentPatient    ConvalescentPatient    ConvalescentPatient
[43] ConvalescentPatient    ConvalescentPatient    ConvalescentPatient
[46] ConvalescentPatient    ConvalescentPatient    HealthyControl
[49] ConvalescentPatient    HealthyControl         ConvalescentPatient
[52] HealthyControl         ConvalescentPatient    HealthyControl
[55] DengueFever               ConvalescentPatient
4 Levels: ConvalescentPatient DengueFever ... HealthyControl
> 
```

Fig. 14 Predicted samples for test data using LDA.



```
                        ConvalescentPatient DengueFever DengueHemorrhagicFever
  ConvalescentPatient                    18           0                      0
  DengueFever                             1          14                      5
  DengueHemorrhagicFever                  0           4                      5
  HealthyControl                          0           0                      0

                        HealthyControl
  ConvalescentPatient                4
  DengueFever                        1
  DengueHemorrhagicFever             0
  HealthyControl                     4
> 
```

Fig. 15 Original Samples versus predicted samples using LDA in R.

4.8 Drug design

Any substance used to treat or prevent an illness or disease when absorbed can be defined as a drug. From the PPI of step 4.4, three genes results which can be used again to create PPI and Regulatory Interaction Network as depicted in Fig. 16 and 17 respectively. The drug should be designed and developed in such a way that it does not disturb the normal chemical process of the body and only affect the target protein (Habib et. al., 2017). Along with target identification specific drug needs to be disclosed to dispose a disease. From these PPI and Regulatory Interaction Networks a common drug can be designed using UniHi tool. The Drug target can be designed for top 50 unique genes or the resultant 3 genes. Here, in Fig. 18, the drug target using 3 resultant genes is shown. After filtering the drug target, the affected and unaffected proteins in the structure can be identified, this is demonstrated in Fig. 19.

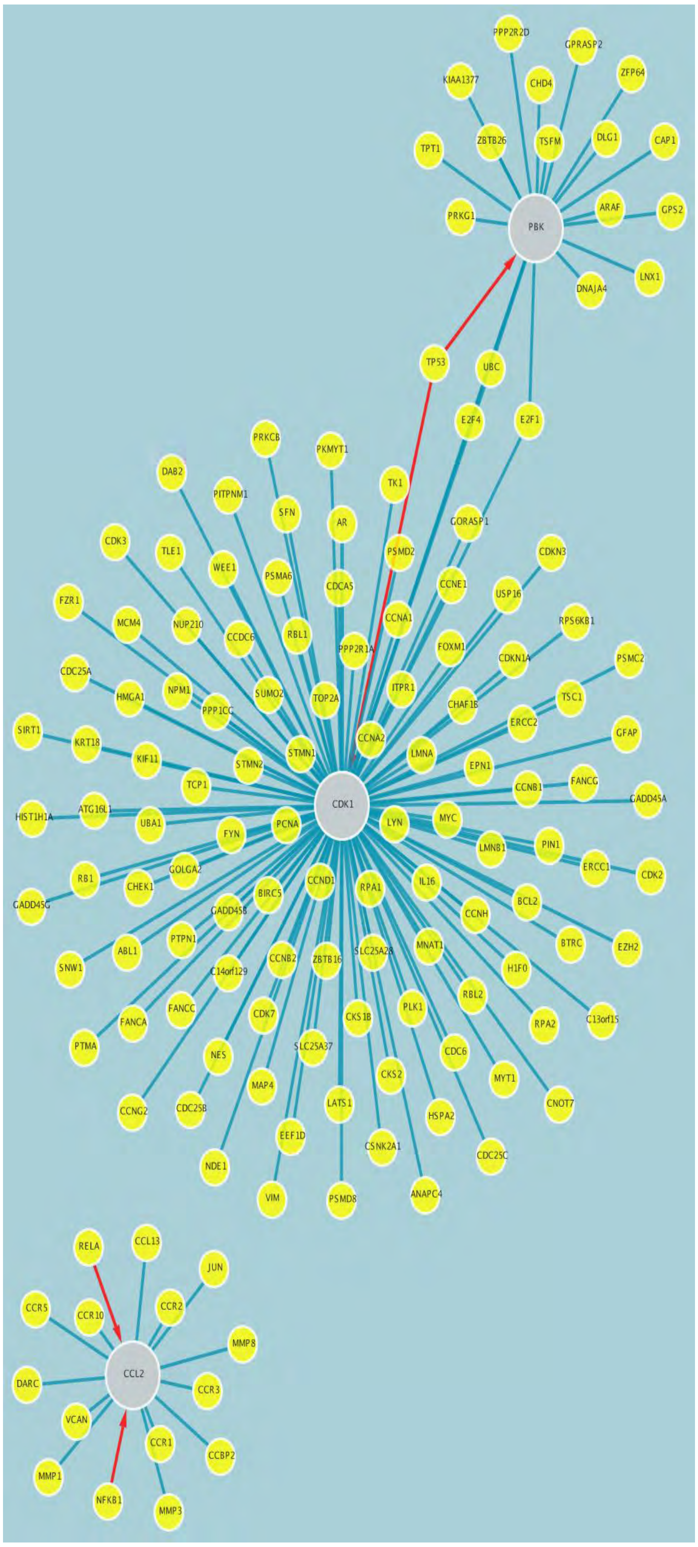


Fig. 16 PPI with 3 genes.

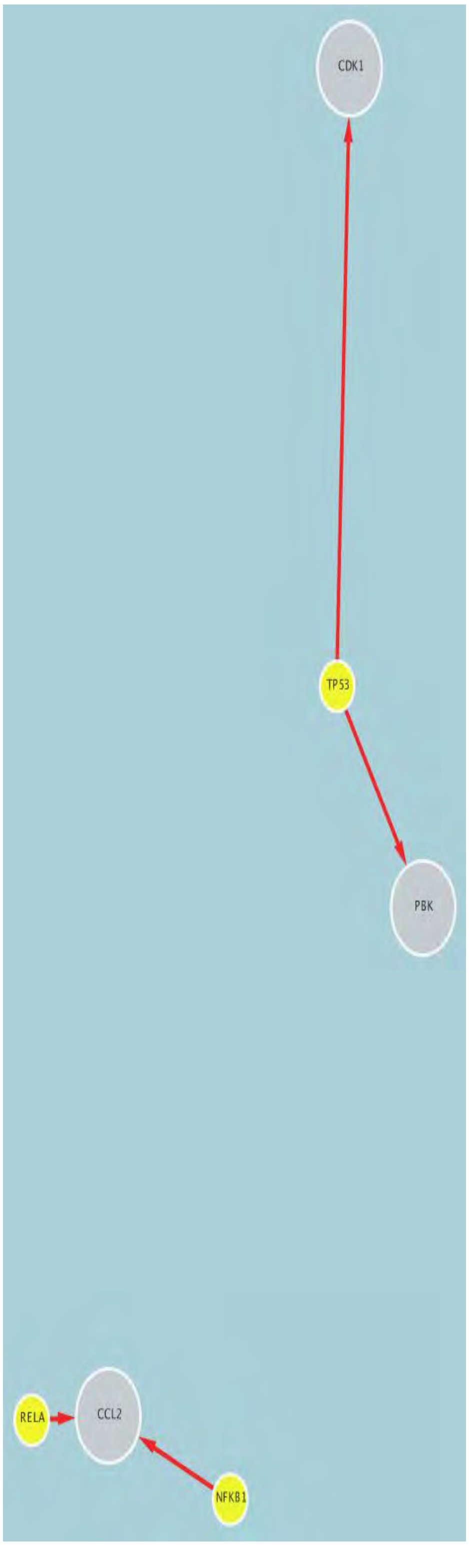


Fig. 17 Regulatory Interaction Network with 3 genes.

The drug target in Fig.18 is the common drugs that can be used to treat DF, DHF and CP but in different proportion. Thus only one drug can be used to remedy all DENGUE unique samples. Filtering this drug give more explicit structure as in Fig. 19.

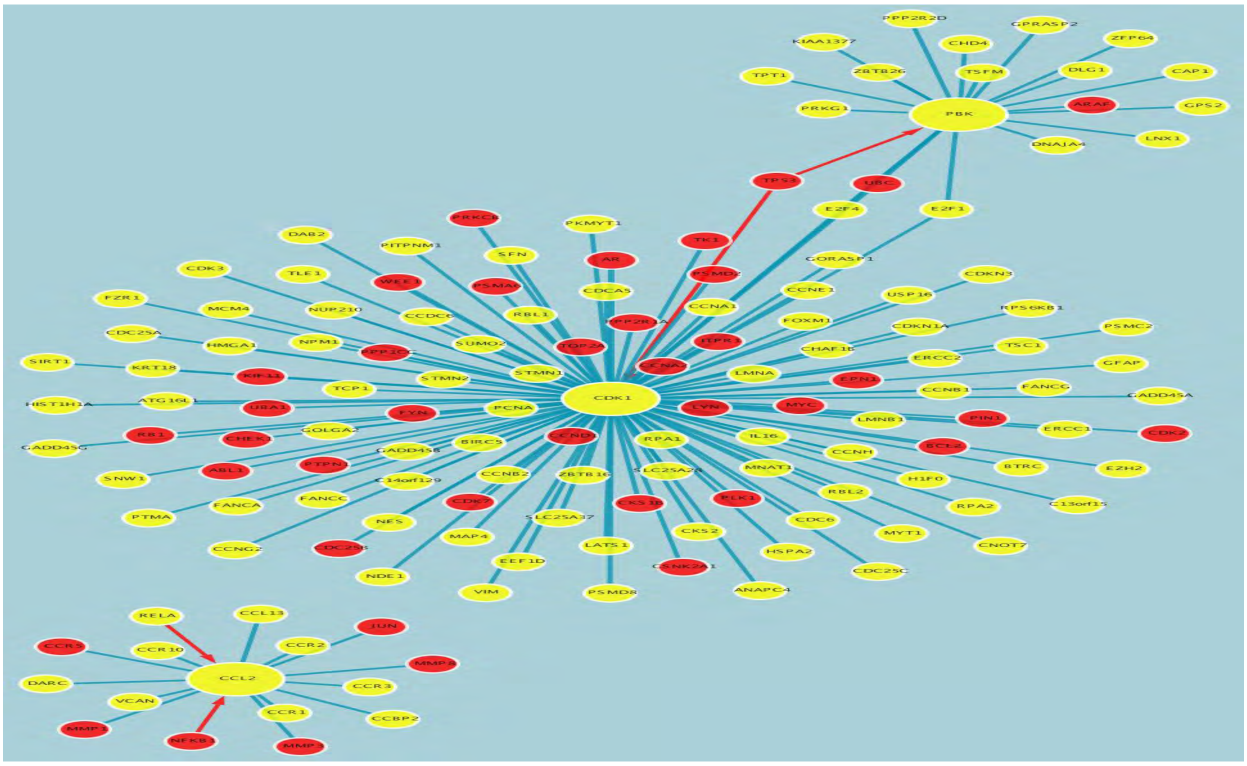


Fig. 18 Drug target network with resulting 3 genes.

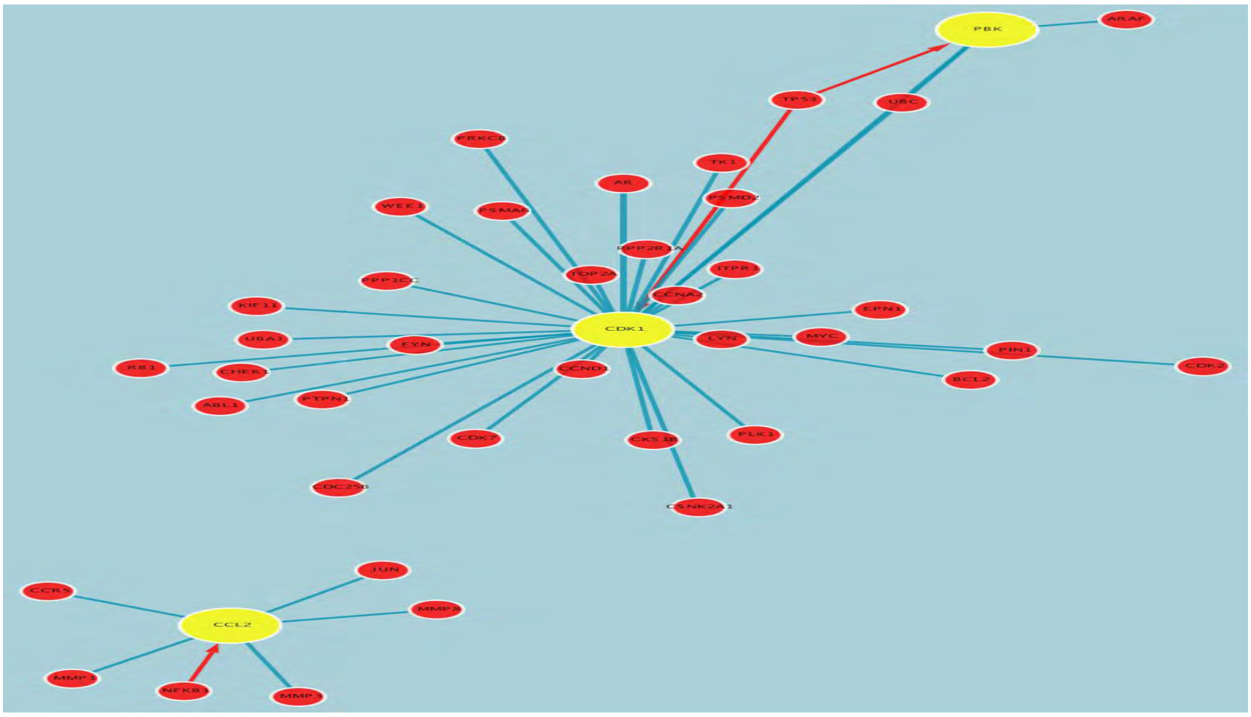


Fig. 19 Drug target network of fig. 18 after filtering.

5 Conclusion

The developments of Bioinformatics tools have disclosed new research area and made uncompromising task easier (Habib et. al., 2017). Presented research work is an application of Bioinformatics. Much more tasks are performed during this research project, which may reveal new research area for other researchers. R Programming Language has a great contribution here because most of the analysis are performed using R. This research is mainly helpful to understand the GEO dataset, Microarray data, PPI, Regulatory Interaction Network, Drug Target, Drug Target Filtering, Classification methods, LDA, PCA, Prediction method etc. At first GEO datasets for DENGUE fever are downloaded using R. Then the numbers of collected genes are reduced after performing several steps of preprocessing, processing and mining, and PPIs, Regulatory Interaction Networks are designed. After that, LDA and PCA methods are applied on the analyzed data and a comparison is made. Based on the best results a prediction method is also used with less error. And at the last step, a common drug for all unique samples is also designed. Therefore, this research creates a new dimension of using R which is the main potentiality of the research work. The future study of this research is to analyze other GEO microarray dataset and come up with a better outcome.

Abbreviations

LDA = Linear Discriminant Analysis; PCA=Principal Component Analysis; PPI = Protein-Protein Interaction; GEO = Gene Expression Omnibus; DENV = Dengue Virus.

Acknowledgment

The authors are grateful to the participants who contributed to this research. No financial supports are provided from any organization for this research project.

References

Analyticsvidhya. Last access: 20/07/17. https://www.analyticsvidhya.com/blog/2016/03/practical-guide-principal-component-analysis-python/

Balakrishnama S, Ganapathiraju A. 1995. Institute for Signal and Information Processing Linear Discriminant Analysis - a Brief Tutorial

Barrett T, Wilhite SE, Ledoux P, Evangelista C, Kim IF, Tomashevsky M, Marshall KA, Phillippy KH, Sherman PM, Holko M, Yefanov A, Lee H, Zhang N, Robertson CL, Serova N, Davis S, Soboleva A. 2013. NCBI GEO: archive for functional genomics data sets--update. Nucleic Acids Research, 41(Database issue): D991-5

Barrett T, et al. 2005. NCBI GEO: mining millions of expression profiles - database and tools. Nucleic Acids Res., 33: D562-D566

Barrett T, Edgar R. 2006. Gene Expression Omnibus: Microarray Data Storage, Submission, Retrieval, and Analysis. Methods in Enzymology, 411: 352-369

Bioinformatics (Wikipedia). Last access: 11/07/17. https://en.wikipedia.org/wiki/Bioinformatics. DM Mutch et. al. 2001. Genome Biology, 2(12)

Edgar R, Domrachev M, Lash AE. 2002. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Research, 30: 207-210

Holter NS, et al. 2000. Fundamental patterns underlying gene expression profiles: simplicity from complexity, Proceedings of National Academy of Sciences of USA, 97: 8409-8414

Jackson J. 1991. A Users Guide to Principal Components. Wiley & Sons, USA

Jesmin T, Waheed S, Emran AA. 2016. Investigation of common disease regulatory network for metabolic disorders: A bioinformatics approach. Network Biology, 6(1): 28-36

Jieping Ye, Ravi Janardan, Qi Li. 2004. Two-dimensional linear discriminant analysis. Advances in Neural Information Processing Systems, 17: 1569-1576

Klingstrom T, Plwczynski D. 2010. Protein-protein interaction and pathway databases, a graphical review. Briefings in Bioinformatics, 12(6): 702-713

LDA [Wikipedia]. Last access: 20/07/17. https://en.wikipedia.org/wiki/Linear_discriminant_analysis.

Luscombe NM, Greenbaum D, Gerstein M. 2001. What is bioinformatics? A proposed definition and overview of the field. Methods Inf Med., 40(4): 346-58

Medicinenet. Last access: 11/07/17. http://www.medicinenet.com/script/main/art.asp?articlekey=16836.

Microarray [Wikipedia]. Last access: 20/07/17. https://en.wikipedia.org/wiki/Microarray_databases.

Habib N, Ahmed K, Jabin I, Rahman MM. 2016. Application of R to investigate common gene regulatory network pathway among bipolar disorder and associate diseases. Network Biology, 6(4): 86-100

Habib N, Ahmed K., Jabin I., Rahman MM. 2017. Drug design and analysis for bipolar disorder and associated diseases: A bioinformatics approach. Network Biology, 7(2): 41-56

Ncbi. Last access: 23/07/17. https://www.ncbi.nlm.nih.gov/gds/.

PCA [Wikipedia]. Last access: 20/07/17. https://en.wikipedia.org/wiki/Principal_component_analysis

Principal_component_analysis. Last access: 20/07/17. http://www.fon.hum.uva.nl/praat/manual/Principal_component_analysis.html

PROGRAMIZ. Last access: 20/07/17. https://www.programiz.com/r-programming.

R[Home]. Last access: 20/07/17. https://www.r-project.org/about.html.

Raychaudhuri S, et al. 2000. Principal components analysis to summarize microarray experiments: application to sporulation time series. Pac Symp Biocomput, 455-466.

Saravanakumar S, Natarajan J. 2011. Microarray Data Analysis and Mining Tools. Bioinformation, 6(3): 95-99

Sebastian R. 2014. Last access: 20/07/17. http://sebastianraschka.com/Articles/2014_python_lda.html.

TechTarget. Last access: 20/07/17. http://searchbusinessanalytics.techtarget.com/definition/R-programming-language.

Wolfram S, Redestig H, Scholz M, Walther D, Selbig J. 2007. PCA methods-a bioconductor package providing PCA methods for incomplete data. Bioinformatics, 23 (9): 1164-1167

Zhang WJ, Feng YT. 2017. Metabolic pathway of non-alcoholic fatty liver disease: Network properties and robustness. Network Pharmacology, 2(1): 1-12