PATIENT CLASSIFICATION USING DEEP LEARNING

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Science and Engineering

by

Sangam Shrestha

Dr. Tin Nguyen, Thesis Advisor

December, 2019
We recommend that the thesis prepared under our supervision by

SANGAM SHRESTHA

entitled

PATIENT CLASSIFICATION USING DEEP LEARNING

be accepted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Tin Nguyen, Ph.D. – Advisor

Frederick C. Harris, Jr., Ph.D. – Committee Member

Binod Guragai, Ph.D. – Graduate School Representative

David Zeh, Ph.D. – Dean, Graduate School

December, 2019
ABSTRACT

With diseases like Alzheimer’s and Influenza still claiming lives, there have been a lot of methods developed in order to combat these diseases. There is a possibility that the key to finding susceptibility towards a disease might lie in the patient’s genetic makeup. The purpose of this thesis is to see if it is possible to predict whether a person is likely to suffer from a certain disease based on gene expression values. In order to achieve this goal, a computational based approach was adopted. Currently, artificial intelligence is producing results that were deemed not possible a few years ago. Moreover, deep learning, one specific branch of artificial intelligence, has been used to produce useful results. It has been used in many new technologies such as self-driving cars, natural language processing, and many other automated systems. This research came up with a method that makes use of a deep learning approach and found that it is indeed effective in classifying patients.
ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor Dr. Tin Nguyen of the Computer Science and Engineering department at the University of Nevada, Reno. He consistently allowed this paper to be my own work and always guided me in the right direction whenever I needed it. I would like to thank the committee members who were involved in the validation for this thesis: Dr. Frederick C. Harris, Jr. and Dr. Binod Guragai. Without their valuable support and guidance, thesis could not have been successfully conducted. I would also like to thank my fellow lab mates: Hung Nguyen, Duc Tran and Bang Tran as the readers of this thesis, and I am really indebted to them for their very valuable comments on this research. Finally, I would like to thank my parents for providing me with support and encouragement throughout my years of study and assisting me in reaching my goals. This accomplishment would not have been possible without them. Thank you.

Author

Sangam Shrestha
# TABLE OF CONTENTS

Abstract
Acknowledgements
Table of Contents
List of Figures

1 Introduction
   1.1 Overview
   1.2 Contributions
   1.3 Outline

2 Background and Related Work
   2.1 Overview
   2.2 Similar works in the field

3 Methods
   3.1 Patient Classification Methodology
   3.2 Data Preprocessing
   3.3 Data Normalization
   3.4 Selection of the statistically significant genes
   3.5 Data Visualization using PCA (Alzheimer's)
   3.6 Data Visualization using PCA (Influenza)
   3.7 Training the model

4 Evaluation
   4.1 Quantitative Results
   4.2 Other machine learning classifiers
   4.3 Results for Alzheimer’s datasets
   4.4 Results for Influenza datasets
   4.5 Analysis of the results

5 Discussion and Conclusion
   5.1 Discussion
   5.2 Summary
5.3 Limitations ................................................................. 52
5.4 Future Work ............................................................. 53

Bibliography .......................... 55
LIST OF FIGURES

1.1 Central Dogma is the flow of information from DNA in order to form protein .......................................................... 2

1.2 Gene Expression Matrix where the rows are the genes and the column are the samples. The samples can be different individuals or tissues. ................................................................. 3

3.1 Data preprocessing is done by taking all the common genes from the datasets and data normalization converts all the gene expression values between 0 and 1. .................................................. 13

3.2 Seven datasets from Alzheimer’s and nine data sets from Influenza along with their corresponding dimensions. The samples are further divided into controls and diseases with their counts ............... 14

3.3 Histogram showing the distribution of data in GSE42026 .............. 15

3.4 Histogram showing the distribution of data in GSE29366 ............... 16

3.5 Histogram showing the distribution of data in GSE4757 ................. 17

3.6 Histogram showing the distribution of data in GSE5281 ................. 17

3.7 Common genes from the datasets ............................................. 18

3.8 Common genes from the Alzheimer’s datasets .............................. 18

3.9 Common genes from the Influenza datasets ................................. 19

3.10 Data Normalization ............................................................... 19

3.11 Equation used to normalize the data ........................................ 20

3.12 The gene expression values are normalized between 0 and 1 ............ 20

3.13 The figure in the left is the gene expression matrix where the rows are the genes and the columns are the samples which have been labeled as control or disease. The image on the right is the application of a t-test between the control and the disease, which provides genes along with their p-values. ............................................... 22

3.14 Threshold is applied to the genes and only those genes with p-values less than 0.05 are selected as significant genes. ......................... 23

3.15 Conversion of Alzheimer’s dataset GSE48350 into two component PCA ........................................................................... 24
3.16 Conversion of Alzheimer’s dataset GSE48350 into three component PCA ............................................. 25
3.17 Conversion of Alzheimer’s dataset GSE4757 into three component PCA ............................................. 26
3.18 Conversion of Alzheimer’s dataset GSE36980 into three component PCA ............................................. 27
3.19 Conversion of Influenza dataset GSE40012 into three component PCA 28
3.20 Conversion of Influenza dataset GSE82050 into three component PCA 29
3.21 Conversion of Influenza dataset GSE71766 into three component PCA 30
3.22 Conversion of Influenza dataset GSE34205 into two component PCA 31
3.23 Conversion of Influenza dataset GSE34205 into three component PCA 32
3.24 Gene expression matrix as input for the multilayer perceptron classifier .................................................. 33
3.25 Multilayer Perceptron classifier that takes gene expression values as input and trains based on the class label .................................................. 34

4.1 The trained model is used for classification on the test dataset . . . . . . 36
4.2 A confusion matrix is created to judge how well the model performed. 37
4.3 Classification report is generated to assess the model. . . . . . . . . . 38
4.4 Figure showing the ROC and AUC for Alzheimer’s dataset GSE48350 41
4.5 Figure showing the ROC and AUC for Alzheimer’s dataset GSE5281 42
4.6 Figure showing the ROC and AUC for Alzheimer’s dataset GSE16759 42
4.7 Figure showing the ROC and AUC for Alzheimer’s dataset GSE39420 43
4.8 Figure showing the ROC and AUC for Alzheimer’s dataset GSE48350 44
4.9 Figure showing the ROC and AUC for the Influenza dataset GSE17156 45
4.10 Figure showing the ROC and AUC for the Influenza dataset GSE21802 45
4.11 Figure showing the ROC and AUC for the Influenza dataset GSE27131 46
4.12 Figure showing the ROC and AUC for the Influenza dataset GSE29366 46
4.13 Figure showing the ROC and AUC for the Influenza dataset GSE34205 47
4.14 Figure showing the ROC and AUC for the Influenza dataset GSE42026 48
4.15 Figure showing the AUC values for both Alzheimer’s and Influenza datasets. The bolded values are the highest AUC values. . . . . . . . . . . . . . . . . . . 49
4.16 Box plot comparing the AUC values for both Alzheimer’s and Influenza datasets. The figure shows the combined results for Alzheimer’s and Influenza. ................................. 49
4.17 Results with the mean and median AUC values for both Alzheimer’s and Influenza datasets. ................................. 50
CHAPTER 1

INTRODUCTION

1.1 Overview

Alzheimer’s and Influenza are still major threats to humanity as they keep claiming a lot of lives. Continuous research is going on in order to prevent further losses from the diseases. One thing that might provide key insights into the nature of these diseases is the genetic makeup of a patient. It might be possible to predict the susceptibility of a person to a certain disease based on their gene expression values. In order to achieve that goal, deep learning, one of the most popular approaches at the moment, was used in this research. With the recent successes in self-driving cars, natural language processing and many other automated systems, it seemed fitting to make use of the approach for the purpose of classifying patients. If deep learning is so effective in all these fields, it should also be helpful in classifying patients as disease or control. This is the inspiration for this thesis.

There are approximately 20,000 genes in the human body, few of which are responsible for causing a disease. Gene Expression is the process by which instructions in our DNA are converted into functional products such as protein. Gene Expression values are the expression level of the particular gene in the particular sample. Transcription is the process in which the DNA sequence of a gene is copied to make a messenger RNA (mRNA) molecule. Translation is the process in which the sequence of the mRNA is decoded to specify the amino acid sequence of a polypeptide. Folding is the process in which the amino acid chain (polypeptide) results in distinct three-dimensional conformations to form a functional complex called protein. This whole flow of information, as can be seen in Figure 1.1, from DNA in order to form protein is known as the Central Dogma.
1.2 Contributions

With the objective of finding if it is possible to predict whether a person is likely to suffer from a certain disease based on the gene expression values, as can be seen in Figure 1.2, seven datasets from Alzheimer’s (GSE1297, GSE16759, GSE4757, GSE39420, GSE5281, GSE48350, GSE36980) and 9 datasets from Influenza (GSE17156, GSE21802, GSE29366, GSE34205, GSE40012, GSE42026, GSE71766, GSE82050, GSE27131) were chosen from the Gene Expression Omnibus (GEO). Datasets were first preprocessed by taking common genes from all the dataset for a particular disease. After the data preprocessing, normalization was done to the data followed by performing a t-test based on the control and disease. Only statistically significant genes were used to train the deep learning model to classify if a person is going to suffer from a certain disease. The model was compared with six other machine learning classifiers. Results on the datasets shows that the developed model outperformed the other models.
The thesis has been divided into five chapters. The Chapter 1 is introduction, the Chapter 2 is background and related work, the Chapter 3 is methods, the Chapter 4 is evaluation and the Chapter 5 is discussion and conclusion. The basic flow of thesis starts with the introduction (Chapter 1) where the classification concept for diseases is discussed. It is followed by background and related work (Chapter 2). In this chapter, past works in the field and the related works are discussed along with their shortcomings. In the methods (Chapter 3), a brief overview is given followed by an explanation on data preprocessing, data normalization, selection of the statistically significant genes, data visualization using PCA (Alzheimer’s),
data visualization using PCA (Influenza) and finally a way to train the model. In the evaluation (Chapter 4), quantitative results are presented. The summary of the results is presented along with some description about other machine learning classifiers. Finally, in the discussion and conclusion (Chapter 5), the summary of the whole thesis is presented along with its current limitations. Future work is proposed to build upon thesis.
CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Overview

In many cases, it so happens that people only visit medical centers only after they suffer from a disease or there is something wrong with them. Would it not be better if it was possible to predict if a person is likely to suffer from a certain disease beforehand so that preventive measures could be taken? A lot of work has been done in the past in this field which makes use of various kinds of data. Some use image scans [21, 31], voice recordings [15] or other data like heart rate [19] to make predictions of a disease. Although all of these are good approaches, for the purpose of this thesis, gene expression values will be used along with deep learning to make disease prediction.

Gene expression is the process by which genetic information is used to make gene products. Gene expression provides a lot of information, such as if a person is prone to diseases or infections. In order to make use of this information, deep learning has been used. With the advancement in computational resources and algorithms [35, 36], it seems like a good fit to adopt this approach.

Although, the method could work for many diseases, for the purpose of this thesis, it has been tested on Alzheimer’s and Influenza datasets. Alzheimer’s has seven datasets (GSE1297, GSE16759, GSE4757, GSE39420, GSE5281, GSE48350, GSE36980) and Influenza has nine datasets (GSE17156, GSE21802, GSE29366, GSE34205, GSE40012, GSE42026, GSE71766, GSE82050, GSE27131). The results from the method are compared with other machine learning approaches like Bernoulli Naive Bayes (BNB), Random Forest (RF), Decision Tree (DT), Linear Discriminant Analysis (LDA), Gradient Boosting (GB) and Adaptive Boosting (Ad-
2.2 Similar works in the field

Some of the similar works [31] make use of image scans to make patient classifications. The image scans are converted into numerical data which are then processed [20, 29, 40, 47] and fed to the model for processing. The model consists of Random Forest tree ensemble classifier which is based on the voting system and makes use of the bagging process to detect and predict Alzheimer’s disease. Bagging in machine learning refers to a process where chunks of data are analyzed in a parallel way.

Likewise, [21] makes use of MRI scans to predict stroke delineation. The MRI scan is converted into image data, normalization is done and feature maps are created. The data is then fed to the Gaussian Naive Bayes classifier [7, 18, 26, 27, 46]. This classifier is used to get the probability where the data is generally continuous and has Gaussian distribution, or commonly called Normal distribution.

Similarly, in [19], heart rate values are used as data which are fed to the discrete wavelet transform. Generally discrete wavelet transform is used to analyze signals and also has applications in speech recognition. The resulting data has a huge amount of features. So, dimensionality reduction is done with the help of principal component analysis (PCA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA) [12, 24, 33, 50]. A t-test is performed on the remaining data and different classification techniques like Linear Discriminant Analysis (LDA), Gaussian Mixture Model (GMM), Probabilistic Neural Network (PNN) and K-Nearest Neighbor (KNN) are applied to separate if the patient is normal or is suffering from Coronary Artery Disease (CAD).

Furthermore, [15] provides a way to diagnose Parkinson’s disease. Since, it is a
disease which cannot be diagnosed through standardized blood tests [8, 16, 44, 52],
datasets containing voice recordings of Parkinson’s patients and healthy patients
are used. In order to train the model, Boosted Decision Tree (BST) is used. It is an
ensemble model made from gradient boosted regression trees.

A lot of work has been done in the field of classification of a person as healthy
or disease. Some of them include works, like [1] where a hierarchical approach
is used, which makes use of the kernel matrix to update the dictionary atoms
only once. The new thing about this method [25, 28, 48, 51] is that the dictionary
is learned in a linearly transformed/coefficient space involving sparse matrices,
rather than the kernel space. The reason why it prevails over other new methods
is that the proposed method has much less computational complexity and does
really well with other pattern classification tasks.

In addition, [13] is a method which achieved better results than the other five
models based on particle swarm optimization [2, 3, 20, 38], Genetic algorithms,
fruit fly optimization, and firefly algorithm. On top of that it also did better than
the machine learning methods including Linear Discriminant Analysis (LDA), LDA
with local learning-based feature selection, and kernel extreme learning machine
in a 10-fold cross-validation scheme. The method is very effective for making the
diagnosis of a disease.

Likewise, in [37], the feature dimension of three datasets is reduced using cor-
relation based feature selection (CFS) algorithm. It is done so by performing clas-
sification performances of 30 machine learning algorithms on the three datasets.
Ensembles of the classifiers are constructed based on RF algorithm to judge the
performance of the corresponding classifiers. Leave-one-out validation is used
to carry out these experiments. To analyze the performance, classification accu-
riacy (ACC), kappa error (KE) and area under the receiver operating characteristic
(ROC) curve (AUC) are used.

Also, in [5], PBL-McRBFN classifier is used to predict PD using micro-array gene expression data obtained from the ParkDB database. Its performance has been evaluated using Independent Component Analysis (ICA). It has also been compared to other methods using a one-way repeated ANOVA test. It also does PD prediction [30, 34, 43, 49] using the standard vocal and gait PD data sets. While comparing PBL-McRBFN with other approaches, it exhibits better results.

Moreover, in [23], the proposed method makes use of a hierarchical clustering algorithm by an average linkage method. It is used to implement model-based expression calculations, two-group comparison, and clustering is available on request. Furthermore, in [4], Meta-cognitive Radial Basis Function Network (McRBFN) method is proposed. McRBFN is inspired by human meta-cognitive learning principles. McRBFN has two components, namely the cognitive component and the meta-cognitive component [17, 32, 42]. The cognitive component computes the optimal output weights with the least computational effort by finding analytical minima of the nonlinear energy function. The meta-cognitive component controls the learning process which chooses the best learning strategy for the current sample. In addition, sample overlapping conditions are considered to minimize the wrong classification.

Other works in this field include [9, 10, 11, 14], which deal with classification of genes and include some the work done in the past in this field. There are a lot of inconsistencies among the lab when processing and collecting data. If we look at the history of artificial intelligence, it has been evolving since the 1950’s. There was a lot of expectations as progress was made. The first emergence of robots and software happened during this stage. However, during the mid 1970’s, without any significant achievement, the first AI winter happened. This led to a major drop
in the funding for AI. Again, in the early the 1980’s, with the breakthrough of the neural network, artificial intelligence started gaining momentum again. During the end of 1980’s, there was still no practical applications of artificial intelligence which led to the second AI winter. In the recent years, with the development of big data and deep learning along with the development in processing power [6, 22, 41], there has been a lot of progress in the field of neural network with the deep learning capability. Some of the top applications include: Apple Siri, Google driver less car, Alpha Go and IBM Watson. Deep learning is very effective in natural language processing, image recognition, cybersecurity and healthcare fields. Also, one of the important fields, it is being used is robotics. Many astounding use, like monitoring a huge amount of agricultural fields, making use of drones to deliver packages like Amazon is doing, venturing into visually degraded environment where it might not be possible for humans and collection of data using rovers in space without any human intervention, have only fuelled towards more innovative approaches to make further advancement using deep learning. There are still a lot of fields, where there are tremendous possibilities that deep learning can be used. Since, it is still in its early phase, researchers suggest that it might be used to do repetitive jobs. One of the key reasons why deep learning is performing so well is the huge amount of data that keep constantly being generated. With the advancement in technology, it is now possible to store and process huge amount of data in a short duration of time.

Likewise, another interesting field of research is finding biomarkers for a disease. Biomarkers could be anything that might indicate a disease. Recently, it is popular in the field of gerontology. Researches are being done in order to find the set of genes which lead to the old age. For instance, in [39], an ensemble of twenty one deep neural networks with varying architecture is created to predict a
person’s age based on the blood test. In order to train the deep neural networks, around 60,000 samples were used and the results were encouraging. The method was able to find five different markers that were significant in predicting human age. In a similar way, [45] shows how deep learning could be used to detect aging related neurodegenerative diseases. One of the key challenges in detecting the diseases like dementia is that there is a lot of variability involved among the patients. It proposes an artificial intelligence based monitoring of the patients in order to make early detection of the diseases. Early detection is very important in diseases like dementia, as treatment in later stages might not be that successful. The inspiration for this thesis comes from the fact that deep learning is making such a huge difference in the world right now. Although, other methods have been used in the recent years [1, 13, 37] there is inconsistency in the way results are generated. Deep learning can handle a huge amount of data. With the addition of hidden layers, it can easily perform non-linear classifications and the whole system could be scaled in such a way that it could be made generalizable.

Since deep learning seems to be doing well in all the areas, it might also be used to predict the susceptibility of a person towards a disease. One of the biggest impediment in applying deep learning in the medical field is that the number of samples might be too low and standard of the data might be good enough to train different classifiers. Since different labs have different ways of collecting and processing the data, uniformity might be one of the key things lacking which might result into inconsistent results among different labs. In order to overcome it, data preprocessing becomes vital to get consistent and reproducible results. There are a different ways to proceed once we have processed the data.

Generally, the datasets are divided into training and testing sets. The datasets are preprocessed and normalized and some methods like t-test is used to look for
only statistically significant data. A statistic like t-test allows the method to see if there is significant difference between the means of the compared samples. Training datasets are processed and fed as input to a classifier and a way to evaluate the developed model is taken. The trained model is fed with the unseen test sets and the results are evaluated using some metrics. The results using the method developed is compared with other similar methods found in the literature. Although, there are different ways to make predictions based on different inputs, one of the fundamental aspect is to make the method inexpensive and available to everyone.
CHAPTER 3

METHODS

3.1 Patient Classification Methodology

There are approximately 20,000 genes in human body. At the moment, big biological data is an important research topic. Gene expression datasets are high-dimensional big datasets because they contain tens of thousands of genes/features with very few patients/samples. Since the number of features is much greater than the number of samples/patients, this is known as the curse of dimensionality. Thus analyzing of these types of datasets has become complicated and challenging for researchers. The objective is to be able to classify the samples/patients into certain categories as control or disease. It is important to have correct classification in order to provide proper treatment and therapies. The genes have gene expression values. Based on these gene expression values, it should be possible to predict whether a person will suffer from a certain disease or is likely to suffer from a certain disease in the future. As can be seen in the Figure 3.1, the proposed methodology takes the seven datasets from Alzheimer’s and nine datasets from Influenza, and takes the common genes among the datasets respectively. It then goes on to normalize the data and then only considers those genes which are statistically significant. In order to get the statistically significant genes, a t-test is performed between control samples and disease samples. The data is then fed to the multilayer perceptron (MLP) classifier for the training.
3.2 Data Preprocessing

To get better results from the data, it is preferable to process the data. Data preprocessing is an important step in Machine Learning. The quality of data directly affects the model’s ability to learn. Thus it is vital that the data is preprocessed before feeding it to the model. For achieving better results from the applied model in Machine Learning, the data has to be formatted in a certain way. There are seven data sets for Alzheimer’s: GSE1297, GSE16759, GSE4757, GSE39420, GSE5281, GSE48350, GSE36980 and nine for Influenza: GSE17156, GSE21802, GSE29366, GSE34205, GSE40012, GSE42026, GSE71766, GSE82050, GSE27131. As seen in Figure 3.2, the number of the genes in GSE1297 is 12,496 and the number of samples is 31 of which 9 are controls and 22 are diseases. The number of the genes in GSE16759 is 19,851 and the number of samples is 8 of which 4 are controls and 4 are diseases. The number of the genes in GSE4757 is 19,851 and the number of samples is 20 of which 10 are controls and 10 are diseases. For GSE39420, the number of the genes in is 19,917 and the number of samples is 21 of which 7 are controls and 14 are diseases. The number of the genes in GSE5281 is 19,851 and the number of samples is 161 of which 74 are controls and 87 are diseases. The number of the genes in GSE48350 is 19,851 and the number of samples is 253 of which 173 are
controls and 80 are diseases. Finally, the number of the genes in GSE36980 is 19,879 and the number of samples is 79 of which 47 are controls and 32 are diseases.

<table>
<thead>
<tr>
<th>Disease</th>
<th>Dataset</th>
<th>Dimension</th>
<th>no. of Controls</th>
<th>no. of Diseases</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alzheimer's</td>
<td>GSE1297</td>
<td>12496 x 31</td>
<td>9</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>GSE16759</td>
<td>19851 x 8</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>GSE4707</td>
<td>19851 x 20</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>GSE38420</td>
<td>19917 x 21</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>GSE5281</td>
<td>19851 x 161</td>
<td>74</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>GSE48350</td>
<td>19851 x 253</td>
<td>173</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>GSE36980</td>
<td>19678 x 79</td>
<td>47</td>
<td>32</td>
</tr>
<tr>
<td>Influenza</td>
<td>GSE17156</td>
<td>12595 x 34</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>GSE21802</td>
<td>19018 x 40</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>GSE29366</td>
<td>19292 x 31</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>GSE34205</td>
<td>19851 x 50</td>
<td>22</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>GSE40012</td>
<td>19278 x 75</td>
<td>36</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>GSE42026</td>
<td>19292 x 52</td>
<td>33</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>GSE71766</td>
<td>18671 x 96</td>
<td>51</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>GSE82050</td>
<td>9804 x 39</td>
<td>15</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>GSE27131</td>
<td>19678 x 14</td>
<td>7</td>
<td>7</td>
</tr>
</tbody>
</table>

Figure 3.2: Seven datasets from Alzheimer’s and nine data sets from Influenza along with their corresponding dimensions. The samples are further divided into controls and diseases with their counts.

Similarly, in case of Influenza, the number of the genes in GSE17156 is 12,595 and the number of samples is 34 of which 17 are controls and 17 are diseases. The number of the genes in GSE21802 is 19,851 and the number of samples is 40 of which 4 are controls and 36 are diseases. The number of the genes in GSE29366 is 19,292 and the number of samples is 31 of which 12 are controls and 19 are diseases. The number of the genes in GSE34205 is 19,851 and the number of samples is 50 of which 22 are controls and 28 are diseases. The number of the genes in GSE40012 is 19,278 and the number of samples is 75 of which 36 are controls and 39 are diseases. The number of the genes in GSE42026 is 19,292 and the number of samples is 52 of which 33 are controls and 19 are diseases. The number of the genes in GSE71766 is 18,671 and the number of samples is 96 of which 51 are controls and 45 are diseases. The number of the genes in GSE82050 is 9,804 and the number of samples is 39 of which 15 are controls and 24 are diseases. Finally, the number of the genes in GSE27131 is 19,878 and the number of samples is 14 of which 7 are controls and 7
are diseases.

The datasets contain the gene expression values. When expressed as a matrix, the row contains the genes and the column contains the samples. Figure 3.3 and Figure 3.4 are the histogram representation of the Influenza datasets. Generally, if we represent some natural phenomenon, it seems to have Gaussian or Normal distribution. However, both the figures seems to have right-skewed unimodal distribution.

Again, the Figure 3.5 and Figure 3.6 are the histogram representation of the Alzheimer’s datasets. Both of them seems to have a right-skewed unimodal distribution. This could be showing that when the samples have disease, they seem to be giving abnormal distribution of data which is a good thing as it might help in better classification of the samples.

![Figure 3.3: Histogram showing the distribution of data in GSE42026](image)

Since, there seems to be some kind of pattern in the datasets, common genes from the datasets belonging to both the datasets are taken respectively as can be seen in Figure 3.7, Figure 3.8 and Figure 3.9. In the Figure 3.8, it can be seen that
Figure 3.4: Histogram showing the distribution of data in GSE29366

after taking common from the seven datasets of Alzheimer’s (GSE1297, GSE16759, GSE4757, GSE39420, GSE5281, GSE48350, GSE36980), there were total of 12,211 genes. The datasets used for training are GSE4757 and GSE36980. Likewise, in the Figure 3.9, after taking common from the nine datasets of Influenza (GSE17156, GSE21802, GSE29366, GSE34205, GSE40012, GSE42026, GSE71766, GSE82050, GSE27131), there were 5,499 genes. The datasets used for training are GSE40012, GSE71766 and GSE82050. It seems that there is a huge difference between the number of common genes between Alzheimer’s. The difference is mainly due to the number of genes present in GSE82050 dataset. As opposed to the other datasets, it only contains 9,804 genes. When taking common with the other Influenza datasets, there were only 5,499 common genes. However, By taking common genes, in both
cases, thousands of genes were removed. The reason for taking common genes is simple. In both kinds of datasets, there are samples which were disease which means there will be the genes responsible for causing the disease as well. It is
better to narrow down the genes in order to train the model.

![Gene Expression Matrix]

**Figure 3.7:** Common genes from the datasets

<table>
<thead>
<tr>
<th>Disease</th>
<th>Dataset</th>
<th>Dimension (no. of genes x no. of samples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alzheimer’s</td>
<td>GSE1297</td>
<td>12496 x 31</td>
</tr>
<tr>
<td></td>
<td>GSE16759</td>
<td>19851 x 8</td>
</tr>
<tr>
<td></td>
<td><strong>GSE4757</strong></td>
<td>19851 x 20</td>
</tr>
<tr>
<td></td>
<td>GSE39420</td>
<td>19917 x 21</td>
</tr>
<tr>
<td></td>
<td>GSE5281</td>
<td>19851 x 161</td>
</tr>
<tr>
<td></td>
<td>GSE48350</td>
<td>19851 x 253</td>
</tr>
<tr>
<td></td>
<td><strong>GSE36980</strong></td>
<td>19878 x 79</td>
</tr>
</tbody>
</table>

*Number of Common Genes: 12211*

**Figure 3.8:** Common genes from the Alzheimer’s datasets

### 3.3 Data Normalization

As can be seen in the Figure 3.10, for all the datasets for both Alzheimer’s and Influenza, gene expression values are collected for the sample. However, there might be a lot of variability in how the data was taken. The range of the values
Figure 3.9: Common genes from the Influenza datasets might be different based on different conditions. It would not make any sense if the model is fed with data from different ranges. In order for the data to be comparable sample-wise, normalization is done. As seen in Figure 3.11, the equation for normalization is quite simple. For each sample, the gene expression value subtracts the minimum value in the sample, which is divided by the value, which is the result of subtraction of minimum value by the maximum value in the sample.

<table>
<thead>
<tr>
<th>Disease</th>
<th>Dataset</th>
<th>Dimension (no. of genes x no. of samples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Influenza</td>
<td>GSE17156</td>
<td>12595 x 34</td>
</tr>
<tr>
<td></td>
<td>GSE21802</td>
<td>19018 x 40</td>
</tr>
<tr>
<td></td>
<td>GSE29366</td>
<td>19292 x 31</td>
</tr>
<tr>
<td></td>
<td>GSE34205</td>
<td>19851 x 50</td>
</tr>
<tr>
<td></td>
<td>GSE40012</td>
<td>19278 x 75</td>
</tr>
<tr>
<td></td>
<td>GSE42026</td>
<td>19292 x 52</td>
</tr>
<tr>
<td></td>
<td>GSE71766</td>
<td>18671 x 96</td>
</tr>
<tr>
<td></td>
<td>GSE82050</td>
<td>9804 x 39</td>
</tr>
<tr>
<td></td>
<td>GSE27131</td>
<td>19878 x 14</td>
</tr>
</tbody>
</table>

Number of Common Genes: 5499

Figure 3.10: Data Normalization

For instance, as seen in Figure 3.12, for sample1, the values 4.5645339, 1.2672205,
Figure 3.11: Equation used to normalize the data

\[
\frac{x - \min(x)}{\max(x) - \min(x)}
\]

Figure 3.12: The gene expression values are normalized between 0 and 1

<table>
<thead>
<tr>
<th>Gene</th>
<th>Sample1</th>
<th>Sample 2</th>
<th>Sample 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(G_A)</td>
<td>4.5645339</td>
<td>4.5458962</td>
<td>4.2584037</td>
</tr>
<tr>
<td>(G_B)</td>
<td>1.2672205</td>
<td>1.0572446</td>
<td>1.5419328</td>
</tr>
<tr>
<td>(G_C)</td>
<td>5.8832878</td>
<td>5.8789360</td>
<td>5.8244021</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gene</th>
<th>Sample1</th>
<th>Sample 2</th>
<th>Sample 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(G_A)</td>
<td>0.3205802</td>
<td>0.3270111</td>
<td>0.3013011</td>
</tr>
<tr>
<td>(G_B)</td>
<td>0.0608709</td>
<td>0.0516393</td>
<td>0.0881175</td>
</tr>
<tr>
<td>(G_C)</td>
<td>0.4244504</td>
<td>0.4322327</td>
<td>0.4241978</td>
</tr>
</tbody>
</table>

and 5.8832878 became 0.3205802, 0.0608709 and 0.4244504 respectively. Likewise, for sample2, the values, 4.5458962, 1.0572446 and 5.8789360 became 0.3270111, 0.0516393 and 0.4322327 respectively. Similarly, for sample3, the values 4.2584037,
1.5419328 and 5.8244021 became 0.3013011, 0.0881175 and 0.4241978 respectively. The important thing to note is that after the normalization, all the values will be in a range from 0 to 1.

3.4 Selection of the statistically significant genes

The selection of the statistically significant genes consists of two steps: t-test application and thresholding. As seen in Figure 3.13, the gene expression matrix is divided into two sets of samples: control and disease. The green color represents the control samples, whereas the red color represents the disease samples. A t-test is applied in order to find if they are statistically different. The way a t-test works is that, it presents a null hypothesis where it is assumed that both the control and disease samples are of the same type. Under this assumption we receive probability values for different genes, which are also called the p-values. If the probability is very high, then the null hypothesis is correct. It means that there is no difference between the control and the disease. However, if the p-values are really small, we can reject the null hypothesis. There is a statistical difference between control and disease, which means that control and disease are of two different types. Thus, genes with such differences in samples between control and disease are called statistically significant genes. Those are probably the genes which are responsible for causing a disease. The equation for the t-test is applied to get the p-values. The numerator consists of the difference between the sample means and the denominator consists of the square root of the summation between the divisions of sample variances divided by their respective sample sizes.

Applying the t-test shows statistically significant genes as can be seen in Figure 3.14. While doing the training, it was found that genes with p-values greater than or equal to 0.05 provides good results. Therefore a threshold with a p-value of
Figure 3.13: The figure in the left is the gene expression matrix where the rows are the genes and the columns are the samples which have been labeled as control or disease. The image on the right is the application of a t-test between the control and the disease, which provides genes along with their p-values.

0.05 is established. When the threshold is applied, it was seen that the number of significant genes was reduced from 12,211 in Alzheimer’s and 5,499 in Influenza datasets to approximately 500 genes. A new gene expression matrix is formed consisting only of the statistically significant genes as seen in Figure 3.14.

3.5 Data Visualization using PCA (Alzheimer's)

It is often important to visualize the data. For this purpose, we have chosen principal component analysis (PCA). As in Figure 3.15, the Alzheimer’s dataset GSE48350 has been transformed into two components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis. The red dot represents the control and the green dot represents the disease. Here the concept of explained variance is important as it tells us how much information (variance) can be attributed to each of the principal components. In this case, Principal
Figure 3.14: Threshold is applied to the genes and only those genes with p-values less than 0.05 are selected as significant genes.

Component 1 contains 0.36206867 and Principal Component 2 contains 0.08514009 variance. So in total, it provides us with 44.71 percent of the total information.

Similarly, in Figure 3.16, the Alzheimer’s dataset GSE48350 has been transformed into three components: Principal Component 1, Principal Component 2 and Principal Component 3. In this case, Principal Component 1 contains 0.36206867, Principal Component 2 contains 0.08514009 variance and Principal Component 3 contains 0.06597389 variance. So in total, it provides us with 51.30 percent of the total information. Looking at the visual representation of the dataset, it seems that a non-linear classifier has to be used in order to classify the dataset.

Again, in Figure 3.17, the Alzheimer’s dataset GSE4757 has been transformed into three components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis and Principal Component 3, which is in the z-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.38070516, Principal Component 2 contains
Figure 3.15: Conversion of Alzheimer’s dataset GSE48350 into two component PCA

0.11106034 and Principal Component 3 contains 0.09585666 variance. So in total, it provides us with 58.75 percent of the total information. Looking at the visual representation of the dataset, it seems that a non-linear classifier has to be used in order to classify the dataset.

Again, in Figure 3.18, the Alzheimer’s dataset GSE36980 has been transformed into three components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis and Principal Component 3, which is in the
Figure 3.16: Conversion of Alzheimer’s dataset GSE48350 into three component PCA

z-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.31065064, Principal Component 2 contains 0.09795239 and Principal Component 3 contains 0.07168921 variance. So in total, it provides us with 48.01 percent of the total information. Looking at the visual representation of the dataset, it seems that a non-linear classifier has to be used in order to classify the dataset.
3.6 Data Visualization using PCA (Influenza)

Furthermore, in Figure 3.19, the Influenza dataset GSE40012 has been transformed into three components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis and Principal Component 3, which is in the z-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.26679203, Principal Component 2 contains
Figure 3.18: Conversion of Alzheimer’s dataset GSE36980 into three component PCA

0.10843038 and Principal Component 3 contains 0.06951615 variance. So in total, it provides us with 44.46 percent of the total information. Looking at the visual representation of the dataset, it seems that a non linear classifier would explain the dataset better than linear classifier.

In a similar way, in Figure 3.20, the Influenza dataset GSE82050 has been transformed into three components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis and Principal Component 3, which is
Figure 3.19: Conversion of Influenza dataset GSE40012 into three component PCA

in the z-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.21366158, Principal Component 2 contains 0.10184416 and Principal Component 3 contains 0.0856327 variance. So in total, it provides us with 40.10 percent of the total information. Looking at the visual representation of the dataset, it seems that a non linear classifier would explain the dataset better than linear classifier.

In addition, in Figure 3.21, the Influenza dataset GSE71766 has been trans-
Figure 3.20: Conversion of Influenza dataset GSE82050 into three component PCA

formed into three components: Principal Component 1, which is in the x-axis and Principal Component 2, which is in the y-axis and Principal Component 3, which is in the z-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.21881909, Principal Component 2 contains 0.20598934 and Principal Component 3 contains 0.09383579 variance. So in total, it provides us with 51.85 percent of the total information. Looking at the visual representation of the dataset, it seems that a non linear classifier would ex-
Figure 3.21: Conversion of Influenza dataset GSE71766 into three component PCA

plain the dataset better than linear classifier.

Also, in Figure 3.22, Influenza dataset GSE34205 has been transformed into two components: Principal Component 1, which is in the x-axis and Principal Component 2 which is in the y-axis. The red dot represents control and the green dot represents disease. In this case, Principal Component 1 contains 0.19014093 and Principal Component 2 contains 0.1314444 variance. So in total, it provides us with 32.15 percent of the total information. Looking at the visual representation
Figure 3.22: Conversion of Influenza dataset GSE34205 into two component PCA

of the dataset, it seems that a non linear classifier would explain the dataset better
than linear classifier.

Again, in Figure 3.23, the Influenza dataset GSE34205 has been transformed
into three components: Principal Component 1, which is in the x-axis and Principal
Component 2, which is in the y-axis and Principal Component 3, which is in the
z-axis. The red dot represents control and the green dot represents disease. In this
case, Principal Component 1 contains 0.19014093, Principal Component 2 contains
0.1314444 and Principal Component 3 contains 0.10253286 variance. So in total,
Figure 3.23: Conversion of Influenza dataset GSE34205 into three component PCA

...it provides us with 42.40 percent of the total information. Looking at the visual representation of the dataset, it seems that a non linear classifier would explain the dataset better than linear classifier.
3.7 Training the model

After the data preprocessing, there are a limited number of genes. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used to map the datasets. When we train an AI using supervised learning, we give it an input and tell it the expected output. That training data has inputs and outputs. Unsupervised learning is the task of machine learning using datasets with no labels. As seen in Figure 3.24, before feeding the matrix to the model, the gene expression matrix has been transposed in order to fit the format of the model. There is also a column to store the label to specify if a sample is control or disease.

<table>
<thead>
<tr>
<th></th>
<th>G_A</th>
<th>G_B</th>
<th>G_C</th>
<th>...</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Control</td>
</tr>
<tr>
<td>S2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Disease</td>
</tr>
<tr>
<td>S3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Control</td>
</tr>
<tr>
<td>S4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Disease</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.24: Gene expression matrix as input for the multilayer perceptron classifier

Multilayer perceptron (MLP) is chosen as the classifier because it can perform
non-linear classification with high accuracy. As can be seen in Figure 3.25, the model expects the gene expression values as input. There are three hidden layers with 300 nodes, and a ReLU activation function. The reason it is called deep learning is because of the number of hidden layers. The maximum number of iterations used is 1000 and the weight optimizer used is called Adam, which is a stochastic gradient based optimizer. For a sample, gene expression values are fed to the network. As the label is known, the weights and the bias get updated with each iteration.

The gene expression matrix with significant genes and known labels is fed to the MLP as input. After the maximum number of iterations, there are final weights and bias. The whole point of training is to get the final weights and bias. While training the MLP (using supervised learning) with the gene expression matrix, the weights and bias of the model gets updated. Datasets GSE4757 and GSE36980
are used for training from Alzheimer’s and datasets GSE40012, GSE71766 and GSE82050 are used for training from Influenza.
CHAPTER 4

EVALUATION

4.1 Quantitative Results

After the training has been done in the gene expression matrix, the next step is to see how well the model will do with datasets that it has not seen. For the Alzheimer’s, the test datasets are GSE1297, GSE16759, GSE39420, GSE5281, GSE48350, and for Influenza the test datasets are GSE17156, GSE21802, GSE29366, GSE34205, GSE42026, GSE27131. As can be seen in the Figure 4.1, The whole point of training the model in multilayer perceptron is to update the weight and bias. Once it has been updated, it makes a prediction based on those values. However, there has to be a way to assess how accurate the model is.

Figure 4.1: The trained model is used for classification on the test dataset

A confusion matrix is good way to represent the results. As can be seen in the Figure 4.2, there are four terms in confusion matrix: true positive, false positive,
false negative and true negative. True positive are when the results are actually positive and were predicted as positive. True negatives are when the results are actually negative and were predicted as negative. False positive are when the results are actually negative but predicted as positive and false negatives are when the results are actually positive but were predicted as negative.

Based on the confusion matrix, a classification report is generated. In the Figure 4.3, the three terms calculated are precision, recall and F1 score. Precision is the true positive divided by the sum of the true positive and the false positive. Recall is the true positive divided by the sum of the true positive and the false negative and F1 score is the multiplication of precision and recall, which is divided by the sum of precision and recall and finally multiplied by 2. Generally researchers prefer precision over the other two to evaluate the results.
4.2 Other machine learning classifiers

Random forest (RF) is one of the most popular supervised learning models. It is an ensemble method, meaning that it makes use of voting to make a decision. It is also a bagging approach because it trains individual models in a parallel way. Each model is trained by a random chunk of the data. Generally the individual models are decision trees. The models make a certain prediction. The decision made by the majority of the models is the final decision of the whole random forest model. The results using this model are really good.

The other machine learning algorithm is adaptive boosting, also called Adaboost. Unlike the random forest model, it uses what is called boosting approach.
It is a way to train individual models in a sequential way. The models learn from mistakes made by the previous model. The reason it is called boosting because it takes all the weak classifiers and combines them into one strong classifier. It learns from previous mistakes by using the updated weights and bias in the next individual model.

Another popular machine learning algorithm is what is called the gradient boosting (GB). Like the name suggests, it is also a boosting approach. However, the main difference is that instead of passing the updated weights and bias to the next individual model, it passes the residual error by the model. The residual error is simply the difference between the actual value and the predicted value. There are many variants of gradient boosting.

Linear Discriminant Analysis (LDA) is a very popular machine learning algorithm. The LDA is a supervised learning algorithm that is mainly used for classification purposes and dimension reduction. The way it works is that it tries to reduce the dimension of the data in such a way so that there is maximum difference between the mean of the two classes.

Likewise, Decision Trees (DT) is a supervised learning method designed to learn from the features (genes) and makes prediction. It has been used both for regression and classification purposes. It makes use of binary trees to make a decision. It generally makes the classification decision by dividing the data into different regions.

Similarly, Bernoulli Naive Bayes is another interesting machine learning algorithm. If X is a random variable which is Bernoulli-distributed, it can assume only two values (say control and disease). The algorithm gives the probability for the value to be control or disease. If the probability that the value is control is p, then the probability that value is disease would be (1-p).
4.3 Results for Alzheimer’s datasets

After tuning the parameters, multilayer perceptron (MLP) was chosen to be the classifier for the model. However, to evaluate how well it is doing, it needs to be compared with other machine learning algorithms. For the purpose of this thesis, it has been compared with Bernoulli Naive Bayes (BNB), random forest (RF), Decision Trees classifier (DT), Linear Discriminant Analysis (LDA), gradient boosting (GB) and adaptive boosting (Adaboost). In order to compare receiver operating characteristic (ROC) curve, area under the curve (AUC) metrics were used. In the y-axis there is a true positive rate, and in the x-axis there is a false positive. There is something called cut off value that determines the curve. Say if a certain cut off probability is assigned, above the cut off is control and below the cut off is disease. The probability could change from 0 to 1, which results into the ROC curve and, the area under the ROC curve is given by the AUC value.

In the Figure 4.4 for the Alzheimer’s dataset GSE1297, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.74, 0.59, 0.71, 0.49, 0.64, 0.70 and 0.60 respectively. The maximum AUC value belongs to the multilayer perceptron classifier. The precision, recall and f1 score values for the multilayer perceptron classifier are 0.785, 0.733, and 0.746 respectively. Likewise, in the Figure 4.5 for the Alzheimer’s dataset GSE5281, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.73, 0.70, 0.69, 0.51, 0.43, 0.52 and 0.67 respectively. The maximum AUC value belongs to the multilayer perceptron classifier. The precision, recall and f1 score values for multilayer perceptron classifier are 0.657, 0.637, and 0.635 respectively. Similarly, in the Figure 4.6 for the Alzheimer’s dataset GSE16759, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.67, 0.67, 0.58, 0.67, 0.50, 0.33 and 0.58 respectively. The maximum AUC value belongs to the multilayer perceptron, Bernoulli Naive Bayes, Random Forest and Decision Tree
classifiers. The precision, recall and f1 score values for multilayer perceptron classifier are 0.829, 0.8, 0.806 respectively. Likewise, the precision, recall and f1 score values for Bernoulli Naive Bayes classifier are 0.714, 0.714, 0.714 respectively. Similarly, the precision, recall and f1 score values for Random Forest classifier are 0.714, 0.714, and 0.714 respectively. In addition, the precision, recall and f1 score values for Decision Tree classifier are 0.602, 0.666, 0.624 respectively.

In addition, in the Figure 4.7 for the Alzheimer’s dataset GSE39420, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.93, 0.93, 0.94, 0.57, 0.55, 0.83 and 0.85 respectively. The maximum AUC value belongs to the Random Forest classifier. The precision, recall and f1 score values for Random Forest classifier are 0.925, 0.9, 0.903 respectively. Furthermore, in the Figure 4.8 for the Alzheimer’s dataset GSE48350, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.73, 0.73, 0.73, 0.54, 0.41, 0.68 and 0.69 respectively. The maximum AUC value belongs to the multilayer perceptron, Bernoulli Naive Bayes and Random Forest.
Figure 4.5: Figure showing the ROC and AUC for Alzheimer’s dataset GSE5281

Figure 4.6: Figure showing the ROC and AUC for Alzheimer’s dataset GSE16759
classifiers. The precision, recall and f1 score values for multilayer perceptron classifier are 0.678, 0.666, 0.671 respectively. The precision, recall and f1 score values for Bernoulli Naive Bayes classifier are 0.670, 0.629, 0.638 respectively. In this way, for the five test datasets for Alzheimer’s, MLP classifier got the highest AUC score for four test datasets.

4.4 Results for Influenza datasets

Similar to the Alzheimer’s datasets, six Influenza datasets were chosen as test sets. In the Figure 4.9 for the Influenza dataset GSE17156, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.88, 0.76, 0.73, 0.38, 0.69, 0.61 and 0.63 respectively. The maximum AUC value belongs to the multilayer perceptron classifier. The precision, recall and f1 score values for multilayer perceptron classifier are 0.798, 0.787, and 0.786 respectively. Likewise, in the Figure 4.10 for the Influenza
dataset GSE21802, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.84, 0.84, 0.94, 0.49, 0.81, 0.89 and 0.71 respectively. The maximum AUC value belongs to the Random Forest classifier and the Bernoulli Naive Bayes classifier. The precision, recall and f1 score values for Bernoulli Naive Bayes classifier are 0.904, 0.743, and 0.800 respectively.

In a similar manner, in the Figure 4.11 for the Influenza dataset GSE27131, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.74, 0.21, 0.38, 0.57, 0.86, 0.88 and 0.57 respectively. The maximum AUC value belongs to the gradient boosting classifier. The precision, recall and f1 score values for gradient boosting classifier are 0.730, 0.733, and 0.716 respectively. Also, in the Figure 4.12 for the Influenza dataset GSE29366, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.90, 0.86, 0.93, 0.61, 0.79, 0.87 and 0.78 respectively. The maximum AUC value belongs to the Random Forest classifier. The precision, recall and f1 score values for Random Forest classifier are 0.789, 0.766, and 0.770 respectively.
Figure 4.9: Figure showing the ROC and AUC for the Influenza dataset GSE17156

Figure 4.10: Figure showing the ROC and AUC for the Influenza dataset GSE21802
Figure 4.11: Figure showing the ROC and AUC for the Influenza dataset GSE27131

Figure 4.12: Figure showing the ROC and AUC for the Influenza dataset GSE29366
Likewise, in the Figure 4.13 for the Influenza dataset GSE34205, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.85, 0.81, 0.83, 0.49, 0.67, 0.76 and 0.61 respectively. The maximum AUC value belongs to the multilayer perceptron classifier. The precision, recall and f1 score values for multilayer perceptro classifier are 0.726, 0.714, and 0.698 respectively.

Figure 4.13: Figure showing the ROC and AUC for the Influenza dataset GSE34205

Furthermore, in the Figure 4.14 for the Influenza dataset GSE42026, the AUC values for MLP, BNB, RF, DT, LDA, GB and Adaboost are 0.93, 0.89, 0.98, 0.74, 0.85, 0.90 and 0.89 respectively. The maximum AUC value belongs to the Random Forest classifier. The precision, recall and f1 score values for Random Forest classifier are 0.880, 0.823, and 0.825 respectively.
4.5 Analysis of the results

Figure 4.15 shows the AUC values for all the seven different classifiers: MLP, BNB, RF, DT, LDA, and Adaboost for all the test datasets for Alzheimer’s and Influenza. In the five test datasets for Alzheimer’s, it seems that the MLP classifier gets the highest AUC values in four of them.

Likewise, in the six test datasets for Influenza, the MLP classifier gets the highest AUC values in one of them. In total, out of 11 datasets for both Alzheimer’s and Influenza, MLP had the highest AUC values in 6 of them. Likewise, Figure 4.16 shows the box and whisker plot, which compares the AUC values for both Alzheimer’s and Influenza datasets. In order to achieve the overall results, both the results from Alzheimer’s and Influenza are combined. As can be seen, the median values for the MLP is higher than any other classifiers.

Finally, in Figure 4.17 shows all the classifiers along with their overall mean
Figure 4.15: Figure showing the AUC values for both Alzheimer’s and Influenza datasets. The bolded values are the highest AUC values.

<table>
<thead>
<tr>
<th>Disease</th>
<th>Datasets</th>
<th>MLP</th>
<th>BNB</th>
<th>RF</th>
<th>DT</th>
<th>LDA</th>
<th>GB</th>
<th>Adaboost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alzheimer’s</td>
<td>GSE1297</td>
<td>0.738</td>
<td>0.585</td>
<td>0.710</td>
<td>0.494</td>
<td>0.636</td>
<td>0.698</td>
<td>0.602</td>
</tr>
<tr>
<td></td>
<td>GSE16759</td>
<td>0.666</td>
<td>0.666</td>
<td>0.583</td>
<td>0.666</td>
<td>0.5</td>
<td>0.333</td>
<td>0.583</td>
</tr>
<tr>
<td></td>
<td>GSE39420</td>
<td>0.928</td>
<td>0.928</td>
<td>0.940</td>
<td>0.571</td>
<td>0.553</td>
<td>0.833</td>
<td>0.845</td>
</tr>
<tr>
<td></td>
<td>GSE5821</td>
<td>0.727</td>
<td>0.701</td>
<td>0.688</td>
<td>0.508</td>
<td>0.425</td>
<td>0.520</td>
<td>0.666</td>
</tr>
<tr>
<td></td>
<td>GSE48350</td>
<td>0.734</td>
<td>0.728</td>
<td>0.727</td>
<td>0.536</td>
<td>0.406</td>
<td>0.675</td>
<td>0.690</td>
</tr>
<tr>
<td>Influenza</td>
<td>GSE17156</td>
<td>0.875</td>
<td>0.757</td>
<td>0.727</td>
<td>0.382</td>
<td>0.694</td>
<td>0.606</td>
<td>0.632</td>
</tr>
<tr>
<td></td>
<td>GSE21802</td>
<td>0.842</td>
<td>0.842</td>
<td>0.944</td>
<td>0.486</td>
<td>0.814</td>
<td>0.888</td>
<td>0.712</td>
</tr>
<tr>
<td></td>
<td>GSE29366</td>
<td>0.899</td>
<td>0.856</td>
<td>0.928</td>
<td>0.610</td>
<td>0.794</td>
<td>0.870</td>
<td>0.775</td>
</tr>
<tr>
<td></td>
<td>GSE34205</td>
<td>0.850</td>
<td>0.807</td>
<td>0.828</td>
<td>0.488</td>
<td>0.668</td>
<td>0.760</td>
<td>0.607</td>
</tr>
<tr>
<td></td>
<td>GSE42026</td>
<td>0.925</td>
<td>0.886</td>
<td>0.975</td>
<td>0.739</td>
<td>0.847</td>
<td>0.896</td>
<td>0.886</td>
</tr>
<tr>
<td></td>
<td>GSE27131</td>
<td>0.738</td>
<td>0.214</td>
<td>0.380</td>
<td>0.571</td>
<td>0.857</td>
<td>0.880</td>
<td>0.571</td>
</tr>
</tbody>
</table>

Mean: 0.811 0.724 0.766 0.550 0.654 0.723 0.688

Figure 4.16: Box plot comparing the AUC values for both Alzheimer’s and Influenza datasets. The figure shows the combined results for Alzheimer’s and Influenza.

and median values for both Alzheimer’s and Influenza datasets. For all the test datasets, the MLP has a mean of 0.811 and median of 0.842. The BNB classifier has mean of 0.724 and median of 0.757. The RF classifier has a mean of 0.766 and median of 0.727. The DT classifier has a mean of 0.550 and median of 0.536. The
Figure 4.17: Results with the mean and median AUC values for both Alzheimer’s and Influenza datasets.

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Mean</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>0.811</td>
<td>0.842</td>
</tr>
<tr>
<td>BNB</td>
<td>0.724</td>
<td>0.757</td>
</tr>
<tr>
<td>RF</td>
<td>0.766</td>
<td>0.727</td>
</tr>
<tr>
<td>DT</td>
<td>0.550</td>
<td>0.536</td>
</tr>
<tr>
<td>LDA</td>
<td>0.654</td>
<td>0.668</td>
</tr>
<tr>
<td>GB</td>
<td>0.723</td>
<td>0.760</td>
</tr>
<tr>
<td>Adaboost</td>
<td>0.688</td>
<td>0.667</td>
</tr>
</tbody>
</table>

LDA classifier has a mean of 0.654 and median of 0.668. The GB classifier has a mean of 0.723 and median of 0.760. Finally, the Adaboost classifier has a mean of 0.688 and median of 0.667.
CHAPTER 5
DISCUSSION AND CONCLUSION

5.1 Discussion

As we can see in both the diseases, Alzheimer’s and Influenza, the MLP seems to do well for classifying the sample as healthy or disease. There were 7 data sets for Alzheimer’s: GSE1297, GSE39420, GSE4757, GSE16759, GSE48350, GSE5281 and GSE36980 and 9 data sets for Influenza: GSE17156, GSE21802, GSE29366, GSE34205, GSE40012, GSE42026, GSE71766, GSE82050, GSE27131. Compared to other classifiers like Random Forest, Bernoulli Naive Bayes, Decision Trees Classifier, Linear Discriminant Analysis, Gradient Boosting, and Adaboost, the proposed method performed well. The key to better performance was the data preprocessing step, which includes taking the common genes from all the data sets and normalizing them. Taking the common data sets removed a lot of genes that had little to do with the disease. Likewise, normalizing them made sure that the gene expression values in different data sets were comparable. When the data was fed to the multilayer perceptron after the preprocessing step, the results improved significantly.

5.2 Summary

Humans have approximately 20,000 genes that serve as a blueprint for the human body. In addition, genes can also be used to predict if a person is susceptible to a certain disease. This prediction can be done using gene expression values. Gene expression is used for the synthesis of a functional gene product. These products are mostly proteins or functional RNA. The input for the method is a gene expression matrix where the rows are the genes and the columns are the samples. Data
prepossessing is done on the matrix and the common genes from all 7 datasets in Alzheimer’s and 9 datasets in Influenza. It drastically reduces the number of the genes, which makes sense because these are probably the genes that are responsible for determining if the person is healthy or not. Since supervised training is performed first, already known sample labels are also combined into the matrix. There are two kinds of labels: control and disease. In order to make the labels numeric, control and disease are converted into 0 and 1 respectively. In the training step, the whole matrix is grouped as control or disease and a t-test is performed with the null hypothesis that there is no major difference between the control and disease. The p-value gives the probability of whether there is any difference between the control and disease. If the p-value is less than 0.05, it means that there is higher probability that the control and disease have major differences. In other words, it means that the control and disease are significantly different. The p-values for all the genes are calculated. A certain threshold of 0.05 is used, and any genes with p-values above that are removed, leaving only those genes that are statistically significant. By combining all of the results, a new matrix is formed, which is supplied to the deep learning model as input. In the deep learning model, there are three hidden layers and a random state is used, in order to make the data reproducible. Cross validation is used to get the confusion matrix and precision, recall and F1 score values. The results of the training is the updated values of the parameters, which could be used to classify diseases in the test datasets.

5.3 Limitations

Right now, the classification of a person as control or disease is done by making use of gene expression values. One of the important factors in deciding how well the model does is the number of samples used for training. The total number of
samples in Alzheimer’s datasets was 573 and the total number of samples in Influenza datasets was 431. It might not be enough to train the model properly. More data with more samples might be necessary to improve the results. As seen in the results for Alzheimer’s and Influenza, it has shown itself to be effective. However, the overall result could still be improved by integrating other information such as gene-gene interaction networks and pathway information. It has only been tested for two kinds of diseases. It works for Alzheimer’s and Influenza, but it may or may not work for the classification of other kinds of diseases.

5.4 Future Work

In this thesis, a method was proposed to find the susceptibility of a patient towards a disease. As mentioned earlier, currently the method only works for two diseases: Alzheimer’s and Influenza. The method has not been tested for other diseases. In order to check the robustness of the method, it should also be tested on other diseases to see how well it does. As data preprocessing step, the method takes common genes from all the datasets and also performs data normalization. Different ways of performing data preprocessing might be used in the future. Likewise, in the method, t-test is used to find the statistically significant genes and p-value threshold of 0.05 is used. Although, this might be a good way to find the genes which might be responsible for a disease, other approaches like different tests could be used. Currently only the deep learning model is used for classification. It might be possible to combine it with other methods to make it more powerful. An ensemble of different deep learning models or even different classifiers could be used to get even better results. In addition different architectures of deep learning model could also be used.

Furthermore, the input for the method right now is just gene expression values.
It might be possible to combine other forms of information like gene-gene interaction networks and pathways to make the method more robust while at the same time making it available and inexpensive for use. Currently the method classifies if a person is susceptible to a disease or not. It might have other applications as well. It could be further extended to find biomarkers for diseases, which might be used for targeting the genes responsible for causing the disease.
BIBLIOGRAPHY


