Republic of Iraq

Ministry of Higher Education and Scientific Research

University of Babylon

College of Information Technology

Software Department

Genetic Immune and Antibody Response Analysis Using Machine Learning Model for Various Vaccines

A Thesis

Submitted to the Council of the College of Information Technology for Postgraduate Studies of the University of Babylon in Partial Fulfillment of Requirements for the Degree of Master in Information Technology/Software

By

Nuha Husham Mohammed Abd-Al Rahman

Supervised by

Asst. Prof. Dr. Sura Zeki ALrashid

2023 A.C. **1445 A.H.**

بسم الله الرحمن الرحيم

يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا

تَعْمَلُونَ خَبِيرٌ

صدق الله العظيم

سورة المجادلة الآية ١١

Declaration

I as a result of this declare that this dissertation entitled "Genetic Immune and Antibody Response Analysis Using machine Learning Model for Various Vaccines", submitted to the University of Babylon in partial fulfilment of requirements for the degree of Master in Information Technology \ Software, has not been proposed as an exercise for a similar degree atany other University. I also certify that this work described here is entirely my own except for experts and summaries whose source is appropriately cited in the references.

Signature:Name:

Date: / / 2023

Supervisor Certification

I certify that this thesis is prepared under my supervision at the Departmentof Software / Collage of Information Technology / Babylon University, by *Nuha Husham* as a partial fulfillment of the requirements for thedegree of Master in Information Technology

Signature:

Supervisor Name: Dr.

Date: / / 2023

The Head of the Department Certification

In view of the available recommendations, I forward this thesis for debateby the examination committee.

Signature:

Asst. Prof. Ahmed Saleem Abbass

Head of Software Department Date: / / 2023

Certification of the Examination Committee

We hereby certify that we have studied the dissertation entitled (**Genetic Immune and Antibody Response Analysis Using machine Learning Model for Various Vaccines**) presented by the student(**Nuha Husham**) and examined him/herin its content and what is related to it, and that, in our opinion, it is adequate with (Viva Result) standing as a thesis for the degree of Masterin Information Technology-Software.

Signature:
Name:
Title:
Date: / / 2023
(Member)

Signature:
Name:
Title:
Date: / / 2023
(Chairman)

Signature:
Name:
Title:
Date: / / 2023
(**Member and Supervisor**)

Signature:
Name:
Title:
Date: / / 2023
(Member)

Approved by the Dean of the College of Information Technology,University of Babylon.

Signature:
Name:
Title: Professor
Date: / / 2023
(Dean of College of Information Technology)

Acknowledgements

First and foremost, I express my heartfelt gratitude to Allah, the Almighty God, for His abundant grace, blessings, and the inspiration that guided me throughout the completion of this work. Without Allah's guidance, this thesis would not have come to the light.

I extend special thanks and deep appreciation to my supervisor Dr. Sura Zeki Al-Rashid, may Allah bless her with success. I consider myself incredibly fortunate to have received her invaluable guidance and supervision. Her profound knowledge, scientific expertise, and practical experience played a crucial role in shaping this research. My gratitude towards her knows no bounds.

I would also like to express my sincere appreciation to all those with whom I have had the privilege to learn and collaborate over the past years, particularly the members of the College of Information Technology at the University of Babylon.

Nuha

2023

Abstract

In the immunology and disease prevention, vaccines are essential in conferring immunity against harmful pathogens and toxins. These immune-boosting agents typically consist of proteins or peptides known as antigens, which stimulate the production of antibodies to shield against potential invaders. Comprehending the constituents of vaccines is paramount for progress in the field and safeguarding public health. This work addresses the problem of identifying informative genes related to vaccine response. Through parallel, sequential, and hybrid feature selection approaches, then compare their effectiveness in selecting relevant genes. The parallel and hybrid feature selection phase highlights 16 genes , while the Sequential feature selection identifies 11 genes.

In addition, the performance of different machine learning models using the chosen attributes are assessed. The parallel feature selection step yielded impressive results, with the Logistic Regression classifier achieving a perfect accuracy score of 1.00 when employing ANOVA (60%) and both CHI^2 selections at 20% and 60%. Similarly, in the sequential feature selection phase, the Logistic Regression classifier also attained a flawless accuracy score of 1.00 for 6 paths for (60%). Furthermore, the hybrid feature selection stage resulted in an accuracy score of 1.00 with LR and ADA classifiers for MI_RF (20%) . Also, with LR and RF classifier coupled with ANOVA_RF at (60%). Finally, CHI^2_RF at (60%) with LR has accuracy score of 1.00.

In addition, to improve the efficacy and accuracy the CNN model is combined with parallel feature selection techniques such as MI (20%), MI (60%), and ANOVA (20%). This combination reveals notable improvement toward accuracy. The performance of the CNN model on features acquired from various feature selection methods, including four paths for 20%, and confirms that sequential feature selection improves system accuracy and predictive capabilities. Hybrid feature selection methods, such as MI_RF (60%), ANOVA_RF (20%), and CHI^2_RF(20%), improves system accuracy to 1.00.

To validate the reliability and generalizability of the model, a separate set of data that had not been previously seen by the system was used for testing. This approach involved randomly selecting 10% of the original dataset, which was completely new to the system. This new dataset was kept separate from the training data, which accounted for 90% of the original dataset. The training data was further divided into a 70% portion for model training and a 30% portion for model testing. The model's performance was then evaluated on the untouched 10% of data, and the results demonstrated its accuracy and success.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
ANN	Artificial Neural Networks
ANOVA	Analysis of Variance
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
DT	Decision Tree
FN	False Negative
FP	False Positive
GEO	Gene Expression Omnibus
KNN	K-Nearest Neighbors
LR	Logistic Regression
MI	Mutual Information
ML	Machine Learning
MLE	Maximum Likelihood Estimation
NAbs	Neutralizing Antibodies
NCBI	National Center for Biotechnology Information's
PCA	Principle Component Analysis
ReLU	Rectified Linear Unit
RF	Random Forest
RFE	Recursive Feature Elimination
RNN	Recurrent Neural Network
SVM	Support Vector Machines
TCGA	The Cancer Genome Atlas
TN	True Negative
TP	True Positive

List of Algorithms

Chapter One

General Introduction

1.1. Introduction

One of the most successful public health initiatives for controlling and preventing infectious diseases worldwide is vaccines. Measles, rubella, and polio deaths, morbidity, and disability have decreased due to vaccination. SARS-CoV-2, which causes the COVID-19 pandemic, is a new and re-emerging disease-causing agent that vaccine development must address. The scientific community is challenged to develop innovative vaccines that can protect against these emerging viruses and provide long-term immunity worldwide [1], [2].

Vaccines are medicinal interventions that are intended to elicit an immune response capable of recognizing and eliminating dangerous microorganisms or poisons. The creation of specialized proteins known as antibodies, which may neutralize these foreign chemicals within the body, aids in this process. Such important molecules are produced by B cells, a kind of white blood cell capable of identifying antigens (i.e., chemicals that elicit an immune response) from outside sources and making appropriate antibodies in response [3]. Immunity to specific pathogens/toxins depends on vaccines. These hazardous substances produce vaccine antigens, usually proteins or peptides. By stimulating our body's defenses, vaccines boost antibody production, allowing us to fight off invaders before they cause serious harm. Thus, understanding vaccine components is essential for immunology and disease prevention research [4], [5]. There are many live-attenuated inactivated subunit, nucleic acid, and polyvalent vaccines for various infections. Live attenuated vaccinations contain weakened but active microorganisms that reproduce in the body to stimulate the immune system. Inactivated vaccines contain destroyed organisms that don't replicate but stimulate the immune system. Subunit vaccines contain only pathogen components needed to induce immunity. Finally, nucleic acid-based immunizations use DNA or RNA

to create antigen proteins and kickstart an immune response against invading organisms' antigens [6], [7].

In vaccination research, bioinformatics and microarray technologies help us understand immune responses and find vaccine efficacy biomarkers. To understand the massive biological data produced by high-throughput technology, scientists have turned to bioinformatics, which combines biology, computer science, and statistics [8]. For example, microarray technology has enabled scientists to simultaneously monitor the expression levels of thousands of genes, considerably facilitating the study of gene expression patterns. Microarray technologies, such as the Gene Expression Omnibus (GEO) database, enable scientists to investigate the complex molecular processes that underpin immune responses to vaccinations [9]. In vaccine development, bioinformatics and microarray analysis can identify gene expression signatures related to vaccine-induced immune responses, such as active immune pathways, biomarkers, and long-lasting immunity. These methods can help identify vaccine candidates and create more personalized vaccination plans by revealing vaccine efficacy and safety [10].

Deep learning and machine learning algorithms have developed as powerful tools for the thorough analysis of complex biological data, including genomic and proteomic information. Deep learning is a subset of machine learning techniques that involves directing artificial neural networks to discover patterns in datasets [11]. Biomedical research has greatly benefited from deep learning approaches to medication creation, medical imaging analysis, and genomics research. In vaccines and immunology, deep learning algorithms have found epitopes or specific sections of the pathogenic antigen that can elicit an immune response, enabling the rational creation of innovative, effective vaccines. Deep learning models could also study virus genetics and chemistry to find therapeutic compounds and vaccines [12], [13].

1.2. Thesis Motivation

The global outbreak of Covid-19 in late 2019 had significant implications for multiple domains, such as public health, economies, and social dynamics on a global scale [14], [15]. Over 605 million confirmed cases and 6.5 million deaths have been reported in the global SARS-CoV-2 outbreak[16]. Coronavirus disease, or Covid-19, is a respiratory infection that mostly affects the lungs. Air travel is its main mode of transmission [17]. Vaccines protect individuals and societies from serious illnesses, hospitalizations, and deaths by preventing contagious diseases. Understanding what makes vaccines effective is essential to improving vaccine development and implementation strategies. Recent advances in machine learning and deep learning allow the analysis of large genomic and proteomic datasets. This has allowed researchers to understand vaccine potency mechanisms. These methods help researchers identify genetic and immune factors that affect vaccine responses and develop predictive vaccine effectiveness models[5], [18].

This thesis employs machine learning and deep learning techniques to examine the effectiveness of various vaccinations in terms of their capacity to elicit antibody responses and genetic immunity. Through the utilization of extensive datasets that include genetic and proteomic data, clinical trial data, and real-world information, the development of comprehensive predictive models will be undertaken to evaluate vaccine effectiveness.

1.3. Research Problem

Genetic immunological and antibody response analysis methods have limitations in assessing vaccine efficacy while taking into account age, gender, genetics, and health. Addressing these limitations is essential to understanding vaccine efficacy and optimizing vaccination strategies for different populations.

1. The consumption of Covid-19 vaccines elicits various physiological responses and outcomes within the human body . The examination of the impact of Covid-19 vaccines at the genetic level will elucidate the specific genes and gene clusters that are influenced by the administration of these vaccines . Subsequently, in response to various Covid-19 vaccines, the immune system and antibody responses will be evaluated through the analysis of gene expression pertaining to specific genes.
2. There is a need to understand the individual response level to vaccines based on genetic expression, as vaccine response varies widely from person to person, in addition to symptoms that may develop in one person but not another.
3. Genetic immunological and antibody response analysis using traditional and statistical methods cannot accurately measure vaccine effectiveness and interactions with age, gender, genetics, and health. As a result, this thesis presents a machine learning (ML) and Deep Learning (DL) method for predicting COVID-19 vaccination-induced immunological and antibody responses.

1.4.Research Questions

The focus of this study centers on comprehending the variables that contribute to the effectiveness of vaccines and employing machine learning algorithms to enhance these factors. Furthermore, the goal is to identify significant biomarkers that possess a strong probability of accurately predicting the effectiveness of vaccinations, thereby facilitating the advancement of more potent vaccine formulations.

1. What factors determine vaccine efficacy, and how can machine learning algorithms be used to fine-tune them?
2. How can biomarkers with a high likelihood of vaccination efficacy be identified using machine learning technologies to help develop more potent vaccines?

1.5. Thesis's Aim and Objectives

The aim of this thesis is to comprehensively explore the efficiency of diverse vaccines in enhancing antibody response and genetic immunity by applying advanced machine learning and deep learning methodologies. The objectives of the thesis are as follows:

1. Select a subset of characteristics (genetic information) from gene expression data using a systematic approach.
2. Create machine learning and deep learning models that use the selected genes to make accurate predictions, then evaluate their performance using various accuracy metrics.
3. Identifying critical genes that are influenced by variables such as vaccinations.
4. Furthermore, by using sophisticated machine learning and deep learning approaches, this research intends to increase overall prediction accuracy.

By achieving these goals, this study hopes to improve current information about

vaccine efficacy while also providing new insights into using machine learning approaches for prediction purposes in vaccination research.

1.6. Challenges of the Research Problem

Several academic challenges may arise when utilizing machine learning and deep learning methods to investigate the efficacy of different vaccines on antibody response and genetic immunity. Microarray data analysis is difficult due to its complexity, which is characterized by high dimensionality, limited sample size, uneven class distribution, noisy data structure, and variability in feature values. These difficulties frequently result in lower classification accuracy and overfitting concerns. This thesis tries to solve these issues by presenting a new method for reducing overfitting and improving classification performance.

1.7. Related Works

In recent years, the discipline of machine learning and deep learning has made major advances in allowing the interpretation of massive volumes of gene expression data. This includes data from vaccine studies, which are becoming increasingly crucial for understanding immunity and generating new vaccines. Identifying dynamic gene expression profiles during sequential vaccination is of special interest to researchers because it can give information on the underlying molecular mechanisms driving vaccine-induced immune responses. Machine learning approaches have proven to be effective in this endeavor, with various research utilizing these techniques to uncover relevant gene expression profiles linked with successful vaccine response. Table (1.1) presents a comprehensive overview of the relevant literature.

Zhang (2022) [19] employed vaccines against COVID-19 that can protect against SARS-CoV-2 infections, but they may have short- or long-term impacts on

COVID-19-related disorders and cause new forms of those diseases. A study using high-performance genomic biomarkers found that the BNT162b2 vaccine lowered MND1 and CDC6 expression while increasing ZNF282 expression in SARS-CoV-2-naive individuals. The immunization showed a negative effect on COVID-19 convalescent octogenarians, increasing MND1 and CDC6 expression while decreasing ZNF282 levels. Max-linear competitive logistic regression classifier with an accuracy of 88.70% will be used to classify BNT162b2 vaccination responses between COVID-19 convalescent patients and SARS-CoV-2 naïve individuals.

Lee et al. (2022)[20] examined the molecular transcriptome, germline allelic variations of immunoglobulin loci, and anti-Omicron antibody levels in 46 office and lab employees from the Republic of Korea after ChAdOx1-BNT162b2 immunization. The original SARS-CoV-2 strain's anti-spike-specific IgG antibody levels grew from 70 to 14,000 to 142,000 AU/ml one, three, and seven days after the second vaccination. Titers against VOC, including Omicron, were two- to three-fold lower but greater than those observed after BNT162b2-BNT162b2 immunization. RNA-seq of peripheral immune cells showed interferon pathway activation and elevated IGHV clonal transcripts expressing neutralizing antibodies. scRNA-seq showed enriched B cell and CD4+ T cell responses in both ChAdOx1-BNT162b2 and BNT162b2-BNT162b2 patients, but a greater clonal development of memory B cells in the former. Heterologous ChAdOx1-BNT162b2 immunization produces a stronger innate and adaptive immune response than homologous BNT162b2.

Rezaee et al. (2022) [21] introduced a two-step technique for gene expression in diverse disorders, which includes identifying highly effective genes using soft ensemble and classifying them using a unique deep neural network. The feature selection approach combines three strategies for selecting wrapper genes using the

k-nearest neighbor algorithm, resulting in a model with strong generalizability and low error rates. Soft ensemble is used to identify the most potent subsets of genes from three microarray datasets associated to DLBCL, leukemia, and prostate cancer. To categorize these datasets, a stacked deep neural network is used, with average accuracies of 97.51%, 99.6%, and 96.34%, respectively.

Abdelwahab et al. (2022) [22] proposed a structured approach using various feature selection techniques to find genes substantially linked with LUAD. On RNA-seq data from the Cancer Genome Atlas (TCGA), the study employs many methods, including mutual information (MI), recursive feature elimination (RFE), support vector machine (SVM) classification model, and embedded Random Forest algorithm. Potential biomarker genes for LUAD are identified by combining the results of these several methodologies. The proposed framework found a total of 12 possible biomarkers that have a strong link with various kinds of lung cancer, including LUAD. A prognosis model is developed using expression profile analysis of these selected biomarkers, yielding a remarkable accuracy rate of up to 97.99%.

Liu and Yao (2022) [23] proposed a gene selection method based on KL divergence to pick relevant genes with larger KL divergence as model features. The resulting deep neural network model employs Focal Loss as the loss function and the k-fold cross-validation method, with k set to five, for model verification and selection. According to our findings, using the validation dataset, this technique yields an AUC value of 0.99, showing strong generalization performance in reliably forecasting lung cancer incidence. As a result, it can be stated that employing a deep neural network based on KL divergence gene selection is an excellent method for predicting the occurrence of lung cancer. With an astounding accuracy record of up to 99.93% and 99.96%, a prognostic model is created utilizing expression profile analysis of these chosen biomarkers.

Al-Hatamleh et al. (2021) [24] did a study in which 2213 participants in Jordan participated in a trial and were all given one of six COVID-19 vaccinations. The researchers examined the reactions and side effects of the subjects. According to the study's findings, the majority of post-vaccination adverse effects are modest and do not endanger the patient's life. The severity of these negative consequences could be predicted using machine learning algorithms. Furthermore, the RF, XGBoost, and MLP shared information with both groups based on the type of vaccine, demographics, and potential side effects. Precision is excellent (0.80, 0.79, and 0.70, respectively).

Papadopoulos et al. (2021) [25] used machine learning algorithms on 302 participants, is to uncover critical factors that are predictive of NAb levels following immunization. Younger patients had higher levels of neutralizing antibodies than older patients, according to the results of PCA, FAMD, k-means clustering, and random forest analysis. Neutralizing antibodies (NAbs) against SARS-CoV-2 are negatively influenced by characteristics such as age, weight, gender, and autoimmune disorders nine months after immunization. It was discovered that characteristics such as age, BMI, and autoimmune illnesses had a negative impact on NAbs. The RF classifier's total prediction accuracy is 66.32%.

Koul et al. (2020) [26] combined recursive feature elimination with mutual information to construct a randomized ensemble technique for picking features from cancer gene expression data. The approach is applied to a dataset of gene expression in leukemia. Comparing classification accuracy for different-sized feature subsets using linear SVM and logistic regression classifiers, 99% and 96% classification accuracy are obtained with a gene subset of size 316.

Chen et al. (2020) [27] employed machine learning methods to detect different host biomarkers related with COVID-19. They identified critical biomarkers using feature selection methods such as Boruta, Max-Relevance, and Min-Redundancy

and achieved outstanding accuracy rates with several classifiers. With an accuracy rating of 93.8%, the Support Vector Machine classifier stood out.

Ahlawat et al. (2020) [28] used unstructured gene expression data to construct Convolutional Neural Network models that can efficiently categorize tumor and non-tumor samples into their respective cancer kinds or normal states. The three CNN models that have been built, namely 1D-CNN, 2D-Vanilla-CNN, and 2D-Hybrid-CNN, are based on different designs of gene embeddings and convolution techniques. To put the models to the test, the researchers used a pooled dataset of over ten thousand samples from thirty-three different cancer types, as well as matched healthy tissues from the TCGA. All groups, including normal cells, have impressively high prediction accuracies ranging from 93.9% to 95%. Furthermore, it was discovered that one model - the1-DCNN - made a substantial contribution by assisting in the identification of 2090 distinct possible biomarker genes thought to be connected across different diseases. Summary of the related works, with more details are listed in Table (1.1).

Table (1.1): Summary of the Related Works

Ref. and year	Dataset	Data Type	Model	Accuracy
Zhang (2022) [19]	GEO database under accession number GSE190747	BNT162b2 vaccine	max-linear competing logistic regression	88.70%
Lee **et al. (2022)**[20]	GEO database under accession number GSE201535	ChAdOx1-BNT162b2 vaccination	Statistical analysis	N\A
Rezaee et al. (2022)[21]	lymphoma dataset	Cancer	DNN	97.51%
	leukemia dataset			99.6 %
	prostate dataset			96. %
Abdelwahab et al. (2022) [22]	TCGA-LUAD	Lung Cancer	SVM	97.99%
Liu and Yao (2022) [23]	TCGA-LUAD	Lung Cancer	DNN	99.93%
	ICGC-LUAD			99.96%
Al-Hatamleh et	The Online Survey	COVID-19 Vaccination	MLP	0.80
			XGBoost	0.79

al. (2021) [24]	Side Effects Following COVID-19 Vaccination		RF	0.70
Papadopoulos et al. (2021) [25]	Demographic and Gene Expression dataset	BNT162b2 vaccine	RF	66.32%
Koul et al. (2020) [26]	Leukemia gene expression dataset	Cancer	Linear SVM	99 %
			Logistic Regression	96 %
Chen et al. (2020)[27]	GEO database under accession number GSE161731	COVID-19	DT	86.7
			KNN	88.2
			RF	92.3
			SVM	93.8
Ahlawat et al. (2020) [28]	Gene expression TCGA dataset	33 cancer types	1D-CNN	95.5
			2D-Vanilla CNN	94.87
			2D Hybrid-CNN	95.7

1.8. Thesis Outline

The present thesis comprises five primary chapters, with the current section serving as Chapter 1. The remaining part of this work is organized as follows:

- **Chapter 2:** This chapter dives into the history and evolution of vaccines. The role of antibodies in immunity and vaccination efficiency is also investigated, as are genetic factors that influence vaccine response and general immune system function. Furthermore, this paper gives a full overview of machine learning deep learning approaches used in vaccination research for better understanding and insight creation in the future in order to design efficient immunization programs.
- **Chapter 3:** This chapter describes the study's design and methodology for data collection, as well as the strategies used to prepare and refine obtained data. The process of selecting or modifying certain features prior to modeling. Machine learning approaches and deep learning algorithms are being implemented.

- **Chapter 4:** The presentation of results and the explanation of numerical data, Evaluation of both machine learning and deep learning models, comparison of their successes, and discussion of the ramifications of these results.
- **Chapter 5:** Outlines the key findings of the study, including their relevance to the discipline and prospective applications. It also discusses any limitations in this study and makes recommendations for further research.

Chapter Two

Theoretical Background

2.1. Introduction

This chapter discusses vaccines in terms of antibody response and genetic immunity. It discusses many biological concepts, such as gene expression and data representation using microarray technology, while also offering datasets to support the presented themes. Furthermore, this chapter provides a literary background on machine learning ideas such as data pre-processing procedures, feature selection methods, and both classic and deep learning models.

2.2.Biological Concepts

This section elucidates fundamental biological principles that underpin the generation of the data utilized, encompassing gene expression and microarray technology.

2.2.1. Vaccines

Immunization effectively controls communicable illnesses. It trains the immune system to fight certain illnesses. Antibodies, which neutralize infections, are produced by vaccines. Vaccination also activates T-cells and innate cells, which fight pathogens [29] . Vaccine antibody responses depend on vaccine kind, delivery method, and genetics. Live attenuated vaccines have stronger and longer-lasting antibody responses than inactivated ones, and injections may produce bigger responses than oral delivery [30]. Some people with specific gene variants coding for components like HLAs may have lower efficacy when vaccinated against certain diseases [31].

Vaccines produce antibody and genetic immune responses. Mobilizing immune-related genes may alter immune cell protein expression. Vaccination may increase gene transcription for cytokine production [32]. Vaccines protect people and society from hazardous illnesses. Vaccine-induced immune responses vary, but their public health value is undeniable [33].

2.2.1.1. BNT162b2 Vaccine

The BNT162b2 vaccine, which was developed by Pfizer-BioNTech, has emerged as a crucial asset in the global battle against the Covid-19 pandemic. The mRNA-based vaccine employs an innovative strategy to induce an immune response against the SARS-CoV-2 virus. BNT162b2 comprises messenger RNA (mRNA) that encodes the spike protein, which is found on the viral surface. Upon administration, the vaccine elicits cellular production of the spike protein, thereby inducing the immune system to identify it as an exogenous entity and initiate a defensive reaction. This encompasses the production of neutralizing antibodies and the stimulation of T cells in order to counteract potential infections [34], [35].

Extensive clinical trials have been conducted to evaluate the safety and efficacy of the BNT162b2 vaccine. The findings from extensive trials have revealed the notable effectiveness of the intervention in mitigating the occurrence of Covid-19 infection, severe illness, and the need for hospitalization. The vaccine has demonstrated efficacy in various age cohorts and ethnic populations, indicating its capacity to provide comprehensive immunity against the virus. The adverse effects associated with BNT162b2 are predominantly mild in nature, encompassing localized reactions at the site of injection as well as transient flu-like symptoms. The effective implementation of BNT162b2 in global vaccination initiatives has been instrumental in containing the transmission of Covid-19 and minimizing its consequences on the well-being of the general population.

2.2.1.2. ChAdOx1 Vaccine

The ChAdOx1 vaccine, which was collaboratively developed by the University of Oxford and AstraZeneca, has emerged as a prominent contender in the worldwide endeavor to mitigate the impact of the Covid-19 pandemic. The viral vector vaccine under consideration employs a non-replicating chimpanzee adenovirus, specifically ChAdOx1, as a vector for the purpose of delivering genetic material that encodes the

spike protein of the SARS-CoV-2 virus. Upon administration, the vaccine's vector infiltrates cellular structures, thereby inducing the synthesis of the spike protein and subsequently initiating an immune reaction [36], [37].

The efficacy of the ChAdOx1 vaccine has demonstrated promising outcomes in various clinical trials. The efficacy of the vaccine was found to be significant in providing protection against Covid-19, resulting in a notable decrease in the likelihood of experiencing severe illness and requiring hospitalization. Moreover, the safety profile of the ChAdOx1 vaccine is generally positive, as it is associated with predominantly mild and temporary adverse effects, primarily occurring at the site of injection or manifesting as mild flu-like symptoms. One of the prominent benefits associated with the ChAdOx1 vaccine is its convenient distribution and storage characteristics, as it can be maintained at standard refrigerator temperatures. The aforementioned attribute has rendered it a valuable instrument in vaccination initiatives, particularly in settings with limited resources.

2.2.2. Gene Expression

The expression of genes, which produces required proteins, is a fundamental process that defines the physical components of living beings. Transcription and translation are the first two steps in this process, which use enzymes to carry information from DNA to RNA, culminating in the production of proteins and other biological substances. One way for measuring gene expression from DNA or RNA is to use a DNA microarray [38].

Genes are the primary genetic building blocks responsible for encoding certain cellular proteins and RNA. These genes are DNA segments that contain critical genetic information required for life processes [39]. Protein synthesis is broadly acknowledged in molecular biology, with two critical steps depicted in Figure (2.1).

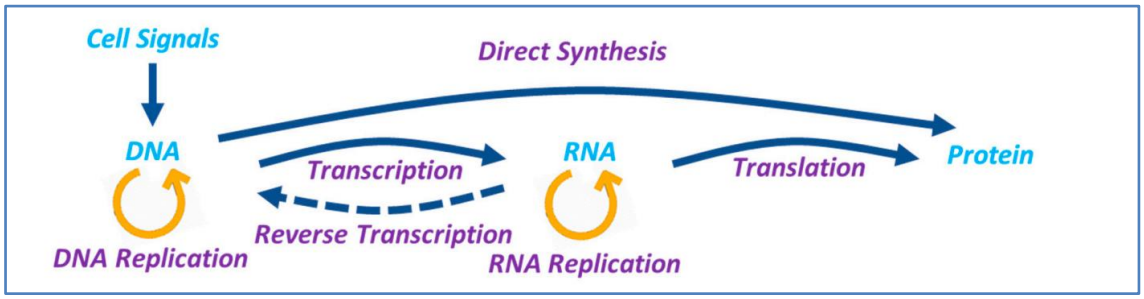


Figure (2.1): Dogma Central to Molecular Biology[39]**.**

Ribosomes serve an important role in gene expression by decoding information from mRNA through translation, which leads to protein production. This intricate process has two unique stages and is essential for scholarly understanding of cellular biology [40].

2.2.3. Microarray Technology

Because of its ability to evaluate hundreds or even thousands of genes in a single sample, microarray technology, also known as DNA chip, has been widely used in global gene expression research. This method is distinguished by its tiny sample size and ability to generate a large number of features using an incomplete rank and non-square matrix. As a result, while developing classifiers, this can result in many solutions [41].

The microarray technique, which allows biomedical researchers to examine the expression levels of thousands of genes at once, is one of the most recent discoveries in experimental molecular biology. Using microarray technology, several hundred data points are typically saved for each gene. The processed data from these images is represented in a gene expression matrix, which has rows expressing different genes and columns denoting the conditions under which they were evaluated [42].

The structure of the gene expression matrix, as illustrated in Figure (2.2), allows researchers to extract vital information about how certain genetic expressions present themselves under different settings, making it a potent instrument for academic investigation within genetics and beyond [43].

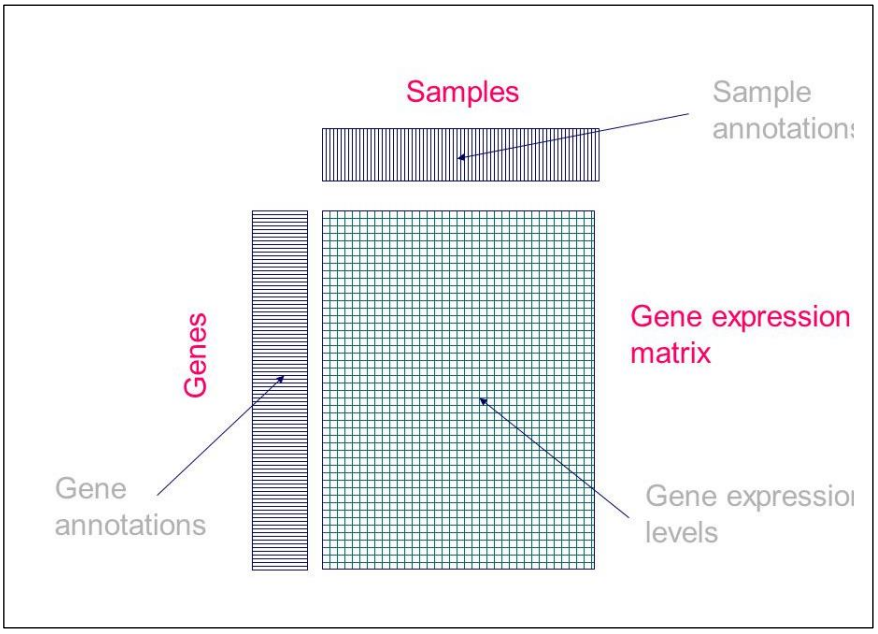


Figure (2.2): Gene Expression Matrix Structure [44]**.**

2.3. Data Preprocessing

The initial and critical phase of data preparation is critical to the building of an accurate and efficient ML model. This essential phase entails arranging data so that it may be analyzed more effectively. Its importance to the model's overall performance cannot be overstated. The primary goals of this stage include, but are not limited to, reducing extraneous data, identifying and managing correlations within the dataset, standardization procedures aimed at ensuring uniformity across variables, all while simultaneously striving for effective feature extraction methodologies - which may also include addressing missing values with normalization technology [45], [46].

2.3.1. Missing values

Missing values in data can be caused by a variety of sources, including human or machine processing errors, non-response to survey questions, and merging unrelated data. These missing values can cause problems such as decreased performance, skewed findings due to variations between complete and partial datasets [47], and the removal of particular features from machine learning models due to a lack of available information. To address these concerns, depending on the quantity of missing

information, several interruption techniques may be used, common options include using mean [48].

$$\mu = \frac{\sum_{i=1}^{n} X_i}{n} \quad (2.1)$$

Where:

μ is the mean of feature "X."

Xi is the value of feature "X" for the ith data point (sample).

n is the total number of data points with non-missing values in feature "X."

2.3.2. Data Normalization:

Data normalization is commonly regarded as the first stage in the processing of a dataset. This procedure is carried out before to using the data, for example, to change the range of values either upwards or downwards. Normalization is beneficial and effective when dealing with classification and regression challenges in datasets by changing characteristic values into more constrained ranges [49].

Min-max normalization is a common data scaling technique that converts variable values into a range between 0 and 1. This normalizing method comes in handy when you need to compare variables with different scales or ranges. The min-max normalization formula entails computing the difference between an initial value (X) and its smallest value (X_min), then dividing that difference by the difference between X_max (the variable's maximum value) and X_min. Using this method, you can retrieve normalized values for each variable in your dataset that falls inside the defined range [50], [51].

$$X\_new = (X - X\_min) / (X\_max - X\_min) \quad (2.2)$$

To do min-max normalization on a given dataset, first identify the minimum and maximum values for each variable in the collection. Once decided, employ these

derived metrics in conjunction with the aforementioned equations to turn each individual point into the normalized format, resulting in uniformity across datasets being compared [50].

2.4. Feature Selection Methods

A dataset is made up of data pieces that can be points, patterns, events, cases, samples, examples, or characteristics. As a result, these data objects are often identified by many properties that encompass an object's fundamental qualities, such as its mass and time of occurrence, among others. A feature may represent a detectable and measurable property, or it may be an intrinsic component of the underlying phenomenon [8]. The main issue with microarray data in bioinformatics is the "curse of dimensionality problem" that prevents significant insights from being extracted. Thus, identifying a gene group from such datasets is difficult. Due to a large gene pool and few samples, running ML models on microarrays takes time [52].

Feature selection involves selecting dataset characteristics that best describe the goal variable. It aims to improve computer efficiency, generalization, and machine learning problem interpretation. Features are selected using three methods, including filters. Filter methods prioritize statistical quality over machine learning. Information gain, mutual information, chi-square, and correlation coefficient are common measurements. Using feature subsets, wrapper-based methods evaluate machine learning algorithms. Based on the model's prediction ability, they select features using forward selection, backward elimination, or recursive feature removal. Model training uses feature selection and embedding[53]. Microarray data analysis uses feature selection mostly because biology and computer science/statistics use it. Feature selection identifies genes from microarray data that distinguish healthy and diseased individuals to help identify relevant genes [10].

Understanding a fundamental biological process necessitates a critical phase in which computational and statistical constraints associated with small sample sizes for

each feature are addressed. This field's expert's focus on fixing issues linked to overfitting, redundancy, and data noise. Trait selection approaches are used to reduce illogical features and improve classification accuracy by lowering the number of genes involved [54].

2.4.1. Mutual Information (MI) Method

Mutual Information is a statistical procedure that is used to determine the relationship between two random variables. It computes the quantity of information provided by one variable regarding the other variable. The information for one variable in relation to others in order to find mutual information must be calculated. Mutual information can be used to determine how well a subset of features covers the output vector. Consider (X) to be a gene and (Y) to be a category label. Set entropy is calculated using H(X) and H(Y), which may be represented using equations (2.3), (2.4), and (2.5), respectively [26], [44].

$$H(\mathrm{X}) = -\sum_{x \in \mathrm{X}} P(x)\ log\ (\mathrm{P}(x)) \quad (2.3)$$

$$H(\mathrm{Y}) = -\sum_{y \in \mathrm{Y}} P(\mathrm{y})\ log\ (\mathrm{P}(\mathrm{y})) \quad (2.4)$$

$$H(\mathrm{X}, \mathrm{Y}) = \sum_{x \in \mathrm{X}} \sum_{y \in \mathrm{Y}} P(x,\mathrm{y})\ log\ (x,\mathrm{y}) \quad (2.5)$$

Where P(x) and P(y) represent the probability of (x) and (y) respectively, P (x, y) represents the joint probability of (x) and (y) values occurring together .Mutual information (M(X, Y)) may be computed when the entropies have been determined as:

$$M(X,Y) = H(X) + H(Y) - H(X,Y) \quad (2.6)$$

Where $H(X, Y)$ is the joint entropy of Y given X. Using mutual information for gene expression analysis with machine learning can be a powerful tool to select relevant features and accurately classify samples. By applying mutual information feature selection in gene expression analysis, the relationship between genes and categories and identify which sets of genes have strong predictive power for a particular category label can be determined.

Algorithm (2.1): Mutual Information [44]

Input: D - a two-dimensional array with S samples and F features.

Output: SD - a subset of data containing the features with the highest mutual information values.

Begin

Step1. Initialize an empty set called features_subset.

Step 2. For each feature F[i] from 1 to the total number of features F:

- **For each class label C [i] from 1 to the total number of classes C:**
  - **Calculate the mutual information between feature F[i] and class label C[i] based on the equations in section (2.4.1).**
- **End of the inner loop.**
- **End of the outer loop.**

Step 3. Select the features F[i] with the maximum mutual information (F[i], C[i]).

Step 4. Sort the selected features in descending order based on their mutual information values.

Step 5. Store the sorted features in the subset of data SD.

End

Algorithm (2.1) illustrates the details of the mutual information process. As a result, MI can be used to focus on X, Y, the subset of attributes that are most likely to be reduced, while simultaneously improving the relevant data H (X, Y). Finally, this contributes to understanding how machine learning models represent the underlying relationships in the original dataset [55], [56]

2.4.2. Analysis of Variance (ANOVA) Method

The statistical process known as analysis of variance (ANOVA) is used to compare the various mean values of the dataset and determine whether there are any significant differences between the means of different groups (classes). The extent to which variation differs within and between groups is being investigated. The following

equations are for computing the sum of squares and the mean square error across and within groups. These calculations will eventually be utilized to compute the F-statistic, which is the ANOVA statistic [57], [58]. There are three types of ANOVA: one-way, two-way, and repeated measures ANOVA. One-way ANOVA has only one independent variable, whereas two-way ANOVA has two independent variables, as illustrated in [59], [60]**.** The group variation is calculated using Eq. (2.6) and Eq. (2.7):

1. The variation between the groups is calculated as:

Sum of squares between the groups

$$SSB = \sum n_i (x_i - \bar{x})^2 \quad (2.6)$$

Mean square between groups

$$MSB = SSB/df \quad (2.7)$$

Where:

df is the degree of freedom, df = (K−1), k is the total number of groups and $\boldsymbol{n_i}$ is the number of samples in group i, x_i. Is the mean of group i, $\bar{x}$ is the grand mean (mean combined for all groups).

2. The Variation within each group can be calculated by using equations (2.8) and (2.9).

Sum of squares within groups

$$SSW = \sum (n_i - 1)\,\sigma_i^2 \quad (2.8)$$

mean squares within groups

$$MSW = SSW/df_w \quad (2.9)$$

Where:

σ is the standard deviation. df*w* is the degree of freedom within groups.

df*w*= (N−K), and N is the number of samples.

The ANOVA-derived F-statistic is utilized for establishing the statistical

significance of inter-group variances **as** presented in Algorithm (2.2). In case the computed F-value exceeds a critical threshold, it implies that notable variations exist among groups[58]. The formula for determining the F-statistic is given in Equation (2.10).

F-statistic is calculated as :

$$F = MSB/MSW \quad (2.10)$$

Algorithm(2.2): Analysis of Variance Method [58]

Input: D - a two-dimensional array of size S * F, where S is the number of samples and F is the number of features.

Output: SD - a subset of the data.

Begin

Step 1: For each feature f*i*:

- **Calculate the between-group mean square (*MS*B) using equation(2.6).**
- **Calculate the within-group mean square (*MSW*) using equation(2.8).**
- **Calculate the F-value (*Fi*) by dividing *MS*B by *MSW*.**
- **Find the corresponding p-value (pi) for each F-value.**

Step 2: For each feature set f*i*:

- If the p-value (pi) is less than 0.001:

- Choose the feature, named fc*i*.

- Else:

- Cancel the feature.

End

2.4.3. Chi-square (χ^2) Method

The chi-square (χ^2) test is a statistical approach that analyzes observed and predicted data to see if there is any significant deviation. It is widely used in the analysis of categorical data, such as survey results or medical trial findings. The chi-square test statistic is calculated using Equation (2.11) [61]:

$$x^2 = \sum_{i=0}^{K} \frac{(O_i - E_i)^2}{E_i} \quad (2.11)$$

Where:
χ^2: is the chi-square
Oi: is the observed value that has been measured in the i[th] class.
Ei: is the expected value.
k:is the total amount of classes .

Algorithm (2.3): Chi-square Method [61]

Input: D - a two-dimensional array of size S * F, where S is the number of samples and F is the number of features.

Output: SD - a subset of data containing features with the highest Chi-square values.

Begin

Step 1:For each sample i = 1 to S:

- **For each feature j = 1 to F:**

  Calculate the Chi-square value between feature F[*i*] and class label C[*i*] using the equation (2.11).

- **End the loop for feature j.**
- **End the loop for sample i.**

Step 2:Select the features F[*i*] with the maximum Chi-square value (F[*i*] ,C[*i*]).

Step 3.:Return the selected features of Chi-square values.

End

To carry out the investigation, an estimate of expected frequency for each category is produced based on either a hypothesis or a null hypothesis. The latter phrase implies that there is no discernible difference between gathered data and expected figures, but the former implies otherwise [61]. A test statistic is then calculated by

adding the squared differences between observed and predicted frequencies; this total is then divided by the expected frequency per category, yielding a value of 2. As a result, if the figure is significant, it reveals a significant discrepancy between actual observations and expected outcomes, causing us to reject our null hypothesis as provided in Algorithm (2.3). The degree of freedom (df) of the test can be calculated by subtracting one from the number of categories [62], [63].

2.5. Machine Learning Techniques

Machine learning is a branch of AI that handles a wide range of issues. To tackle these challenges, several algorithms have been created. Machine learning advancements have enabled complex and high-capacity biomedical data to be computationally assessed, making machine learning models more practical in the medical business.

2.5.1. Adaboost Model

Adaboost, short for Adaptive Boosting, is a common ensemble learning method for classification and regression problems in machine learning. The Adaboost method entails combining numerous weak models to form a powerful combined model. Adaboost's main principle is to train weak learners iteratively on different subsets of the training data, with each iteration providing more weight to samples that were misclassified in the prior iteration. As a result, following weak learners focus more on challenging examples and learn to correct prior models' mistakes. The final ensemble model is built by merging all of the weak learners' predictions, often via a weighted majority vote [64]. The weak models are often low-accuracy decision trees, but the ensemble of these models reaches a higher level of accuracy. The Adaboost approach works by increasing the weight assigned to misclassified samples in each round of training, so that the succeeding weak model focuses on accurately classifying the misclassified examples from previous rounds. Adaboost entails the following phases in terms of equations and algorithms:

Algorithm (2.4): Adaboost Algorithm [64]
Input: Training set consisting of features and corresponding labels. **Number of weak models (T).**
Output: Ensemble model consisting of T weak models.
Begin **Step 1: Initialize the weights of the training instances to be equal.** **Step 2:For t = 1 to T:** - **Train a weak learner on the weighted training instances.** - **Calculate the error rate of the weak learner.** - **Update the weights of the training instances, giving more weight to instances that were misclassified by the weak learner.** **Step 3: Make the final prediction by combining the predictions of the weak learners.** **End**

The Adaboost method produces an accurate and robust model capable of correctly predicting the labels of incoming input samples [65].

2.5.2. Random Forest (RF)

Random Forest (RF) is a popular classification strategy that uses a combination of decision trees to improve prediction precision and robustness. This adaptable technique is suitable for both regression and classification challenges. The basic idea behind Random Forest is to build several decision trees by randomly selecting subsets of characteristics and training samples for each tree. Following that, the final anticipated outcome is generated by aggregating all outputs from all trees in the forest ensemble[66], [67].

In the realm of machine learning, Random Forest (RF) is a popular ensemble

learning approach used for categorization problems. This technique works to improve upon the accuracy and reliability of predictions made by individual models by combining the results of several decision trees. RF's exceptional feature selection capabilities is what sets it apart from competing algorithms. Feature selection involves determining which variables are most important to the correctness of the model [68].

When using feature selection techniques like Select From Model with random forests, the "get_ support" function is often available as part of the resulting feature selector object or model. This function returns a Boolean vector where each element represents whether a particular feature is selected (True) or not selected (False) by the feature selection method in combination with the random forests algorithm By using the "get_ support" function, can access the Boolean vector and identify which features have been selected as important or relevant for the given feature selection approach. This allows you to filter or subset your original feature set, keeping only the selected features for further analysis or modeling [69] . Random Forest has grown in popularity due to its capacity to handle high-dimensional data, estimate feature importance, and avoid overfitting. The Random Forest algorithm's major properties and phases are as follows [70]:

Algorithm (2.5): Random Forest Algorithm [70]

Input: Training dataset D with S samples and F features.

Number of trees N in the Random Forest ensemble.

Output: Random Forest classifier model with N decision trees.

Begin

Step1: For i= 1 to N:

a. Randomly select B samples with replacement from the training dataset D to form a bootstrap sample D[i].

b. Randomly select M features from the F features to consider at each node for splitting.

c. Build a decision tree T[i] using the bootstrap sample D[i] and the selected features.

d. Store the decision tree T[i] in the Random Forest ensemble.

End for

Step2: For each sample in the test dataset:

a. For each decision tree T[i] in the Random Forest ensemble:

i. Traverse the decision tree T[i] based on the selected features for that tree.

ii. Assign the sample to the corresponding leaf node in T[i] .

End for

b. Record the predicted class for the sample based on majority voting among the decision trees.

End .

For classification, the final prediction is determined by majority voting among the trees technique for classification tasks see Figure (2.3).

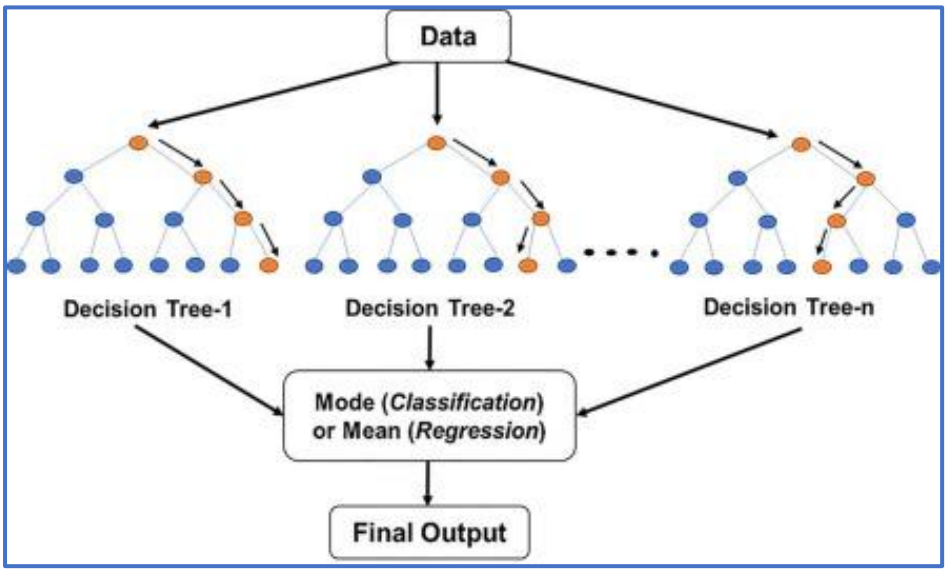


Figure (2.3): Architecture of the random forest classifier [71]

2.5.3. Logistic Regression (LR)

The logistic regression (LR) model is a suitable statistical method for addressing binary classification tasks. This classifier uses probability scores as the predicted values of the dependent variable to examine the relationship between a dichotomous dependent variable and one or more (categorical or continuous) independent variables. There is no requirement for the independent variables to have a normal distribution, linearity, or variance homogeneity[72].

Logistic Regression (LR) is a well-known statistical learning approach that is commonly used for binary classification applications. Despite its name, logistic regression is more commonly used for classification than regression. It simulates the interaction of a set of input variables (features) and a binary result variable. The Logistic Regression algorithm is comprised of the following Function [73], [74]:

1. **Sigmoid Function:** Logistic Regression uses the sigmoid function (also known as the logistic function) to transform the linear combination of input features into a probability value between 0 and 1. The sigmoid function maps any real-valued number to the range [0, 1] and is defined as:

$$\sigma(z) = 1 / (1 + e^{(-z)}) \quad (2.12)$$

where z is the linear combination of input features and their corresponding weights.

2. **Hypothesis Function:** The hypothesis function of Logistic Regression calculates the predicted probability of the positive class (e.g., class 1) based on the input features and their associated weights. It is formulated as:

$$h\theta(x) = \sigma(\theta^T * x) \quad (2.13)$$

Where hθ(x) is the predicted probability, σ is the sigmoid function, θ is the weight vector, and x represents the input features.

3. Cost Function and Gradient Descent: The cost function, also known as the log loss or binary cross-entropy loss, is used to measure the difference between the predicted probabilities and the true labels. The goal is to minimize the cost function using optimization algorithms such as gradient descent.

4. Decision Boundary: A decision boundary in binary classification is set by a threshold value on the projected probability. If the anticipated probability is greater than the threshold, the instance is categorized as positive; otherwise, it is classified as negative.

Using the sigmoid-shaped logistic function, each weighted feature vector is assigned a value between 0 and 1[67]. LR can assess the likelihood of an occurrence by fitting the data to a logistic curve, as shown in Figure (2.4).

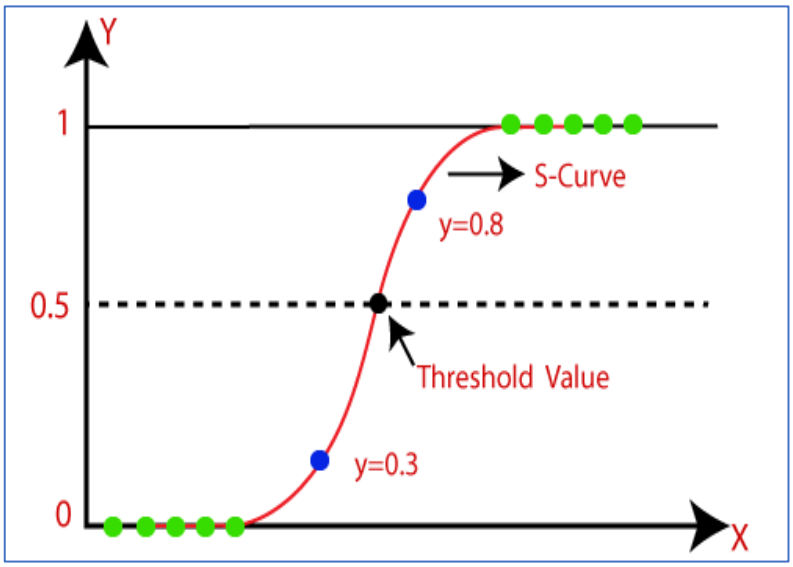


Figure (2.4): Method of LR's Logistic Curve[67]**.**

2.6. Deep Learning (DL)

To enable robots to learn new information on their own, researchers have resorted to a new subject known as "deep learning," which tries to develop theories and algorithms that mimic human neural networks. Deep learning is a type of machine learning that was developed first as an artificial intelligence (AI) technique to replicate how humans learn in a certain domain [75].Because each layer of a deep learning algorithm builds on the one below it, the phrase "hierarchical learning" defines how these algorithms are structured. Most existing machine learning algorithms are expected to have a linear structure. In 2007, deep learning was first introduced [76]. Deep learning is a cutting-edge machine learning technique that bridges the gap between traditional ML and artificial intelligence. Deep learning has numerous applications, including but not limited to object detection, speech recognition, and even medicine [77].

Deep learning was developed to overcome the limitations of human intelligence in AI problem solving. Deep neural networks outperform shallow ML methods in most applications involving the processing of text, image, video, speech, and audio data, making DL particularly effective in many domains with enormous,

high-dimensional data. The rapid advancement of ML algorithms and processing is one reason for deep learning's appeal [78], [79].

2.6.1. Deep Neural Networks (DNN)

A deep neural network (DNN) is an artificial neural network with multiple hidden layers. MLP is the most common Artificial Neural Networks (ANN) structure in DNN. Neurons in a neural network are linked and collaborate across multiple layers. Because the number of weights in a DNN is enormous, sometimes in the hundreds or millions, training the samples necessitates a considerable investment in computational labor and data sources. The Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN) are two of the most well-known and powerful deep learning network topologies. Numerous improvements in the last decade have made it possible to train such a network [80], [81].

The multilayered organization of the nervous system reflects its complexity, as shown in Figure (2.5).

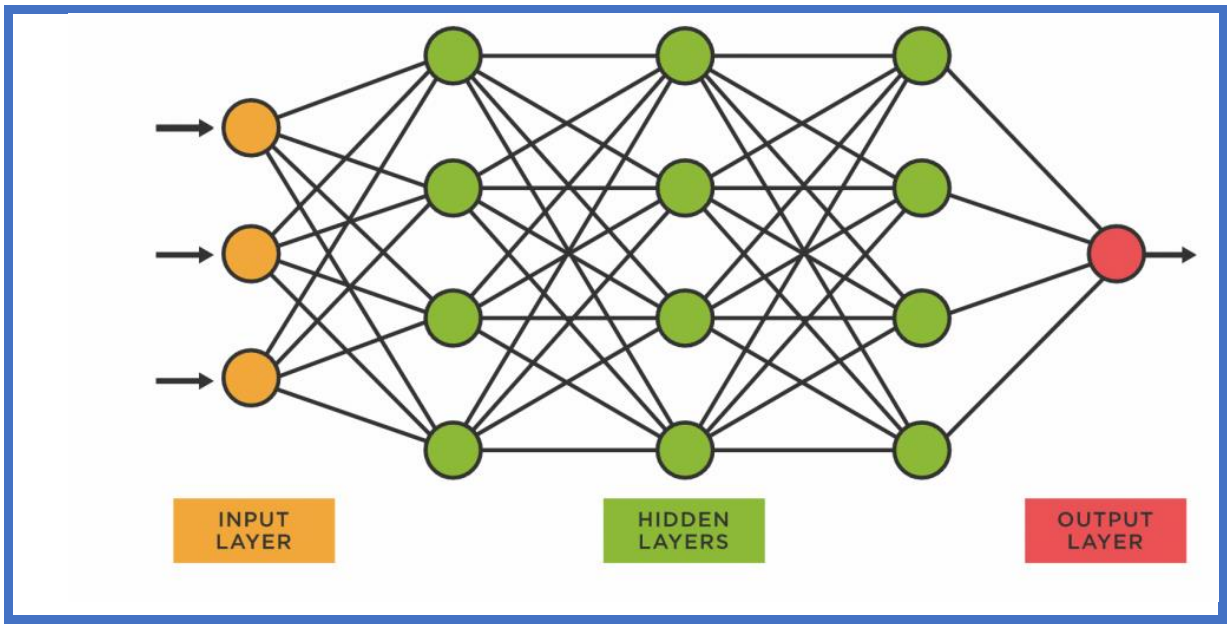


Figure (2.5): Structure of a deep neural network [82]

2.6.2. Types of Learning Algorithms

In many ways, deep learning is critical to human survival. It has had a significant impact on a variety of sectors, including illness detection, precision medicine, and voice recognition. Feature extraction from big datasets, for example. Furthermore, DL learning can help overcome the constraints of shallow networks. Deep neural networks (DNNs) achieve their jobs by utilizing complicated algorithms and designs comprised of numerous (deep) layers of modules [83], [84]. Some of the most popular deep learning algorithms are listed in Figure (2.6).

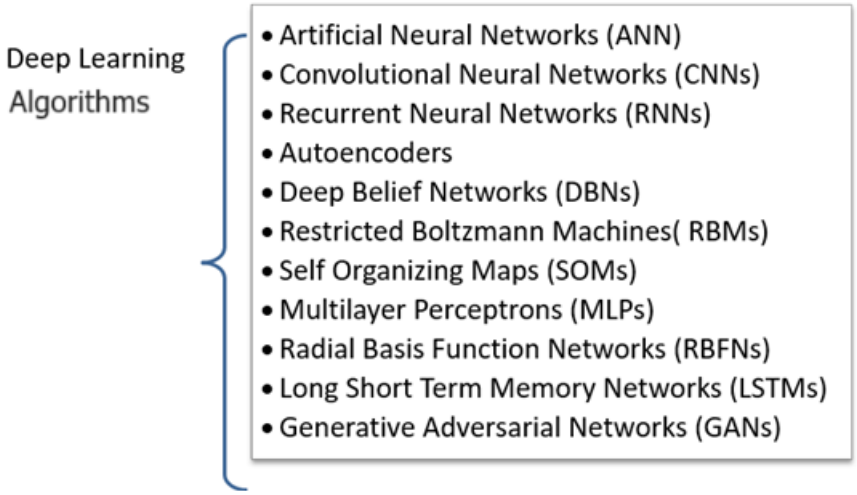


Figure (2.6): Deep Learning Algorithms [85] .

These methods are crucial in deep learning since they can be applied to almost any data type and require a lot of data and computer power to solve a variety of challenging tasks.

2.6.3. Convolutional Neural Network (CNN)

The convolutional neural network (CNN) is a typical type of deep neural network used in machine vision. The Convolutional Neural Network (CNN) is a deep learning technology that attempts to imitate how the human brain operates [86]. It belongs to the feed-forward network class of artificial neural networks. Convolutional neural networks (CNNs) networks are similar to multi-layer

networks (Perceptrons) in that they can merge several networks with local connections into a single, unified network. In addition to increased accuracy in automated diagnostic systems, CNN also offers promise in the domain of disease prediction. CNNs have gained popularity in the AI community as a result of their tremendous data-processing capacity [87].

In a convolutional neural network (CNN), the output of each layer is used as input for the next layer. The network is comprised of an input layer and an output layer. The network's hidden layers are located between the input and output nodes. Every "layer" consists of only one "activation function" program. Overall, the CNN model improved prediction accuracy by emphasizing more complex interactions between genes and the target set [88].

2.6.3.1. Components of CNN Architecture

Convolutional neural networks can be trained to mimic human cognitive abilities. The ability to predict future outcomes is a significant benefit of this network. A typical CNN has two main parts: the Feature extraction and the classifier [89]:

A) *Feature Extractor*

The initial step in the data processing pipeline for a CNN is feature extraction and the generation of a feature map. In CNN, each of the various filters has a distinct purpose. Because of this, many different types of feature maps were created, each of which stands for a different group of filters. The output of the features extraction method is a low-dimensional features vector that is given into a classifier. Multiple layers make up the feature extractor (multiple convolution layers with optional pooling layers). To create feature maps, a convolution layer is used to combine the input and filter before being reduced in a pooling layer. In order to extract increasingly complex features, the system

is iterated by feeding the output feature maps back into the system as input feature maps. Finally, a low-dimensional feature vector is created by flattening the reduced-dimensional feature maps[90].

B) *Classifier*

The best features from each of the retrieved feature maps are then combined into a single low-dimensional feature vector for use in training a classifier. The classifier will report the likelihood of an input belonging to a specific class. To do this, a classifier composed of one or more completely linked layers is used [90].

2.6.3.2. CNN Architecture

A CNN's input layer, which reflects the model's input (the features that were intentionally picked), can be any size without affecting the network as a whole. When processing gene expression data using a convolutional neural network (CNN), the input layer is typically a (n m)-by-(m) matrix, where n is the sample size and m is the number of features [91], [92]. Figure (2.7) illustrates the several layers that make up a neural network. Specifically, they are as follows:

1. Convolutional layer.
2. Pooling Layer.
3. Fully Connected Layer.
4. Non-Linearity Layers
5. Dropout Layer
6. Adam Optimization Algorithm

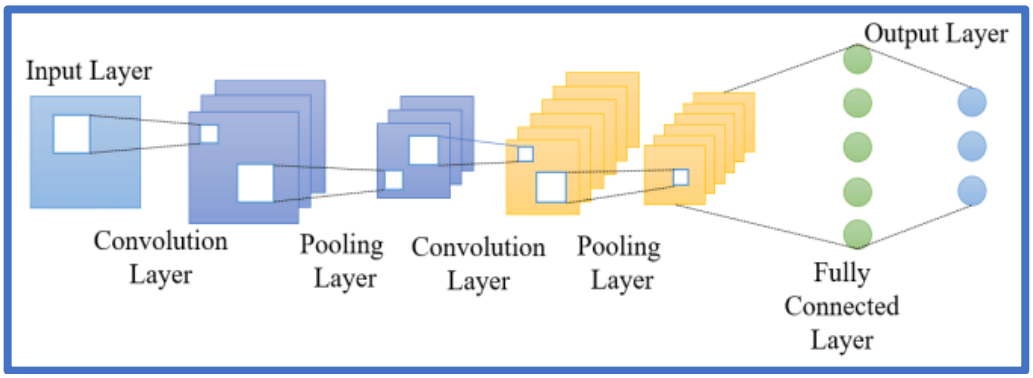


Figure (2.7): A simplified representation of a convolutional neural network (CNN)[93].

1) Convolutional layer

The convolution layer, which can handle high-dimensional data, is the backbone of a convolutional neural network (CNN). The first layer in a convolutional neural network is the convolution layer, which communicates with the second layer (the pooling layer) in part. The pooling layer, as shown in Figure (2.8), may be fed information from a 3x3 window of input neurons that incrementally travels over the data from top left to bottom right at regular intervals (the "stride" number, which is generally 1). When the kernel hits the end of a column, it shifts down one cell until it reaches the end of a row, and so on, until all of the information has been recorded [94].

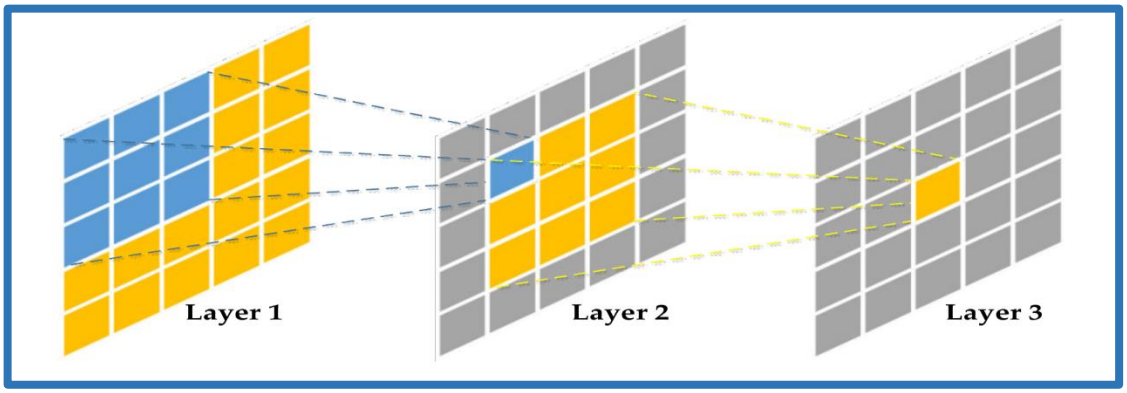


Figure (2.8): Schematic diagram of the receptive field in CNNs[94].

The receptive field is a tiny window region created from the input data. To extract features, a tiny section of the input data will be convolved with a shared weights window called kernel or filter . Figure (2.9) depicts the convolution process.

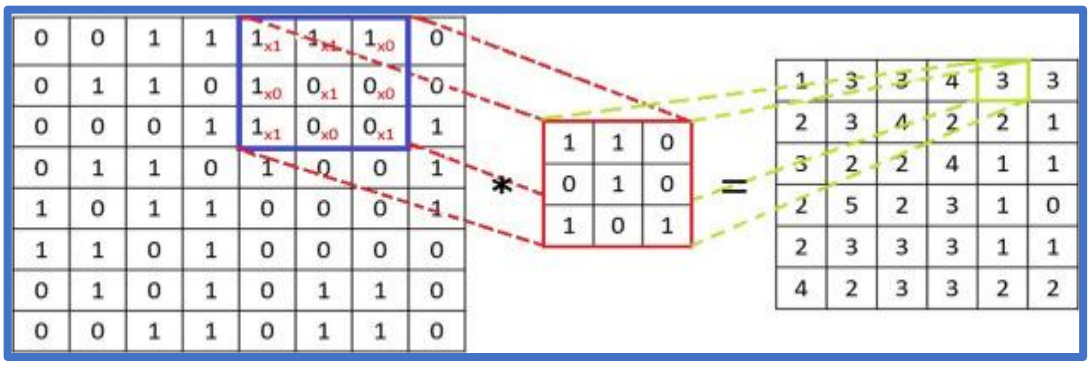


Figure (2.9): The Convolution Layer [95].

Many distinct filters are applied to a single entry. The activation maps are integrated in the convolutional layer to produce a single output file that serves as the input data for the succeeding layer. The values in the filter matrix are appropriately represented by the default weights. Each filter must have individual values for these parameters in order for its output matrices to have distinguishing characteristics or features [95].

Convolutional neural networks (CNNs) rely heavily on these hyperparameters:

a) **Number of filters**: Filters can be used, and there are many to choose from, all of which have somewhat different dimensions.

b) **Filter size**: An essential CNN hyperparameter is the filter size. It establishes the region of the input data that the filter examines (the receptive field). Careful consideration must be given to filter size selection to ensure that significant patterns are captured without the filter outgrowing the input data.

c) **Stride**: Local receptive field of a filter is created by concurrently shifting a certain number of cells. A single cell can move in both horizontal and

vertical directions simultaneously. When the stride is too short, there will be overlap, and vice versa.

d) **Padding** : Padding is a concept incorporated into the CNN architecture to help enhance accuracy. Padding is used to control how much the output of the convolutional layer is shrunk.

The convolutional layer's feature map is substantially less in size than the input image. The resulting feature map prioritizes information closer to the map's centers over data near the map's boundaries. To protect the feature map from decreasing to a practical size, blank rows and columns are put at the image's borders. The following equations are used to calculate the size of the final feature map: The relationships between the feature map size, the kernel size, and the stride are defined in (2.14) and (2.15) [96].

$$W_{nx} = W_{n-1x} - F_{nx}S_{nx} + 1 \tag{2.14}$$

$$W_{ny} = W_{n-1y} - F_{ny}S_{ny} + 1 \tag{2.15}$$

where (S_{nx}, S_{ny}) is the stride size, (F_x, F_y) is the kernel size, and (W_{nx}, W_{ny}) is the size of the output feature map. The layer index is denoted by n here.

2) Pooling Layer.

CNN's output is the outcome of a sequence of layers that mix convolutional processing with pooling. This layer's major function is to provide reduced-dimensional output by compressing the input dimensions while retaining the most relevant details. In this layer, maximum and average pooling are utilized to reduce dimensionality. If maximum or average pooling [97] is used, the pooling layer receives an input feature map and pools its features into non-overlapping blocks, each of which returns a single value. Figure (2.10) depicts the best potential pooling.

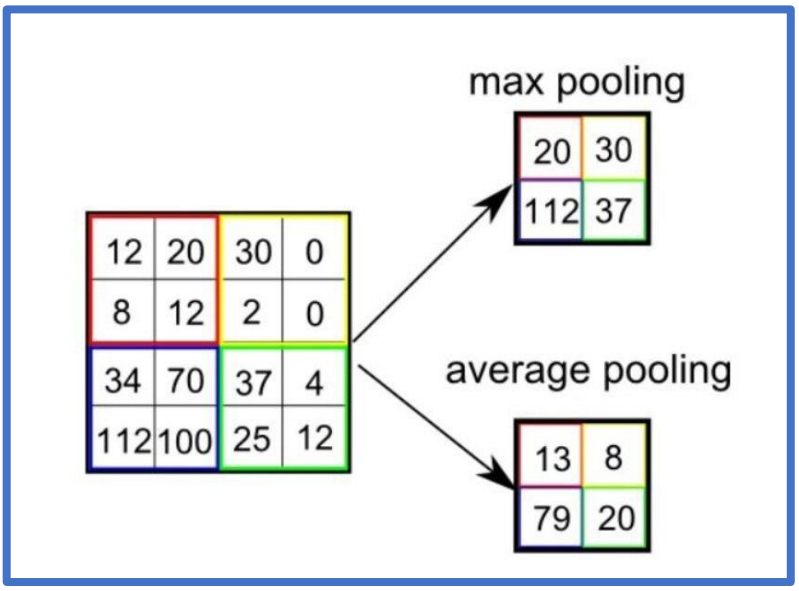


Figure (2.10) : Max-Pooling Operation with a 4X4 block size [97].

3) Fully Connected Layer (Classification layer) (FC).

The network's last layer is a completely connected layer (FC), also known as a dense layer. The layer is said to be fully linked when all of the neurons in the layer below are connected to all of the neurons in the layer above. The generated feature map must be flattened into a feature vector to complete the connection between the output layer and the previous layer. The final CNN layer utilizes either a softmax or sigmoid activation function to classify the learned data, and the number of neurons in the output layer is proportional to the number of classes. These activation algorithms improve multi-class and binary class classification performance, respectively [98]. The connection between finished feature maps and a completely connected layer is depicted graphically in Figure (2.11).

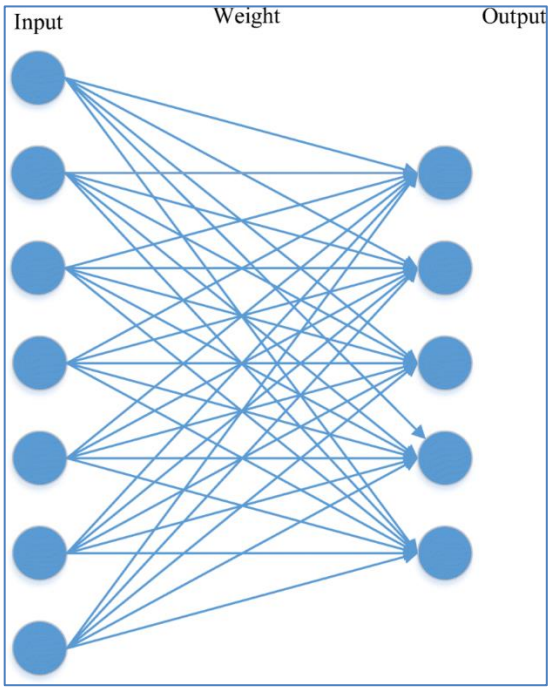


Figure (2.11): Connection Between convolution layer and Fully Connected Layer [9 9] .

4) Non-Linearity Layers

To introduce non-linearity into the activation map, non-linearity layers are typically placed immediately after the convolutional layer. There is a vast range of possible non-linear operations; some of the most common ones are outlined below[85].

- Sigmoid: is the mathematical form for the sigmoid nonlinearity. A real number is then transformed into a floating-point value.

$$f(x) = 1/1 + \exp(-x) \qquad (2.16)$$

- Rectified Linear Unit (ReLU): This function is found using the ReLU maximization formula f(x) =max (0, x). In other words, the threshold is initially set to zero before the activation is performed.
- In contrast to a flat ReLU, the slope of an activation function known as a Leaky ReLU is just slightly negative. In Eq. (2.17), the formal definition of Leaky ReLU:

$$f(x)_{LeakyReLU} = \begin{cases} x & if\ X > 0 \\ mx & if\ x \leq 0 \end{cases} \quad (2.17)$$

- The softmax function operates on an input vector by exponentiating each element and then normalizing the resulting values through division by the sum of all exponential values. This normalization step ensures that the output values fall within a range of 0 to 1, while also ensuring their collective sum equals 1, thereby rendering them suitable for probabilistic representation. Mathematically speaking, one can express the formula for computing the softmax function as In Eq. (2.18):

$$Softmax(x_i) = \frac{exp(x_i)}{\Sigma_j\, exp(x_j)} \quad (2.18)$$

Where x_i refers to an individual input value corresponding to a specific class. $\exp(x_i)$ represents the result obtained from applying the exponential function to x_i. The term $\text{sum}(\exp(x_j))$ denotes aggregation over all classes, measuring their respective exponential magnitudes.

5) Dropout Layer

Over fitting of the training dataset happens when all features are added to the fully connected layer. Over fitting occurs when a model performs exceptionally well on training data but fails to generalize to new data. To address this issue, a dropout layer is used during training to arbitrarily prune the network, deleting neurons and their connections [100]. Figure (2.12) depicts a dropout.

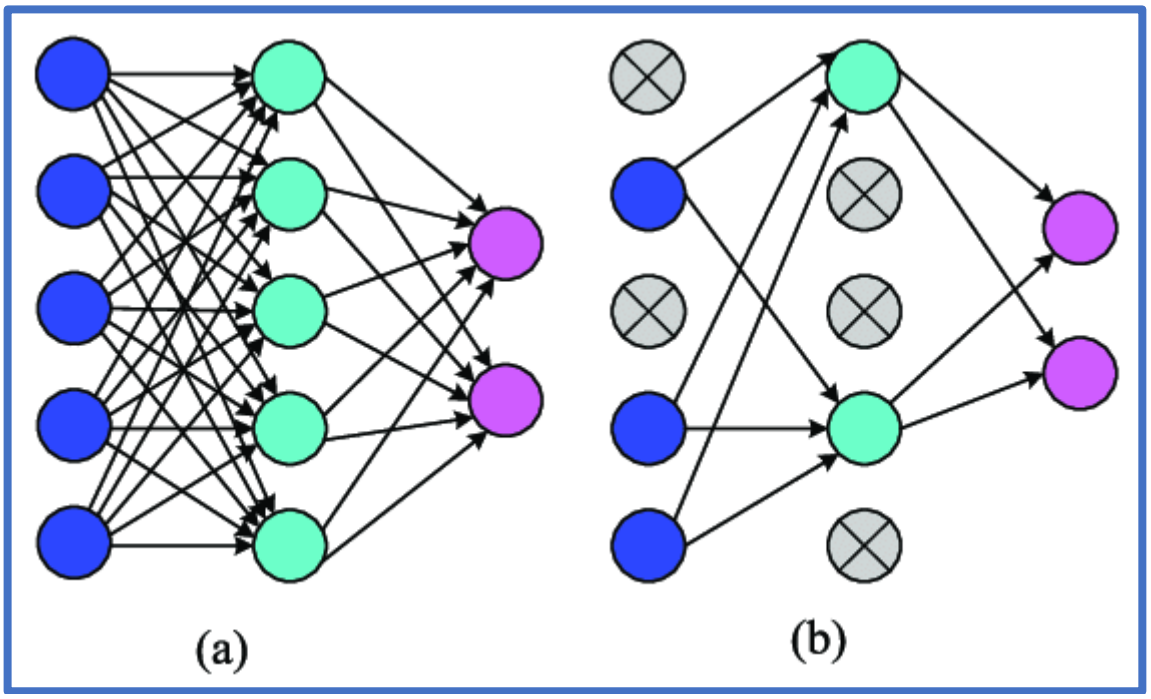


Figure (2.12) : Schematic comparison of (a) a regular neural network and (b) a neural network trained with Dropout[101]

6) Adam Optimization Algorithm

In contrast to stochastic gradient descent, Adam can make iterative changes to the network weights using only the training data. AdaGrad and RMS Prop are Adam's go-to tools for solving sparse gradients in noisy environments. Adam automatically adjusts the rate of learning for each parameter to speed up convergence and improve performance. Adaptive learning rates for each parameter are calculated by averaging their prior gradient and squared values. These equations let Adam fine-tune each parameter's learning rate by estimating the gradient's momentum and variance over time. Last update iteration [102], [103]:

$$\theta = \theta - \alpha * m / (\sqrt{v} + \varepsilon) \quad (2.19)$$

where the first and second moments are m and v, is the learning rate, and is a small constant to prevent division by zero. By dynamically modifying the learning rate of each parameter in response to its prior gradients, the Adam optimizer blends momentum-based optimization with adaptive learning rates to enhance the training of deep learning models[104].

2.7. Performance Evaluation

In order to train a machine learning model to its full potential, evaluation metrics are crucial. As such, choosing appropriate assessment criteria is a crucial step in differentiating and achieving the best model [60].

2.7.1. Confusion Matrix

Various metrics can be utilized to assess the effectiveness of specific classification algorithms from an academic standpoint. These include accuracy, f1-score, precision, and recall. The calculation of these measures is based on the computation of a confusion matrix - a table that summarizes the number of correctly or incorrectly predicted examples by a given classification model as depicted in Figure (2.13). Below are detailed descriptions for each value presented in this table[105]:

1. True positive (TP) refers to the correctly classified positive instances.
2. A false negative (FN) refers to instances where positive examples are inaccurately identified as negatives.
3. A false positive (FP) refers to negative instances that are mistakenly forecasted and categorized.
4. A true negative (TN) refers to instances that are correctly classified as negative by the model of classification.

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Figure (2.13): Confusion matrix [105].

2.7.2 Performance Metrices

A. **Accuracy**, or the degree to which a model is likely to accurately predict outcomes, is defined by the proportion of correct predictions relative to the total number of predictions, as shown in Eq. (2.20):

$$Accurcy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.20}$$

B. **Precision**: refers to how accurately a group of documents describes its subject, and thus how precisely they were classified. Class ci, symbolized by the symbol (Pi), has an accuracy that can be quantified as follows, as shown in Eq. (2.21):

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{2.21}$$

C. **Recall** measures how well a classifier can identify documents as belonging to a given class (as demonstrated by Eq. (2.22). Class ci recall (Ri) can be calculated using the formula:

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{2.22}$$

In this case, TP_i points to a true-positive value. FP_i stands for false positives and FN_i represents false negatives.

D. **F1 Measure**: is the precision-recall synchronization rate. Overall system performance is good if $F1$ is high. Given Eq (2.23), the following is a description of $F1$:

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{2.23}$$

Chapter Three

The Proposed System

3.1. Introduction

This chapter elaborates on the main vision of this thesis. It proposes a system for classifying gene expression data using machine learning and deep learning techniques. Section 3.2 describes the proposed system's block diagram. Section 3.2.1 illustrate dataset collection. Section 3.2.2 discusses the preparation steps for preparing the data for following phases. Then, during the feature selection stage, a subset of the total genes is chosen, as described in Section 3.2.3. Section 3.2.4 follows with classification models for machine learning and deep learning model. Finally, Section 3.3 discusses the proposed system's evaluation methodology.

3.2. The Proposed System Design

To achieve the goal of building a system for assessing genetic immune response and antibodies using machine learning, a five-stage strategy is provided. Dataset collection, data preprocessing, feature selection, classifier model creation, and evaluation are the names given to these components. The first stage, data preprocessing, entails dealing with missing values and normalizing the "GSE201535" dataset. The proposed system's second stage involves feature selection using approaches such as Mutual Information, Chi-Square, and Analysis of Variance. This stage is in charge of restricting the scope from the initial set of genes to the genes that are most important to vaccinations. Feature selection approaches are used sequentially, in parallel, and in a hybrid fashion. The goal is to discover the subset of genes that have the strongest link with vaccinations by using these feature selection approaches in various ways (sequential, parallel, or hybrid). This step is critical for lowering the dimensionality of gene expression data as well as boosting the efficiency and effectiveness of future classification algorithms. The proposed system's third level includes machine learning and deep learning classification models such as Random Forests, ADABoost, and Logistic Regression. Finally, each model is graded based on its performance metrics. Figure 3.1 depicts a schematic illustration of the proposed system. Algorithm (3.1) shows the detailed

stages for the suggested system.

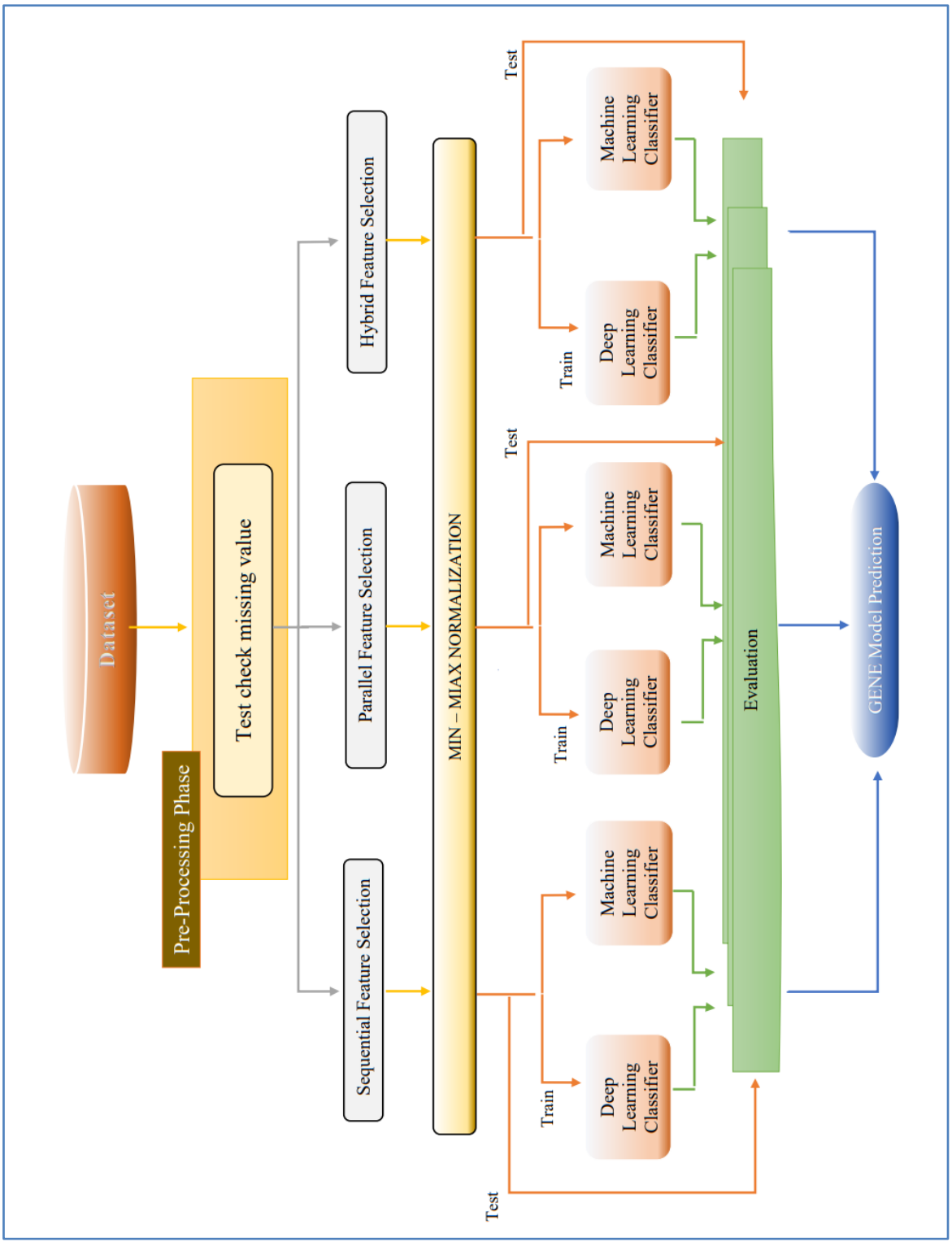


Figure (3.1): Block Diagram of the Proposed System

Algorithm (3.1) :The Proposed System

Input: Dataset D containing genetic information of individuals who received Covid-19 vaccines

Output: Selected informative genes

Stage 1. Dataset collected

Stage 2. Preprocess the dataset:

Step 1.Normalize the data to ensure consistent scaling according to Eq(2.1).

Step 2. Missing values using mean.

Stage 3. Feature Selection:

Step 1. Perform Parallel Feature Selection:

- **Apply the (MI, ANOVA, CHI^2) feature selection method to select informative genes according to Algorithm (2.1), (2.2) , and (2.3).**
- **Store the selected genes in a subset of data (SD).**

Step 2. Perform Sequential Feature Selection:

- **Apply sequential feature selection methods (e.g., MI_CHI^2_ANOVA, MI_ANOVA_CHI^2, CHI^2_MI_ANOVA, CHI^2_ANOVA_MI, ANOVA_MI_ CHI^2, and ANOVA_ CHI^2_MI) to identify relevant genes.**
- **Store the selected genes in the SD.**

Step 3. Perform Hybrid Feature Selection:

- **Combine feature selection methods (MI, ANOVA, CHI^2) with Random Forest method (MI_RF, ANOVA_RF, CHI^2_RF) to select informative genes.**
- **Store the selected genes in the SD.**

Stage 4. Machine Learning Classification:

Step 1. Split the SD into (70:30) training set TR and testing set TS.

Step 2 . Run one of the ML Algorithms listed below:

1. **Call Adaboost Algorithm (2.4) and evaluate the algorithm's performance.**
2. **Call Random Forest Algorithm (2.5) and evaluate the algorithm's performance .**
3. **Call Logistic Regression Algorithm using Eq. (2.12) and (2.13), then Evaluate the algorithm's performance.**

Stage 5. CNN Model:

Step 1. Build a Convolutional Neural Network (CNN) model.

Step 2. Train the model using TR.

Step 3. Evaluate the model's performance on TS.

Stage 6. Evaluate and Compare Results:

Step 1. Compare the performance of the different feature selection approaches and machine learning models.

Step 2. Analyze the results to determine the effectiveness of each approach in identifying informative genes related to vaccine response.

Step 3. Return the selected genes from the feature selection steps.

End

3.2.1. Dataset

The blood transcriptome data of 161 samples collected from the GEO database under registration number GSE201535 were analyzed in this work. Vaccinees are divided into two groups based on the dose of COVID-19 they

received, with group I receiving the first dose and group II receiving the second. Three subsets are provided within each group to represent different timepoints after vaccination. Subset I-D0 represents day zero following ChAdOx1 vaccination, subset I-D2-4 represents days two to four post-vaccination, and subset I-D7 represents day seven post-vaccine. Similarly, subset II-D0 represents day zero after BNT162b2 injection, whereas subsets II-D1-4 and II-D7-10 correspond to days one through four and days seven through ten, respectively.

3.2.2. Preprocessing Stage

Preprocessing is an essential step to prepare data for suitability with machine learning models. In the proposed system, preprocessing includes two fundamental steps: handling missing values and normalization.

3.2.2.1. Handling missing values:

The missing values are dealt with using a process called substitution, which involved replacing each one with a new value that was determined by calculating the mean of the gene column that is missing a value as in equation(2.1).

3.2.2.2. The Normalization:

The min-max normalization strategy is used to normalize the values of all genes, according to Equation (2.2) provided in Chapter 2 in the section on Normalization. The numeric values of the genes in the dataset that are used as input to the machine learning models have all been normalized so that they fall between zero and one. This guarantees that the models can read the data correctly. The algorithm provides further information about the procedure.

3.2.3. Feature Selection Stage

Due to the high dimensionality of the embedded features in the dataset, feature selection is required to reduce dimensionality and pick the most relevant characteristics associated to vaccinations, resulting in high-performance classification models. The system seeks to determine a subset of attributes that have the strongest link with vaccines by employing feature selection algorithms in parallel, sequential, or hybrid fashions which is depicted in Figure (3.2). This technique aids in the elimination of irrelevant or redundant elements, the reduction of computational complexity, and the enhancement of the efficiency and effectiveness of the subsequent process.

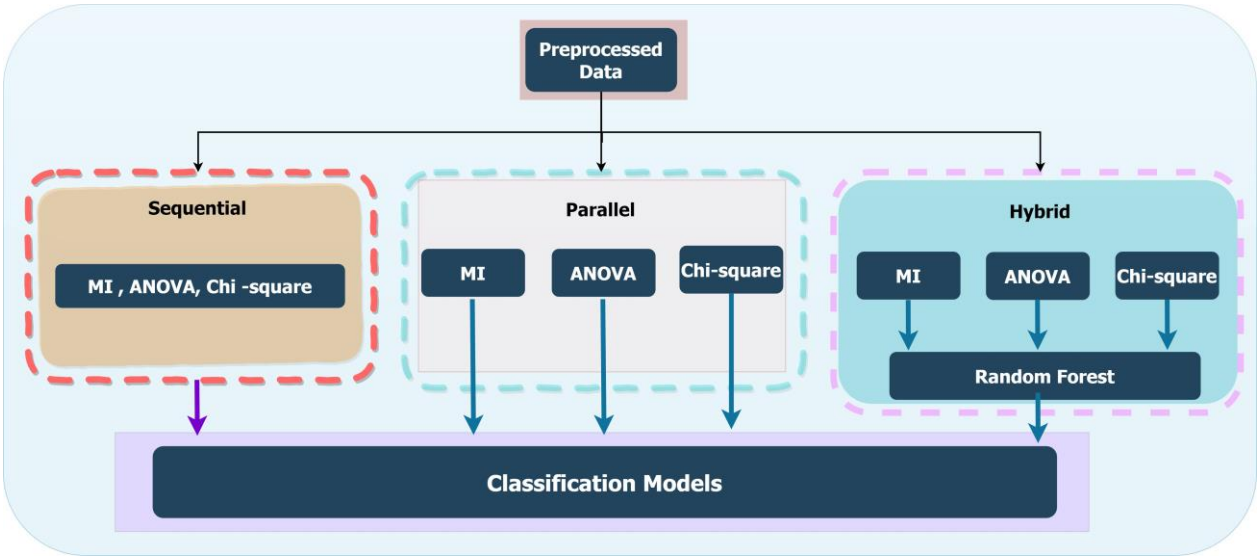


Figure (3.2): The Feature Selection Methods Parts.

Individual feature selection approaches are employed during the parallel feature selection stage to construct a subset of features that will subsequently serve as inputs to the classification model. A feature selection method is used in the sequential feature selection context, resulting in a subset of features that forms the input for another feature selection method. Feature selection is carried out in several sequences and utilizing various feature selection methods, with the results of each sequence ultimately serving as inputs to classification models. Individual

feature selection approaches, such as the filtering approach, are utilized during the hybrid feature selection stage to build a subset of features that serve as inputs to the Random Forest algorithm, which is used as an embedding strategy. The selected features from this hybrid approach are then utilized as inputs in the classification model.

3.2.3.1. Mutual Information Method

Mutual Information (MI) is a popular feature selection metric in gene expression data classification. It establishes the statistical relationship or shared information between genes and class labels. The mutual information between each gene (feature) and the class labels is determined using the equation (2.5) shown in section (2.6.1) of chapter two, which shows the Mutual Information formula. The genes are then rated according to their mutual information scores, with higher scores indicating a stronger link between gene expression and class labels. It also chooses the best features based on a predefined criterion (percentage). Figure (3.3) depicts the mutual information strategy.

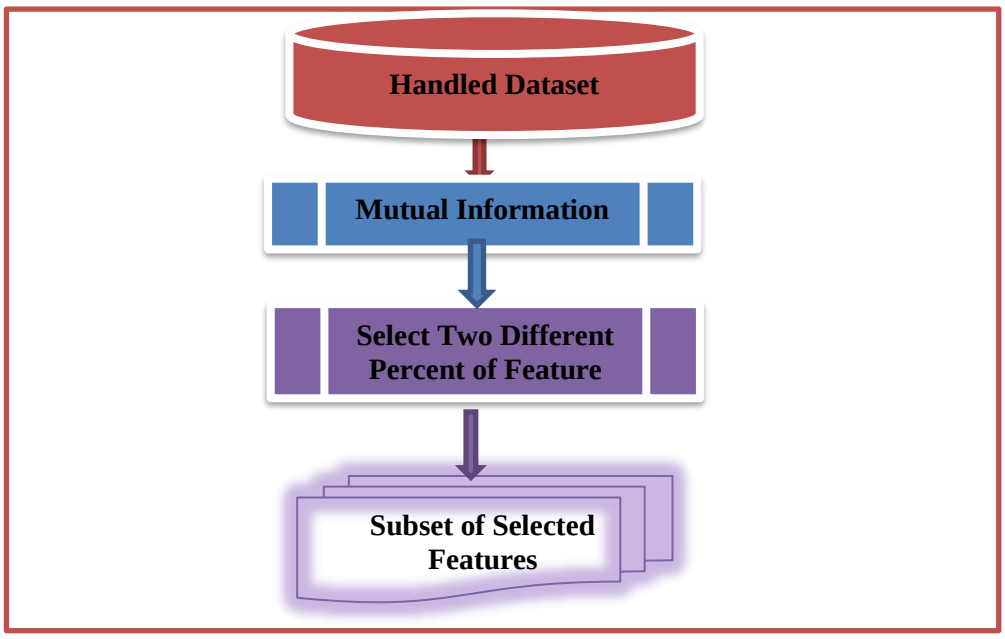


Figure (3.3): The Mutual Information Feature Selection Process.

3.2.3.2. Analysis of Variance Method

ANOVA is a statistical method for identifying advantages in genetic expression data. ANOVA identifies genes with significant expression differences between these groups. The ANOVA is calculated for each gene to evaluate genetic expression variation across groups. The F-Statistics Equation (2.10), which is mentioned in section (2.6.2) of Chapter 2, is used to rank the results of the ANOVA study. P- value is chosen less than 0,001 to choose the important features [58]. Genes with higher F-statistics show significant differences in expression levels across groups and are therefore more relevant for classification. Finally, the top-ranking characteristics are chosen using a predetermined criterion. Figure (3.4) depicts the ANOVA approach.

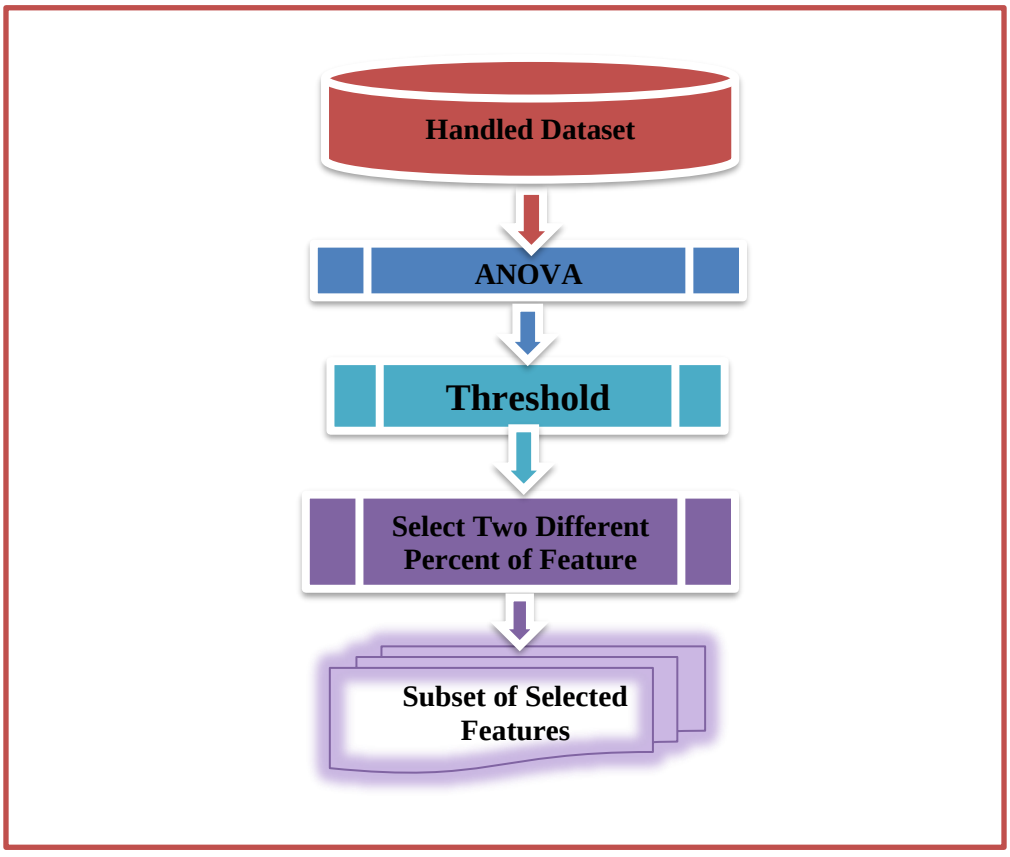


Figure (3.4): Block Diagram of Analysis of Variance Method.

3.2.3.3. Chi-Square Method

In gene expression data, the chi-square (2) test is used to examine the

statistical significance of the association between a feature (gene) and the target variable (class label). The chi-square test is applied to each gene using the equation (2.11) stated in Chapter 2 section (2.6.3). The gene is considered statistically significant if the estimated chi-square value for it exceeds the critical value. Genes are prioritized according to their chi-square statistics, with higher chi-square values deemed more informative and related to the target variable. Finally, based on the predetermined criteria (%), the top-ranked characteristics are chosen, and the other genes are discarded from further research. As illustrated in Figure (3.5).

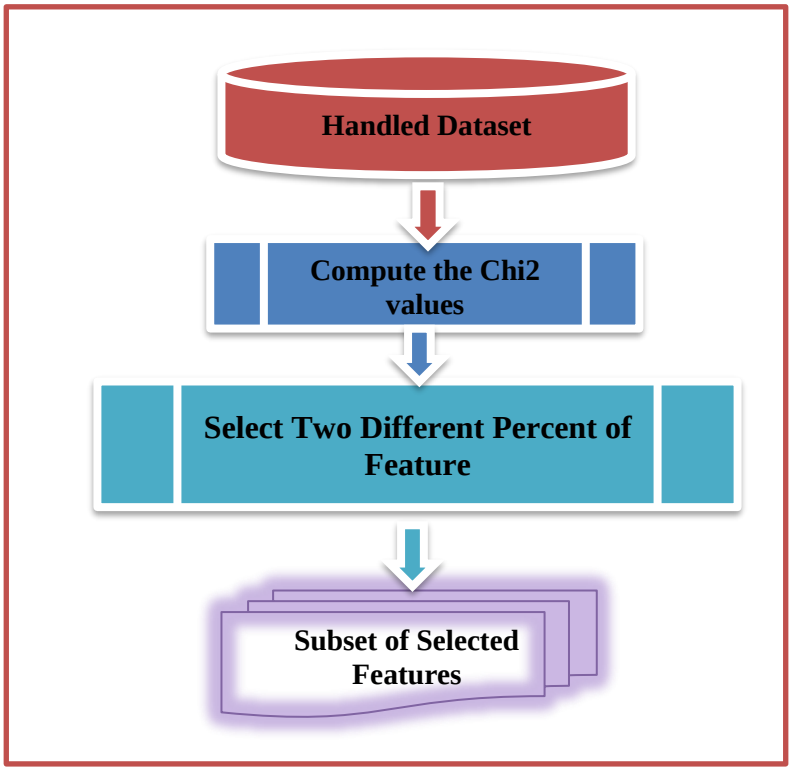


Figure (3.5): Block Diagram of chi-square Method.

3.2.3.4. Random Forest Method

The Random Forest machine learning method may effectively perform feature selection and classification tasks. Using the "get_ support" function, each feature in Random Forest is allocated a score based on its importance. This function returns an array of Boolean values, where True represents features with more importance than the mean importance and False represents the remaining

features. The Random Forest approach is depicted in Figure (3.6).

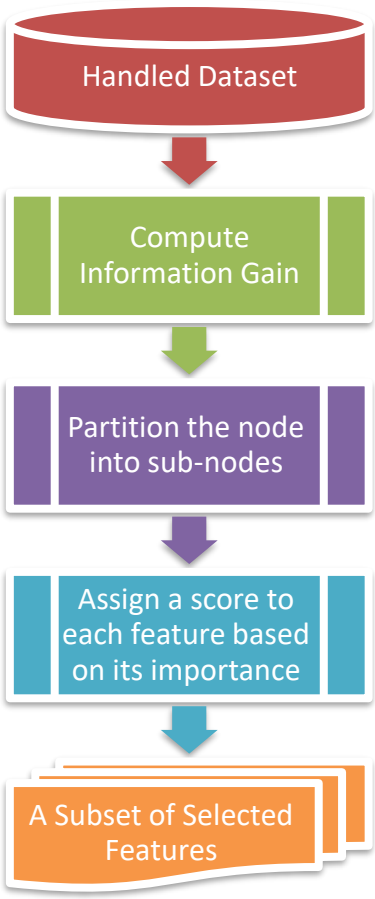


Figure (3.6): Block Diagram of Random Forest Method.

3.2.4. Machine Learning Classification Stage

Machine learning models are used in this study for two different purposes: classification and feature selection, to identify the most important genes associated with vaccines. Three of the machine learning models, namely Random Forests, ADABoost, and the Logistic Regression Machines model, are utilized to achieve the classification of the gene dataset. Figure (3.7) displays the block design for machine learning classification models.

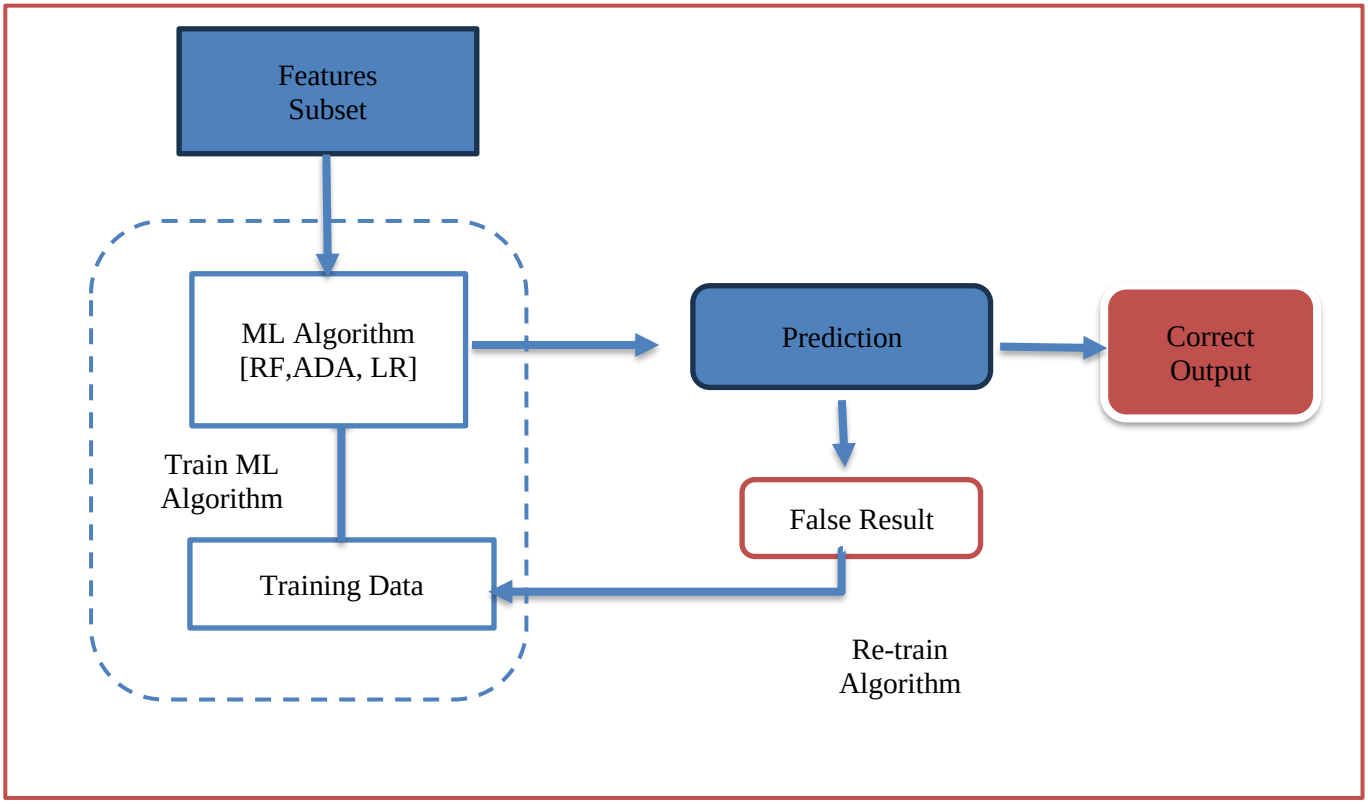


Figure (3.7): A block diagram of the machine learning classification models.

3.2.4.1. Random Forest

Each random forest tree independently predicts the class label for each test set instance according to Algorithm (2.5). To determine the instance's forecast, each tree "votes" for a class label. The instance's prognosis is determined by the most popular class label. Figure (3.8) shows a random forest classification model.

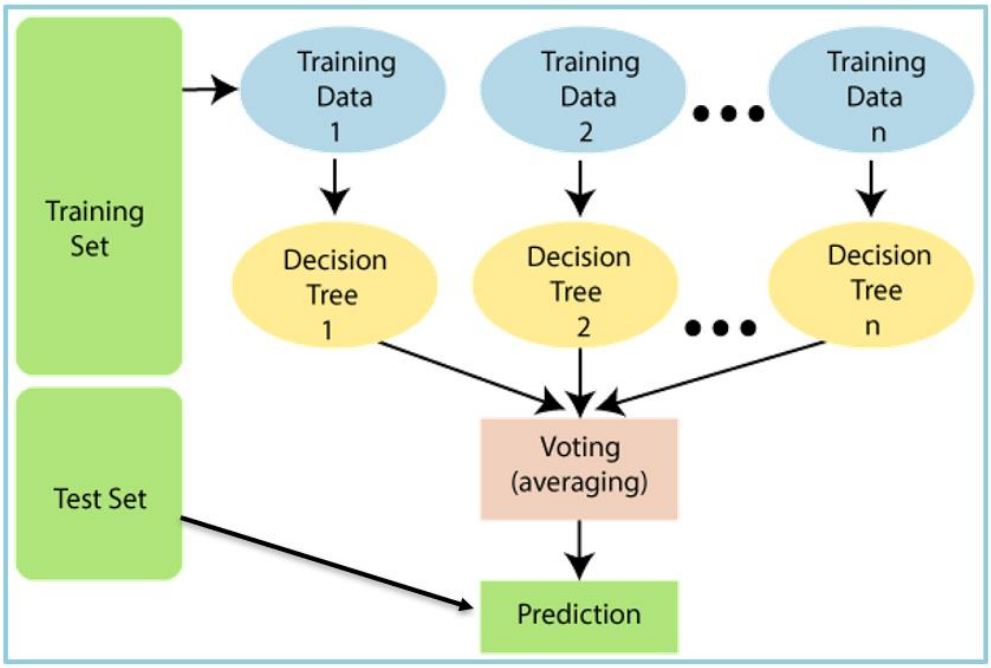


Figure (3.8): Schematic representation of the Random Forest Classification Model.

3.2.4.2. Logistic Regression

To grade a gene's input, logistic regression employs a linear predictor function, applying the logistic function (also known as the sigmoid function) to this score yields the class likelihood of a gene. The logistic regression equation (2.12) and (2.13) uses input features and coefficients to represent this change. Logistic regression optimizes these coefficients iteratively to generate accurate predictions. Negative log-likelihood minimization (also known as the cross-entropy loss) is a common loss function. Logistic regression identifies genes in feature space by modifying coefficients iteratively to minimize loss. Figure (3.9) shows a Logistic regression classification model.

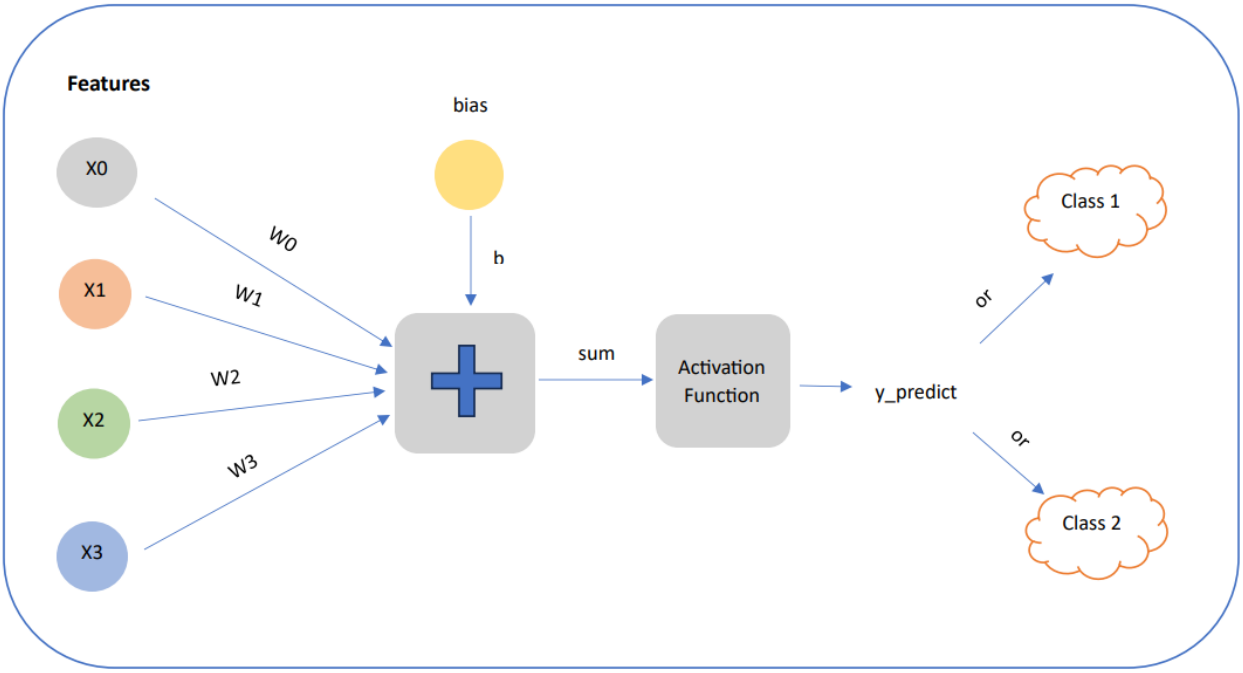


Figure (3.9): Schematic representation of the Logistic Regression Classification Model.

3.2.4.3. Adaboost

On the original dataset, the AdaBoost meta-estimator trains a classifier. Following that, the classifier is trained many times on the same dataset, with updated weights for incorrectly classified cases. With this weight shift, classifiers can prioritize difficult instances according to Algorithm (2.4). Because the algorithm gives misclassified observations more weight, they have a better chance of being correctly classified in subsequent rounds. A Logistic regression classification model is depicted in Figure (3.10).

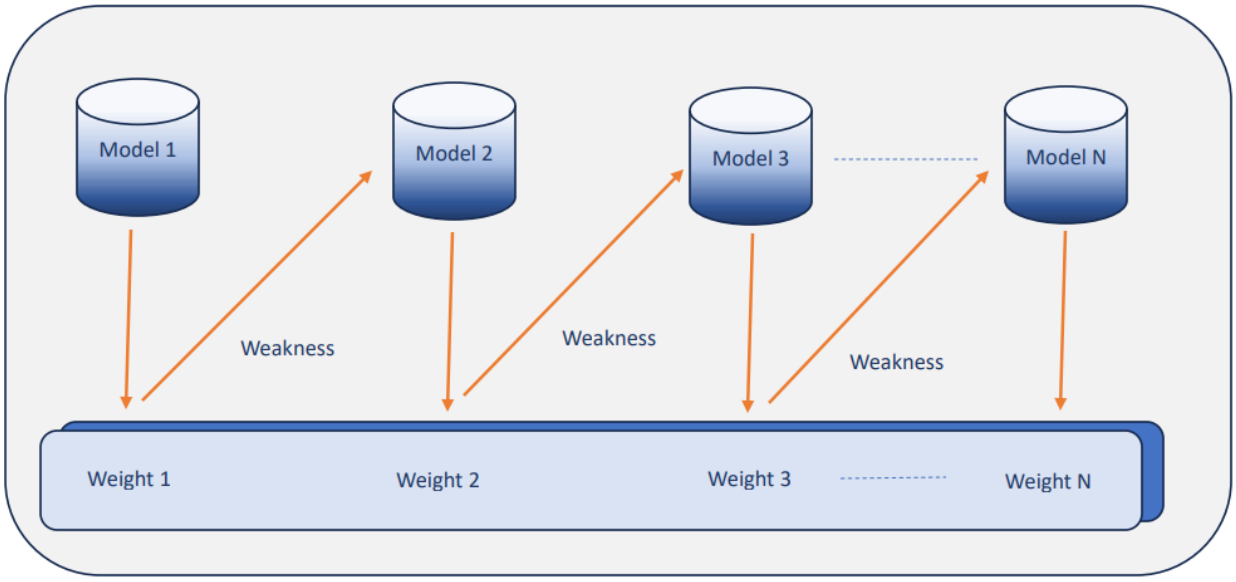


Figure (3.10): Schematic representation of the Adaboost Classification Model.

3.2.5. Deep Learning Classification Stage

In the fast expanding and complicated field of deep learning, our suggested solution was meant to go above and beyond existing capabilities by leveraging more efficient and powerful algorithms while also improving feature extraction capabilities. The goal is to extract extraordinary levels of accuracy and insights from data that were previously unattainable without such cutting-edge tools.

3.2.5.1. One-Dimensional Convolutional Neural Network (1D-CNN) Model

The CNN algorithm is introduced as a contribution to gene expression research. The training dataset is separated into 70% for training and 30% for testing to discover vaccine-related characteristics. The deep neural model is trained, and its performance is measured using four metrics of accuracy.

The experimental results on the gene expression dataset showed that the one-dimensional convolutional neural network (1D-CNN) model outperformed the other topologies investigated. The chosen input data considerably improved the

effectiveness of the chosen 1D-CNN model, yielding satisfactory results and better prediction performance. Figure (3.11) shows the structure of the CNN model that was used:

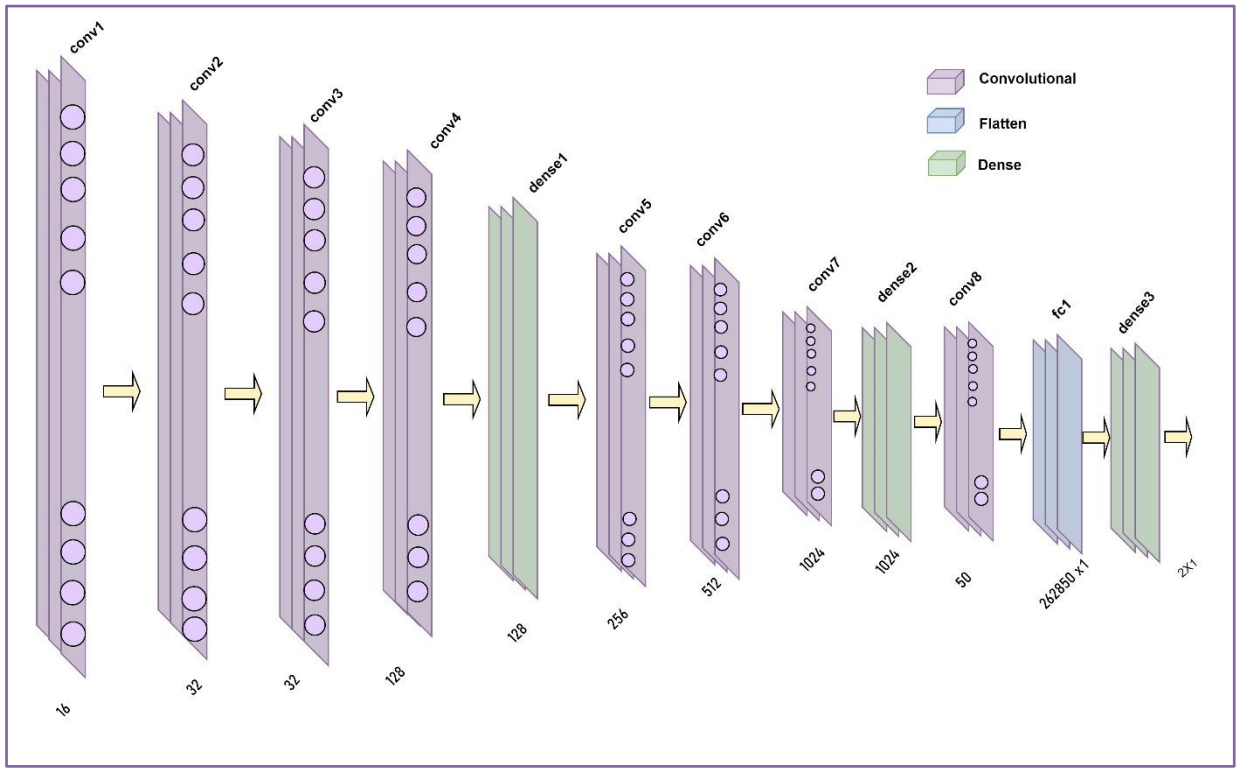


Figure (3.11): 1D CNN classification model layers .

The model being illustrated in Figure (3.11) is a Convolutional Neural Network (CNN) structure that has been specifically designed for a particular task, specifically focusing on extracting features related to the impact of vaccines. It consists of multiple layers, each with its own distinct function in the process of extracting features and performing classification.

- The model originates with a sequence of convolutional layers, namely conv1d_1, conv1d_2, conv1d_3, and conv1d_4, which are tasked with identifying pertinent patterns and characteristics within the input data. The convolutional layers employ varying filter sizes and quantities of filters to extract a wide range of significant features from the input data. The convolutional layers in this study employ the Leaky Rectified Linear Unit

(Leaky ReLU) as the activation function. This choice of activation function introduces non-linearity to the output, thereby augmenting the model's ability to acquire intricate representations.

- Following each convolutional layer, the feature maps undergo a series of max-pooling layers, specifically max_pooling1d_1, max_pooling1d_2, max_pooling1d_3, and max_pooling1d_4, in order to decrease the spatial dimensions of the feature maps. Max-pooling is a technique that facilitates the reduction of feature map dimensions while preserving the most salient information. This process enhances computational efficiency and mitigates the potential for overfitting within the model.

- After the application of the convolutional and max-pooling layers, the subsequent layer, known as dense_1, is incorporated into the model architecture. This layer functions as a fully connected layer, consisting of 128 units. The dense layer serves the purpose of further processing the features that have been extracted from the preceding layers, thereby enabling the acquisition of more sophisticated and abstract representations.

- Subsequently, the model proceeds with the inclusion of supplementary convolutional layers, namely conv1d_5, conv1d_6, and conv1d_7, along with their corresponding max-pooling layers, denoted as max_pooling1d_5, max_pooling1d_6, and max_pooling1d_7. The purpose of these layers is to extract more intricate and abstract features from the data, thereby enhancing the model's capacity to detect complex patterns.

- The dense_2 layer is an additional fully connected layer comprising 1024 units, which serves to further enhance the acquired features and facilitate their utilization in the ultimate classification task.

- The model completes with the dense_3 layer, comprising 2 units that correspond to the output classes of the classification task. The softmax activation function is frequently employed in the final layer of a convolutional neural network (CNN) for the purpose of multi-class classification. This function transforms the raw output scores into probability distributions, thereby facilitating the interpretation and comparison of the model's predictions.

The model contains a total of 3,828,792 parameters, all of which are trainable as shown in table (3.1). This implies that these parameters are adjusted during the training phase in order to optimize the model's performance for the specified task. The architecture of the model and the arrangement of its layers are designed with the objective of efficiently acquiring and representing the inherent patterns within the input data. This characteristic renders it a potent instrument for the particular classification task under consideration.

Table (3.1): 1D CNN classification model layers.

Layer (type)	**Output Shape**	**Param #**
conv1d_1 (Conv1D)	(None, 5271, 16)	64
leaky_re_lu_1 (LeakyReLU)	(None, 5271, 16)	0
max_pooling1d_1 (MaxPooling1)	(None, 5271, 16)	0
conv1d_2 (Conv1D)	(None, 5269, 32)	1568
leaky_re_lu_2 (LeakyReLU)	(None, 5269, 32)	0
max_pooling1d_2 (MaxPooling1)	(None, 5269, 32)	0
conv1d_3 (Conv1D)	(None, 5267, 32)	3104
leaky_re_lu_3 (LeakyReLU)	(None, 5267, 32)	**0**
max_pooling1d_3 (MaxPooling1)	(None, 5267, 32)	**0**
conv1d_4 (Conv1D)	(None, 5265, 128)	12416

leaky_re_lu_4 (LeakyReLU)	(None, 5265, 128)	**0**
leaky_re_lu_5 (LeakyReLU)	(None, 5265, 128)	**0**
max_pooling1d_4 (MaxPooling1)	(None, 5265, 128)	**0**
dense_1 (Dense)	(None, 5265, 128)	16512
conv1d_5 (Conv1D)	(None, 5263, 256)	98560
leaky_re_lu_6 (LeakyReLU)	(None, 5263, 256)	**0**
max_pooling1d_5 (MaxPooling1)	(None, 5263, 256)	**0**
conv1d_6 (Conv1D)	(None, 5261, 512)	393728
leaky_re_lu_7 (LeakyReLU)	(None, 5261, 512)	**0**
max_pooling1d_6 (MaxPooling1)	(None, 5261, 512)	**0**
conv1d_7 (Conv1D)	(None, 5259, 1024)	1573888
leaky_re_lu_8 (LeakyReLU)	(None, 5259, 1024)	**0**
max_pooling1d_7 (MaxPooling1)	(None, 5259, 1024)	**0**
dense_2 (Dense)	(None, 5259, 1024)	1049600
conv1d_8 (Conv1D)	(None, 5257, 50)	153650
flatten_1 (Flatten)	(None, 262850)	**0**
dense_3 (Dense)	(None, 2)	525702
Total params:	3,828,792	

3.3. Evaluation of the Proposed Stages

The suggested system's final stage is model evaluation. This stage seeks to increase the assessment's reliability and is used to measure the success of the machine learning (ML) model and the Convolutional Neural Network (CNN). The test dataset is used to assess the model's performance quality. The performance of the model is assessed by training it on the gene expression dataset and then testing it on the same dataset. The proposed model performs several training iterations while maintaining rigorous loss and accuracy controls in the training and validation sets. After each iteration of the suggested model, the accuracy of the training and validation data is impacted.

Metrics such as accuracy, recall, precision, and loss function are calculated as in Eq.(2.19) Eq.(2.23) and play a significant role in confirming the validity of the conclusions and the reliability of the model used.

Chapter Four
Experimental Results and Discussion

4.1 Introduction

This chapter outcomes and discussions of the proposed system are covered here. The primary purpose of this study was to evaluate how different vaccines alter the antibody response and genetic immune system using cutting-edge deep learning and machine learning approaches. The classification algorithms Adaboost, Random Forest, Logistic Regression, and Convolutional Neural Network (CNN) are used to investigate and make conclusions from the GSE201535 dataset. Three feature selection procedures to improve the models' prediction ability are used: sequential, parallel, and hybrid based on mutual information, ANOVA, and the CHI-squared test. The sections that follow present the thesis findings in great detail.

4.2. The Proposed System Requirement

The proposed system is coded in Python, specifically Python 3.6.5 with the essential Python libraries. The operating system is Windows 10 in 64-bit mode. The computing power is provided by an Intel Core i3 3120M processor running at 2.50 GHz. There is 4 GB of RAM available.

4.3. GSE201535 Dataset Description

Data from people who have been immunized against various diseases are combined into the GSE201535 dataset [20], which may be accessed through the Gene Expression Omnibus (GEO) database at the National Center for Biotechnology Information (NCBI). To ensure that the machine learning models produce the best outcomes possible. The genetic dataset used in this investigation was generated from individuals who were inoculated with different Covid-19 vaccinations at different time intervals. The dataset used in this study is GSE201535, which contains gene expression data from 161 blood samples. These samples were taken from a group of 15 people who had received the BNT162b2 vaccination. The blood of each participant is drawn three times

before vaccination, between 2 to 4 days post-vaccination, and between 7 to 10 days post-vaccination. Additionally, for the ChAdOx1 vaccine, gene data is obtained from another set of individuals (39 in total) on three occasions within the subsequent timeframe of 2 to 7 days after vaccination.

Figure (4.1) depicts each text file as a record of a blood sample to represent the temporal changes in gene expression. Which has two columns. The first column contains the name of the gene. The second column contains the level of the gene's biomarker in the blood sample. Within their distinct datasets, the expression data for each blood sample is saved in individual text files compressed as.zip format. To enable further analysis, it is essential to extract and arrange this data into a structured tabular format who are split into six groups according to the dose and timing of their inoculations (I-D0, I-D2-4, I-D7 for first-dose ChAdOx1 at day 0, day 2, and day 7; II-D0, II-D1-4, II-D7-10 for second-dose BNT162b2 at day 0, day 2, and day 7).

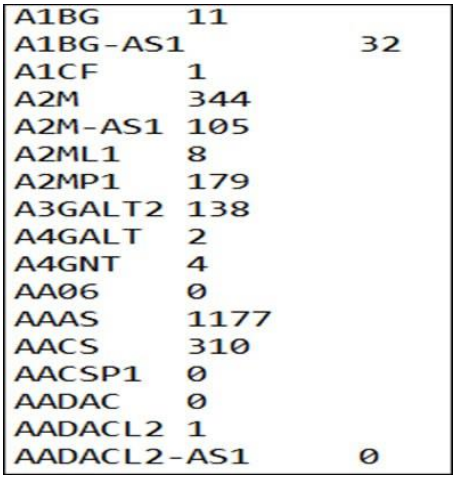


Figure (4.1): A sample of the original gene data in txt file

As illustrated in Figure (4.2), each file name contains critical information such as the serial number of the experiment (the session of blood drawn), vaccine name, whether the vaccine dose is the first or second dose, person ID, person genes and date of blood draw.

id_num	experiment	vaccine	Person_id	I_or_II	Day	A1BG	A1BG-AS1	A1CF	A2M	A2M-AS1	A2ML1	A2MP1	A3GALT2	A4GALT	A4GNT	AA06
0	GSM6066188	BNT	13	II	D4	32	44	3	94	42	5	46	22	0	0	
1	GSM6066187	BNT	13	II	D0	37	61	2	321	56	2	92	12	1	0	
2	GSM6066189	BNT	13	II	D7	39	56	1	98	49	0	55	6	0	0	
3	GSM6066194	BNT	23	II	D4	28	51	6	312	92	7	81	24	7	5	
4	GSM6066192	BNT	17	II	D7	28	38	5	106	37	0	50	11	2	0	
5	GSM6066193	BNT	23	II	D0	21	32	4	125	37	4	45	24	3	0	
6	GSM6066190	BNT	17	II	D0	122	72	4	372	58	8	96	24	2	2	
7	GSM6066191	BNT	17	II	D3	9	71	5	260	50	1	60	8	0	2	
8	GSM6066196	BNT	24	II	D0	15	55	8	165	88	7	86	17	0	2	
9	GSM6066198	BNT	24	II	D9	19	50	3	367	79	13	89	34	0	6	
10	GSM6066197	BNT	24	II	D2	17	45	2	148	60	3	48	16	1	1	
11	GSM6066199	BNT	25	II	D0	39	57	1	140	36	6	64	23	0	0	
12	GSM6066195	BNT	23	II	D7	27	59	11	128	60	3	65	17	8	3	
13	GSM6066202	BNT	26	II	D0	23	41	6	129	40	15	29	14	4	2	
14	GSM6066201	BNT	25	II	D7	23	60	3	172	69	4	76	17	2	0	
15	GSM6066200	BNT	25	II	D1	33	62	9	244	33	2	56	19	0	0	
16	GSM6066203	BNT	26	II	D4	22	51	7	149	39	8	39	11	2	0	
17	GSM6066205	BNT	27	II	D0	20	46	5	271	75	2	64	17	0	0	
18	GSM6066207	BNT	27	II	D7	15	63	9	170	106	9	93	13	0	0	
19	GSM6066204	BNT	26	II	D8	4	43	4	89	32	1	19	11	4	0	
20	GSM6066206	BNT	27	II	D3	10	52	5	196	81	6	78	20	0	0	
21	GSM6066208	BNT	36	II	D0	24	52	17	539	73	1	133	4	0	3	
22	GSM6066210	BNT	36	II	D10	27	57	11	112	67	4	107	18	2	3	
23	GSM6066209	BNT	36	II	D3	19	48	1	192	68	7	77	15	3	2	

Figure (4.2): Dataset after reading process.

Only four participants were given a second dose of ChAdOx1, while the majority were given BNT162b2. The details of the GSE201535 dataset are shown in Table (4.1). Figure (4.3) depicts the description of class numbers in the data.

Table (4.1): Details of the GSE201535Dataset.

Title of dataset	GSE201535 dataset
Number of classes	2
Number of instances	161
Number of genes	26, 364

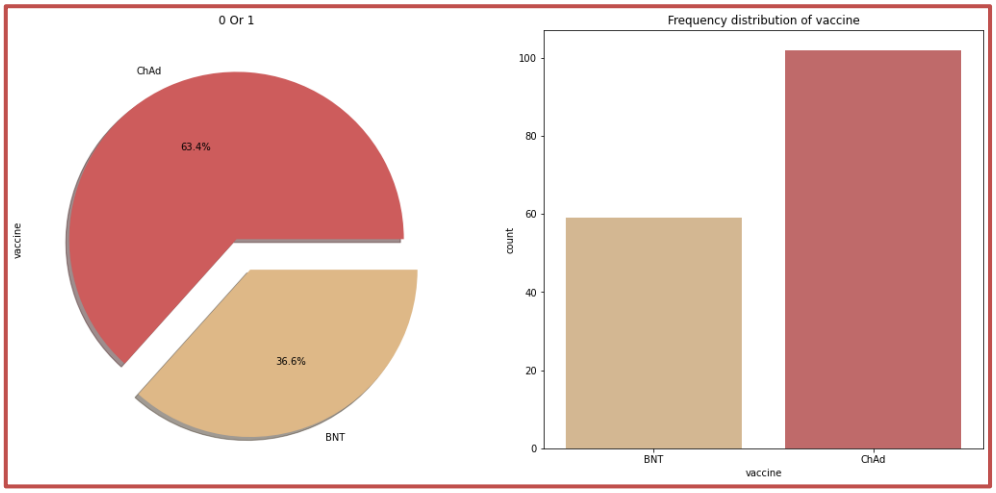


Figure (4.3): Number of Classes in the GSE201535Dataset.

4.4. Results of Preprocessing Stage

In order to simplify the process of applying classification models to the GSE201535 Dataset, it is necessary to perform a pre-processing step that consists of two separate steps.

4.4.1. Handling missing values:

The dataset had no missing values, so analysis did not involve incomplete or null data. The database's completeness and accuracy helped the investigation without the need for additional data recovery. Each sample had gene values for all genes. A complete dataset allowed us to use all gene-related information, yielding reliable, accurate results for this study.

4.5. ML Classification Stage Results

Because there are so many genes, gene expression data has a high dimension. This is one of the data's qualities. To address this issue, it is vital to identify the genes that contribute the most to the vaccine and to eliminate the genes that are unneeded. This thesis used three distinct feature selection methods to identify a subset of the GSE201535 dataset that contains the genes that are thought to be the most important in relation with. These strategies were implemented through parallel, sequential, and hybrid processes, respectively.

4.5.1. Parallel Feature Selection Part

This section seeks to provide a thorough summary of the findings of our parallel feature selection procedure. The primary goal is to greatly reduce the dimensionality of our dataset while retaining its integrity by employing three unique ways for producing subsets that are subsequently used as inputs into various classification models. To effectively achieve this goal, we first analyzed the data using three popular feature extraction methods: mutual information-based approach, analysis-of-variance technique, and chi-square test. From there, each created output served as a viable input choice in selecting relevant classifiers required for making correct predictions on fresh

instances.

4.5.1.1. The Result of the Parallel Feature Selection

Three feature selection techniques to accurately predict vaccination efficacy are employed: Analysis of Variance, CHI^2, and Mutual Information. Each method revealed the most important characteristics for predicting vaccine effectiveness. As indicated in Table (4.2), the approaches independently picked 20% (5,273 features) and 60% (15,818 features) from a pool of initial sets having 26, 364 characteristics. The Mutual Information technique prioritized distinct characteristics with significant information gain, suggesting a high association with vaccine effectiveness prediction. Various feature selection techniques are used to get a collection of key variables that may differ considerably between efficacy categories and distinguish between groups. Among these methods, the ANOVA method is used to discover variables with statistical significance, whilst the CHI^2 method is used to pick critical traits for correct vaccine efficacy classification. This method improves analysis efficiency by focusing on specific traits thought to be crucial in facilitating classification accuracy through discriminative ability among different groups within the supplied data set.

Table (4.2): Number of Genes Selected using Parallel Feature Selection.

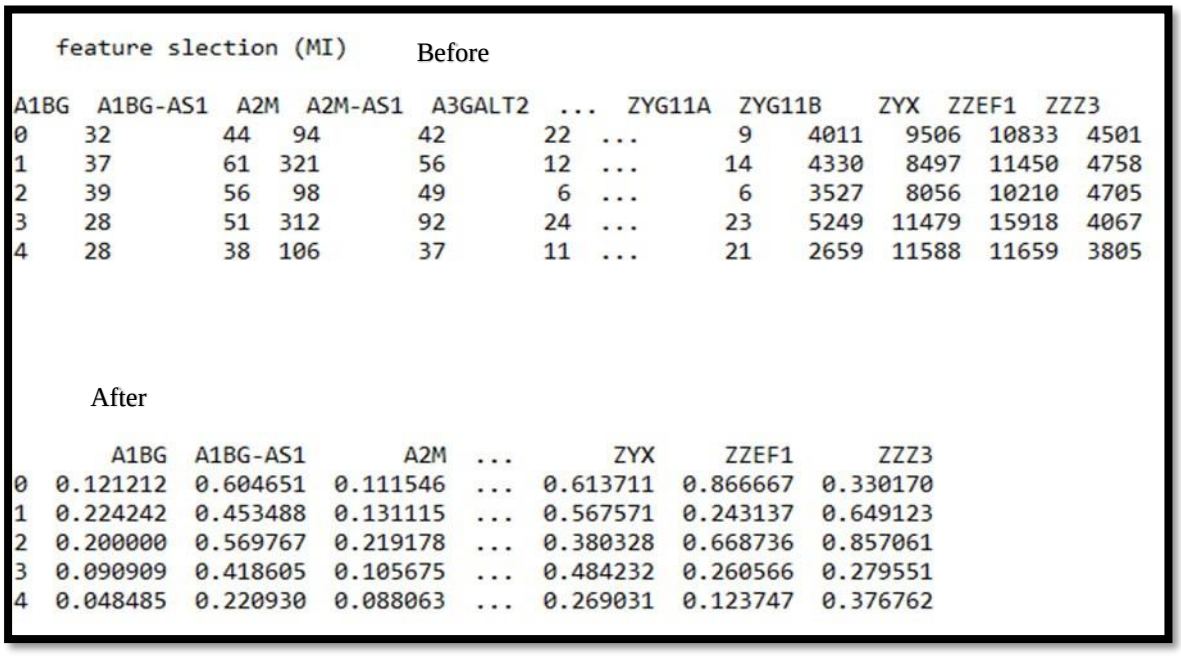
Feature Selection methods	The Number of Selected Genes		
	Percent	Original	Reduced
MI ***CHI^2*** ***ANOVA***	20%	26, 364	5,273
	60%	26, 364	15,818

4.5.1.2. The Result of the Normalization

Normalization is critical in ensuring that all features selected during the parallel feature selection stage are on a comparable scale. This stage is critical to ensuring that no single feature dominates the analytical process. The Min-Max normalization technique on the dataset in this study is used, utilizing the equation (2.1) provided in

Chapter Two, Section 2.3.2. This normalization method to convert the feature values to a predefined and confined range is used, often ranging from 0 to 1. This normalizing technique is especially useful and practical for classification problems since it provides for fair comparison and eliminates any potential bias caused by the original feature scales.

Figure 4.4 shows the dataset before and after Min-Max normalization for MI(20%) to demonstrate the influence of normalization. The dataset is shown in Figure 4.5 before normalization, and the feature values show significant variation in terms of magnitude and range for the ANOVA (20%). Figure 4.6, on the other hand, depicts the dataset after normalization, with all feature values scaled for the CHI^2 (20%). This normalization technique ensures that each parallel feature selection stage feature contributes equally to the analysis, removing any potential biases created by the original feature scales. Furthermore, normalization improves analysis accuracy by ensuring uniform and standardized feature representation.



feature slection (MI) Before

	A1BG	A1BG-AS1	A2M	A2M-AS1	A3GALT2	...	ZYG11A	ZYG11B	ZYX	ZZEF1	ZZZ3
0	32		44	94	42	22	...	9	4011	9506	10833	4501
1	37		61	321	56	12	...	14	4330	8497	11450	4758
2	39		56	98	49	6	...	6	3527	8056	10210	4705
3	28		51	312	92	24	...	23	5249	11479	15918	4067
4	28		38	106	37	11	...	21	2659	11588	11659	3805

After

	A1BG	A1BG-AS1	A2M	...	ZYX	ZZEF1	ZZZ3
0	0.121212	0.604651	0.111546	...	0.613711	0.866667	0.330170
1	0.224242	0.453488	0.131115	...	0.567571	0.243137	0.649123
2	0.200000	0.569767	0.219178	...	0.380328	0.668736	0.857061
3	0.090909	0.418605	0.105675	...	0.484232	0.260566	0.279551
4	0.048485	0.220930	0.088063	...	0.269031	0.123747	0.376762

Figure (4.4): Normalization of the data using the min-max algorithm for MI (20%) .

feature slection (ANOVA) Before

	A1BG-AS1	A1CF	A2M	A2M-AS1	AAAS	...	ZXDB	ZXDC	ZYG11B	ZZEF1	ZZZ3
0	44	3	94	42	1814	...	4567	4483	4011	10833	4501
1	61	2	321	56	1632	...	6179	5574	4330	11450	4758
2	56	1	98	49	2500	...	4471	4303	3527	10210	4705
3	51	6	312	92	2604	...	5266	6740	5249	15918	4067
4	38	5	106	37	2058	...	3596	4829	2659	11659	3805

After

	A1BG-AS1	A1CF	A2M	...	ZYG11B	ZZEF1	ZZZ3
0	0.604651	0.235294	0.111546	...	0.155791	0.866667	0.330170
1	0.453488	0.000000	0.131115	...	0.334149	0.243137	0.649123
2	0.569767	0.058824	0.219178	...	0.784122	0.668736	0.857061
3	0.418605	0.470588	0.105675	...	0.283578	0.260566	0.279551
4	0.220930	0.058824	0.088063	...	0.337412	0.123747	0.376762

Figure (4.5): Normalization of the data using the min-max algorithm for ANOVA (20%).

Chi2 feature slection Before

	A1BG-AS1	A2M	A2M-AS1	AAAS	AACS	...	ZXDC	ZYG11B	ZYX	ZZEF1	ZZZ3
0	44	94	42	1814	504	...	4483	4011	9506	10833	4501
1	61	321	56	1632	519	...	5574	4330	8497	11450	4758
2	56	98	49	2500	504	...	4303	3527	8056	10210	4705
3	51	312	92	2604	545	...	6740	5249	11479	15918	4067
4	38	106	37	2058	555	...	4829	2659	11588	11659	3805

After

	A1BG-AS1	A2M	A2M-AS1	...	ZYX	ZZEF1	ZZZ3
0	0.604651	0.111546	0.210526	...	0.613711	0.866667	0.330170
1	0.453488	0.131115	0.644737	...	0.567571	0.243137	0.649123
2	0.569767	0.219178	0.263158	...	0.380328	0.668736	0.857061
3	0.418605	0.105675	0.000000	...	0.484232	0.260566	0.279551
4	0.220930	0.088063	0.328947	...	0.269031	0.123747	0.376762

Figure (4.6): Normalization of the data using the min-max algorithm for CHI^2 (20%).

4.5.1.3. The Result of the ML Classification Model

Following the selection of relevant features via parallel feature selection, the normalized dataset features were entered into three models for classification: Random Forest, Adaboost, and Logistic Regression. These models aided in the analysis of data patterns for forecasting vaccination efficacy based on selected variables, allowing researchers to get insights into model correctness and performance.

In the classification phase, the Random Forest method is used to assess the precision of forecasting several categories of vaccine efficacy using selected attributes from the parallel feature selection step and a standardized dataset. For increased performance, the classifier employs an ensemble learning strategy that incorporates knowledge shared by 100 trees with no fixed maximum depth value. As a result, this collective strategy makes accurate forecasts possible. The next step is to use the Adaboost algorithm to see how reducing the number of features affects classification accuracy. Adaboost is a method of ensemble learning that combines numerous weak learners to generate a robust classifier. To achieve this, we adopt the Adaboost classifier with a maximum of 50 estimators (n estimators), which considers the reduced feature set achieved from parallel feature selection and normalization stages. The main goal is to appraise model performance in terms of precision, recall, F1-score as well as overall accuracy.

Similarly, the Logistic Regression technique is used to study how feature reduction affects classification performance. Following the preliminary stage of feature selection, which is carried out in parallel and utilizing the normalized dataset, selected features are used to train a model using logistic regression. Following that, the generated model is tested on various datasets to determine its classification accuracy. Logistic Regression, in essence, employs input features or attributes to estimate class probabilities during linear classification operations.

Following feature selection, the efficacy of three distinct classification models is evaluated: Random Forest, Logistic Regression, and Adaboost. In the training and testing phases, (70:30) splits of the dataset are used. We used accuracy as an evaluation metric to assess the efficacy of these models. These metrics can be used to assess the algorithms' ability to appropriately categorize levels of vaccine efficacy. The classifications generated using the Random Forest, Logistic Regression, and Adaboost models are shown in Table (4.3) below.

Table (4.3): Performance Evaluation Results for Parallel Part**.**

Feature Selection Method	Classifier	Accuracy		Precision		Recall		F1-score	
		(20%)	(60%)	(20%)	(60%)	(20%)	(60%)	(20%)	(60%)
MI	RF	0.96	0.96	0.93	0.93	0.97	0.97	0.95	0.95
	ADA	0.98	0.90	0.97	0.83	0.99	0.94	0.98	0.87
	LR	0.98	0.98	0.97	0.97	0.99	0.99	0.98	0.98
ANOVA	RF	0.98	0.96	0.97	0.93	0.99	0.97	0.98	0.95
	ADA	0.96	0.96	0.93	0.93	0.97	0.97	0.95	0.95
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
CHI^2	RF	0.96	0.96	0.93	0.93	0.97	0.97	0.95	0.95
	ADA	0.96	0.94	0.93	0.90	0.97	0.96	0.95	0.92
	LR	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

As indicated in Table (4.3) for the Performance Evaluation Results for the Parallel Part, the accuracy of the classification models approached 1.00 in some circumstances. This displays a score of 100% accuracy, indicating that the models correctly categorized all of the data. The Logistic Regression classifier attained an accuracy score of 1.00 in the 60% feature selection scenario using the ANOVA feature selection method. This suggests that the ANOVA-selected characteristics are very informative, and the classifier can properly predict whether or not the vaccines will function. The Logistic Regression classifier with CHI^2 feature selection approach produced an accuracy score of 1.00 for both the 20% and 60% feature selection scenarios. This illustrates that the CHI^2 feature selection method is effective at obtaining the crucial elements required for

precise categorization.

These findings emphasize the importance and effectiveness of feature selection in improving classification model performance. The capacity to produce exact estimates is strongly reliant on the selection of relevant data. Acquiring a perfect accuracy score of 1.00, as shown in Figure (4.7), illustrates the capabilities of the applied feature selection approaches and the skills of the logistic regression classifier in correctly predicting vaccine efficacy. Overall, the results demonstrate that the feature selection technique was successful, and that the combination of logistic regression with ANOVA and the CHI^2 feature selection approaches produced good results. Because of the models' faultless correctness, the conclusions of this thesis are more believable and reliable, highlighting the relevance of feature selection in boosting the performance of classification models.

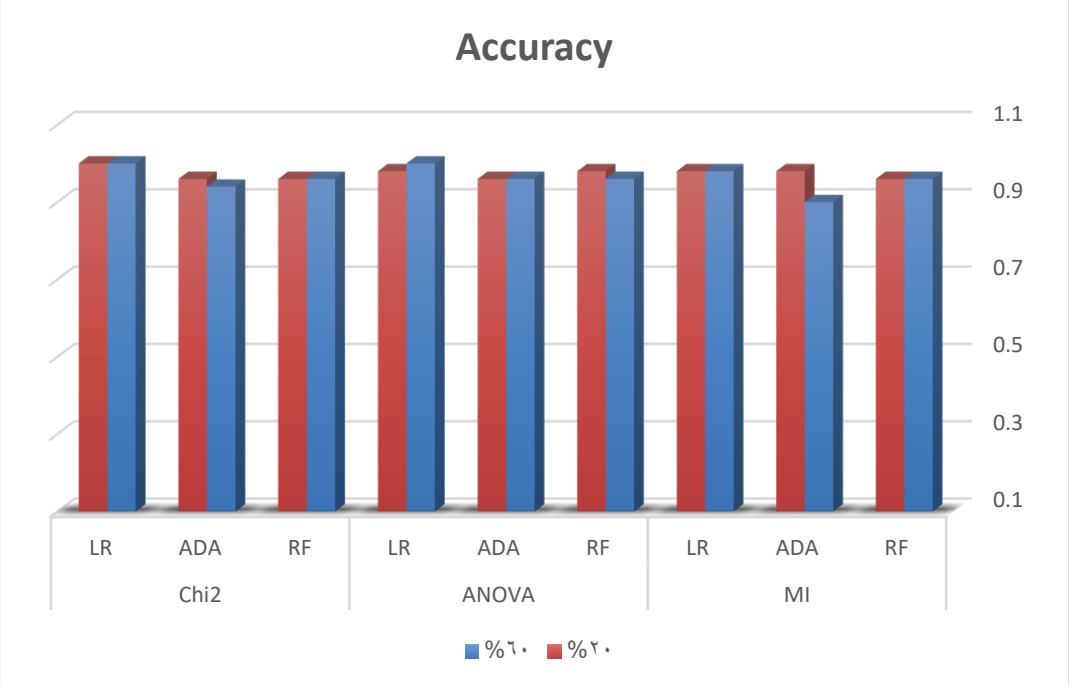


Figure (4.7): Analysis of Parallel Section Performance.

4.5.1.4. Important features produced from Parallel feature selection

Both the (60%) and the (20%) approaches have produced useful features for the feature selection sequence as shown in Figure (4.8) . Genes such as

['ZFC3H1', 'NEU3', 'C1RL-AS1', 'SNAPC1', 'XIST', 'CCDC103', 'C15orf26', 'ATP13A5-AS1', 'DDX11L10', 'SMC5', 'FAM222B', 'LINC00641', 'TRAPPC10'].

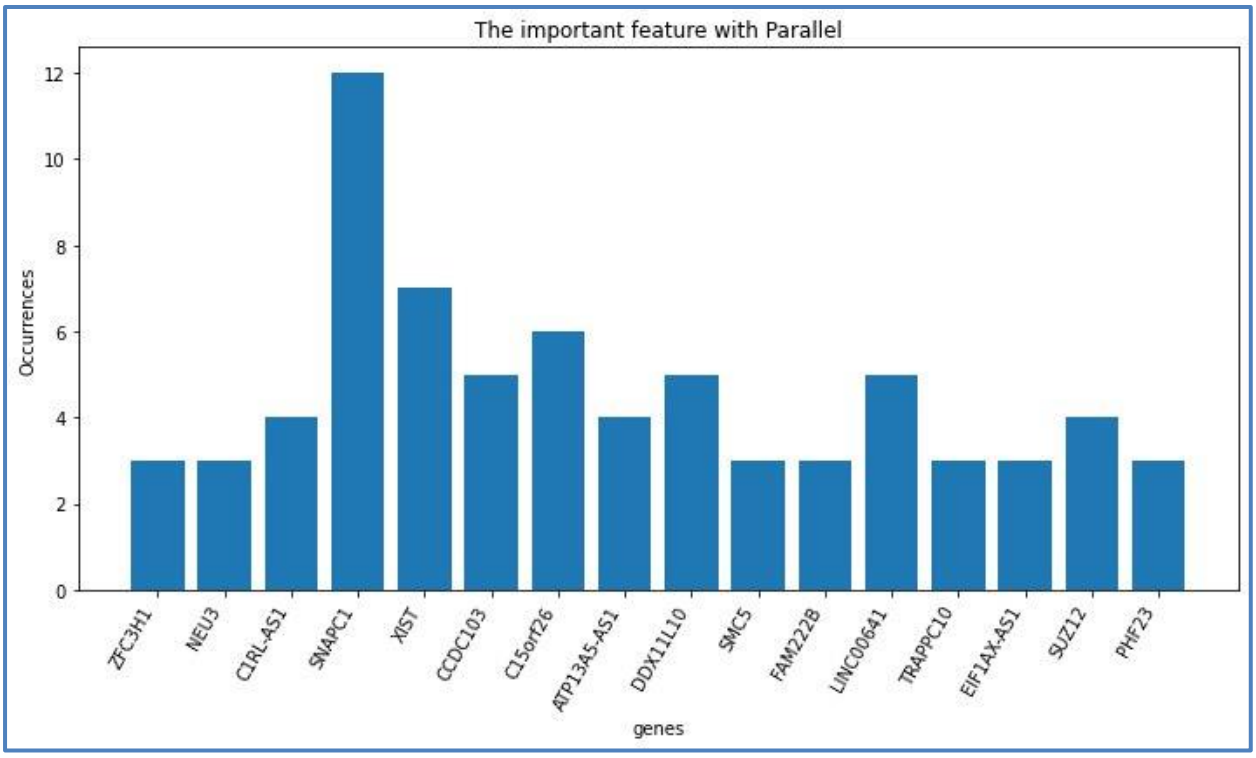


Figure (4.8): Important features from Parallel feature selection**.**

4.5.2. Sequential Feature Selection Part

The findings of the sequential feature selection technique, which tried to identify the traits that contributed the most informatively to the prediction of vaccination efficiency, are presented in this section. There are several sequential implementations of feature selection methods available, in which the subset obtained by one approach is fed into succeeding methods.

4.5.2.1. The Result of the Sequential Feature Selection

Methods for selecting features MI_CHI^2_ANOVA, MI_ANOVA_CHI^2, CHI^2_MI_ANOVA, CHI^2_ANOVA_MI, ANOVA_MI_CHI^2, and ANOVA_CHI^2_MI are all examples of ANOVA. Several techniques are used on the initial pool of 26,364 genes to minimize the number of characteristics. The final gene counts for feature

retention at (60%) differed between techniques, with 15,818, 9,791 and 5,694 genes discovered using each method, respectively. Various strategies have been proposed in the field of gene selection to retrieve important genes from a vast pool. Methods such as MI, ANOVA, and CHI^2 are included in this category. When just (20%) of the features were kept, the first strategy yielded 5,273 genes, the second yielded 1,055 genes, and the third yielded 211 genes. These chosen genes showed substantial significance to the dependent variable, making them amenable to future study.

Table (4.4): Number of Genes Selected using Sequential Feature Selection**.**

Feature Selection methods	The Number of Selected Genes			
	Percent	First	Second	Third
MI_ CHI²_ANOVA ***MI_ANOVA_ CHI²*** ***CHI² _MI_ANOVA***	20%	*5,273*	*1,055*	*211*
CHI² _ANOVA_MI ***ANOVA_MI_ CHI²*** ***ANOVA_ CHI² _MI***	60%	*15,818*	*9,791*	*5,694*

4.5.2.2. The Result of the Normalization

Normalization ensures that all characteristics of the sequential feature selection are on the same scale. This phase ensures that no single feature dominates the analysis. The dataset was Min-Max normalized using equation (2.1) from Section 2.3.2. of Chapter Two. Normalization the feature values to fit inside a predetermined range, which was often between 0 and 1, using this procedure. This normalization method provides fair comparison and reduces bias from the original feature scales for classification tasks.

Figures (4.9) through (4.14) show the results of using the Min-Max normalization technique to the MI_ CHI^2_ ANOVA (20%), MI_ ANOVA_

CHI^2(20%), CHI^2_ MI _ANOVA(20%), CHI^2_ ANOVA_ MI(20%), ANOVA_ MI_ CHI^2(20%), and ANOVA_ CHI^2_ MI (20%) datasets. This approach to normalization ensures that each feature from the sequential feature selection stage contributes equally to the analysis, removing any feature scale biases in the process. The normalization process aids in the standardization of feature representation, which increases analytical accuracy.

feature slection(ch_mi_anova)

	A2M	AAAS	ABCB7	ABCC1	ABCC3	...	ZSWIM8	ZXDC	ZYG11B	ZZEF1	ZZZ3
0	94	1814	2208	6762	342	...	3312	4483	4011	10833	4501
1	321	1632	1921	6507	349	...	3042	5574	4330	11450	4758
2	98	2500	2005	5775	195	...	4046	4303	3527	10210	4705
3	312	2604	2410	6651	1668	...	5638	6740	5249	15918	4067
4	106	2058	1609	6094	784	...	3738	4829	2659	11659	3805

	A2M	AAAS	ABCB7	...	ZYG11B	ZZEF1	ZZZ3
0	0.111546	0.323722	0.354381	...	0.155791	0.866667	0.330170
1	0.131115	0.824377	0.636598	...	0.334149	0.243137	0.649123
2	0.219178	0.666448	0.746778	...	0.784122	0.668736	0.857061
3	0.105675	0.472477	0.446521	...	0.283578	0.260566	0.279551
4	0.088063	0.281127	0.385309	...	0.337412	0.123747	0.376762

Figure (4.9): Normalization of the data using the min-max algorithm for CHI^2_MI_ANOVA (20%).

feature slection(ch_anova_mi)

	AAK1	ABCC1	ABCC3	ABCC5	ABCD4	...	ZSWIM8	ZXDC	ZYG11B	ZZEF1	ZZZ3
0	12319	6762	342	2470	2176	...	3312	4483	4011	10833	4501
1	16072	6507	349	2144	2199	...	3042	5574	4330	11450	4758
2	14569	5775	195	2921	2762	...	4046	4303	3527	10210	4705
3	16906	6651	1668	3781	3689	...	5638	6740	5249	15918	4067
4	11407	6094	784	2423	2910	...	3738	4829	2659	11659	3805

	AAK1	ABCC1	ABCC3	...	ZYG11B	ZZEF1	ZZZ3
0	0.620047	0.703324	0.292793	...	0.155791	0.866667	0.330170
1	0.265367	0.410782	0.234234	...	0.334149	0.243137	0.649123
2	0.542668	0.644205	0.104297	...	0.784122	0.668736	0.857061
3	0.293760	0.344295	0.173597	...	0.283578	0.260566	0.279551
4	0.165289	0.162803	0.309078	...	0.337412	0.123747	0.376762

Figure (4.10): Normalization of the data using the min-max algorithm for CHI^2_ ANOVA _MI (20%).

featureslection (mi_anova_ch)

	ABLIM3	ACRBP	ADD1	ADD3	AFF1	...	ZNF331	ZNF609	ZNF641	ZNF655	ZRANB2
0	147	853	16226	33809	12960	...	2716	12403	4020	8909	8680
1	247	986	15415	31634	14714	...	4004	14293	3737	9676	8319
2	91	513	19353	38092	12180	...	4841	11638	4792	10713	10563
3	284	1183	19258	36061	16127	...	4999	12299	6634	16745	11092
4	477	1174	17331	25932	10665	...	3853	10588	4127	10145	6786

	ABLIM3	ACRBP	ADD1	...	ZNF641	ZNF655	ZRANB2
0	0.245501	0.197212	0.464177	...	0.426907	0.382810	0.309845
1	0.152462	0.222349	0.638863	...	0.534761	0.203711	0.274558
2	0.046180	0.127514	0.629941	...	0.608418	0.432980	0.582965
3	0.196604	0.115174	0.375942	...	0.311537	0.198410	0.433518
4	0.355518	0.251371	0.334369	...	0.312664	0.075540	0.260288

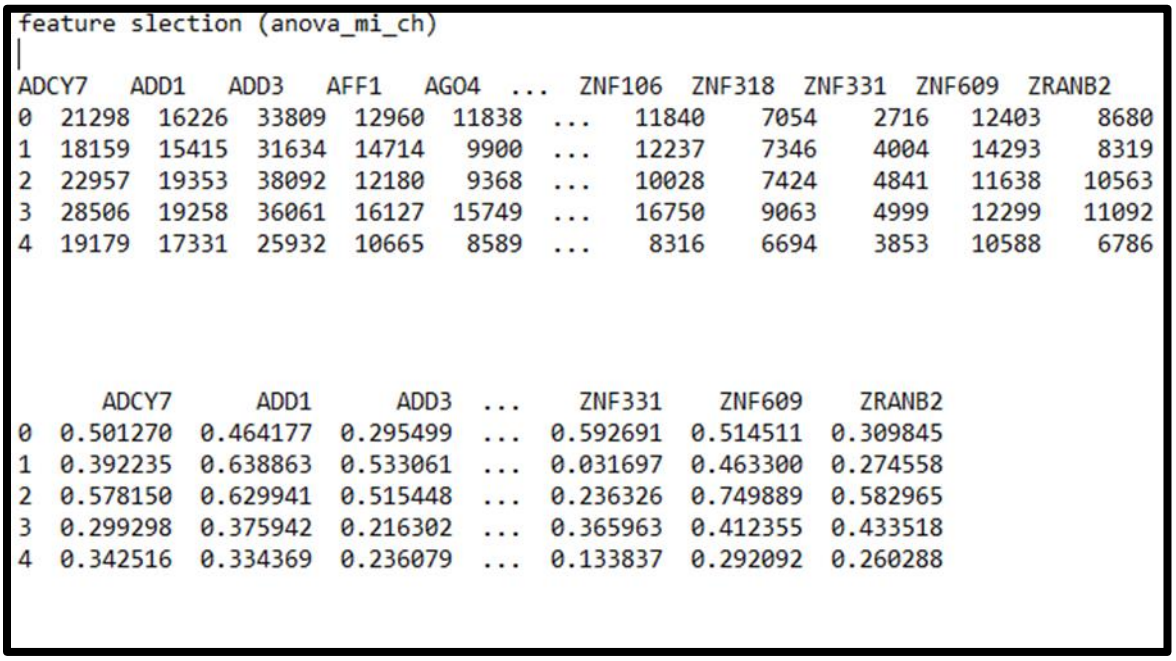
Figure (4.11): Normalization of the data using the min-max algorithm for MI_ANOVA_ CHI^2 (20%).

feature slection(mi_ch_anova)

	ABCC1	ABLIM3	ADD1	ADD3	AFF1	...	ZNF609	ZNF641	ZNF75D	ZRANB2	ZRSR2
0	6762	147	16226	33809	12960	...	12403	4020	3534	8680	3105
1	6507	247	15415	31634	14714	...	14293	3737	3635	8319	2905
2	5775	91	19353	38092	12180	...	11638	4792	3851	10563	3518
3	6651	284	19258	36061	16127	...	12299	6634	4115	11092	3633
4	6094	477	17331	25932	10665	...	10588	4127	3188	6786	3593

	ABCC1	ABLIM3	ADD1	...	ZNF75D	ZRANB2	ZRSR2
0	0.703324	0.245501	0.464177	...	0.282137	0.309845	0.373623
1	0.410782	0.152462	0.638863	...	0.479132	0.274558	0.149187
2	0.644205	0.046180	0.629941	...	0.810017	0.582965	0.554274
3	0.344295	0.196604	0.375942	...	0.262771	0.433518	0.164919
4	0.162803	0.355518	0.334369	...	0.229382	0.260288	0.146303

Figure (4.12): Normalization of the data using the min-max algorithm for MI_ CHI^2_ANOVA (20%).

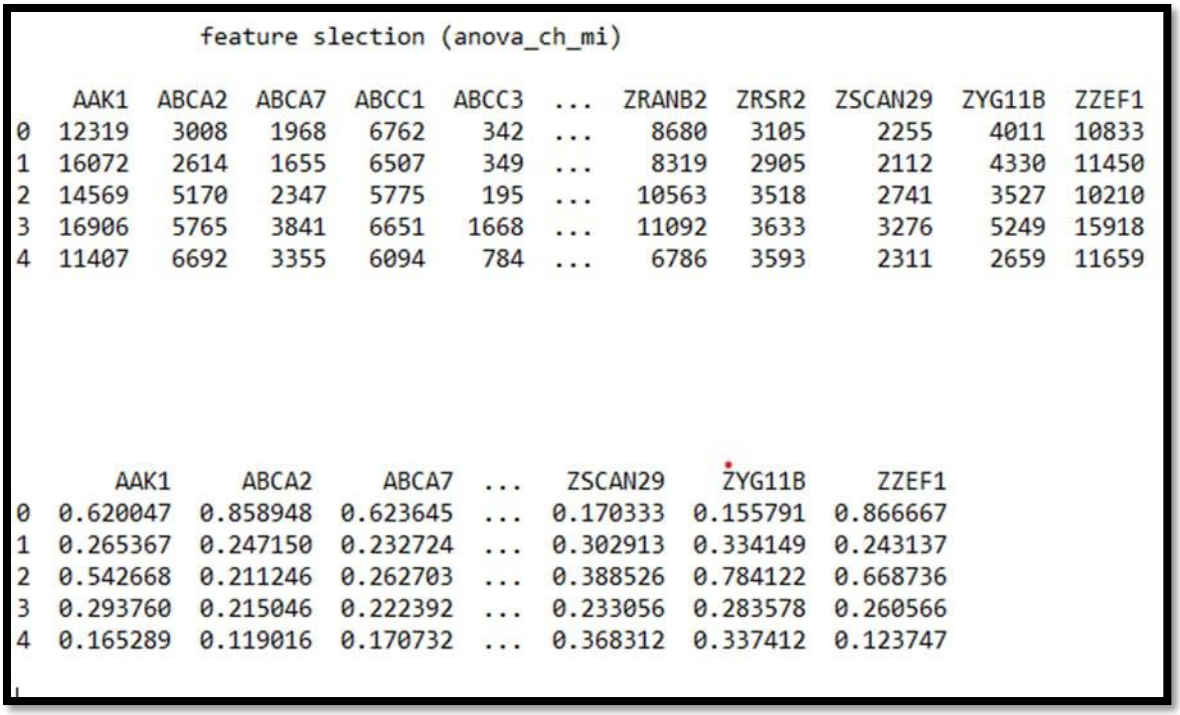


feature slection (anova_mi_ch)

	ADCY7	ADD1	ADD3	AFF1	AGO4	...	ZNF106	ZNF318	ZNF331	ZNF609	ZRANB2
0	21298	16226	33809	12960	11838	...	11840	7054	2716	12403	8680
1	18159	15415	31634	14714	9900	...	12237	7346	4004	14293	8319
2	22957	19353	38092	12180	9368	...	10028	7424	4841	11638	10563
3	28506	19258	36061	16127	15749	...	16750	9063	4999	12299	11092
4	19179	17331	25932	10665	8589	...	8316	6694	3853	10588	6786

	ADCY7	ADD1	ADD3	...	ZNF331	ZNF609	ZRANB2
0	0.501270	0.464177	0.295499	...	0.592691	0.514511	0.309845
1	0.392235	0.638863	0.533061	...	0.031697	0.463300	0.274558
2	0.578150	0.629941	0.515448	...	0.236326	0.749889	0.582965
3	0.299298	0.375942	0.216302	...	0.365963	0.412355	0.433518
4	0.342516	0.334369	0.236079	...	0.133837	0.292092	0.260288

Figure (4.13): Normalization of the data using the min-max algorithm for ANOVA_ MI_ CHI^2 (20%).



feature slection (anova_ch_mi)

	AAK1	ABCA2	ABCA7	ABCC1	ABCC3	...	ZRANB2	ZRSR2	ZSCAN29	ZYG11B	ZZEF1
0	12319	3008	1968	6762	342	...	8680	3105	2255	4011	10833
1	16072	2614	1655	6507	349	...	8319	2905	2112	4330	11450
2	14569	5170	2347	5775	195	...	10563	3518	2741	3527	10210
3	16906	5765	3841	6651	1668	...	11092	3633	3276	5249	15918
4	11407	6692	3355	6094	784	...	6786	3593	2311	2659	11659

	AAK1	ABCA2	ABCA7	...	ZSCAN29	ZYG11B	ZZEF1
0	0.620047	0.858948	0.623645	...	0.170333	0.155791	0.866667
1	0.265367	0.247150	0.232724	...	0.302913	0.334149	0.243137
2	0.542668	0.211246	0.262703	...	0.388526	0.784122	0.668736
3	0.293760	0.215046	0.222392	...	0.233056	0.283578	0.260566
4	0.165289	0.119016	0.170732	...	0.368312	0.337412	0.123747

Figure (4.14): Normalization of the data using the min-max algorithm for ANOVA _CHI^2_MI (20%).

4.5.2.3. The Result of the ML Classification Model

The Random Forest and Logistic Regression models both utilized a smaller set of characteristics generated via sequential feature selection. The Random Forest classifier's default parameters were utilized, so it had 100 estimators and a maximum depth of 0. This configuration was chosen to enhance efficiency while minimizing the danger of overfitting. The Logistic Regression technique is utilized to investigate how the decreased feature set influenced classification accuracy further. A Logistic Regression model is trained using the characteristics of interest and then tested on a different dataset.

Furthermore, the chosen characteristics are fed into the Adaboost algorithm, which is then applied to the dataset. Adaboost is an ensemble learning method that combines the output of numerous mediocre classifiers to build a single, excellent classifier. Setting the number of estimators (n estimators) to 50 determines the maximum number of boosting iterations before the process ends. By incorporating the features determined through sequential feature selection into the Random Forest, Logistic Regression, and Adaboost classification models, we can compare the accuracy and number of genes required by each model. This comprehensive method allows for a full investigation of the classification results and provides insights into the effectiveness of the selected variables in forecasting vaccine efficacy classes.

After feature selection, three distinct classification models (Random Forest, Logistic Regression, and Adaboost) are evaluated. The dataset is frequently partitioned into 70:30 training and testing halves. For measuring the success of these models, we used precision as our criterion. These measurements allow us to assess the algorithms' capacity to appropriately classify vaccine efficacy levels. Table 4.5 displays classifications made with the Random Forest, Logistic Regression, and Adaboost models.

To assess the reliability and generalizability of the model, an independent dataset

that was not previously used by the system was employed for testing. This approach involved randomly selecting 10% of the original dataset as a completely new set of data. The training data, which constituted 90% of the original dataset, was further divided into a 70% portion for training and a 30% portion for internal validation purposes. Subsequently, the performance of the model on this separate test set comprising untouched data was evaluated to demonstrate its accuracy and effectiveness.

Table (4.5): Performance Evaluation Results for Sequential Part**.**

Feature selection method	Classifier	Accuracy		Precision		Recall		F1-score	
		(20%)	(60%)	(20%)	(60%)	(20%)	(60%)	(20%)	(60%)
MI_CHI^2_ANOVA	RF	0.98	0.96	0.97	0.93	0.99	0.97	0.98	0.95
	ADA	0.96	0.98	0.93	0.97	0.97	0.99	0.95	0.98
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
MI_ANOVA_CHI^2	RF	0.96	0.96	0.95	0.93	0.95	0.97	0.95	0.95
	ADA	0.94	0.92	0.95	0.89	0.96	0.92	0.92	0.90
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
CHI^2_MI_ANOVA	RF	0.98	0.96	0.97	0.93	0.99	0.97	0.98	0.95
	ADA	0.96	0.98	0.93	0.97	0.97	0.99	0.95	0.98
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
CHI^2_ANOVA_MI	RF	0.96	0.96	0.95	0.93	0.95	0.97	0.95	0.95
	ADA	0.94	0.92	0.90	0.89	0.96	0.92	0.92	0.90
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
ANOVA_MI_CHI^2	RF	0.98	0.94	0.97	0.90	0.99	0.96	0.98	0.92
	ADA	0.90	0.96	0.85	0.93	0.90	0.97	0.87	0.96
	LR	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
ANOVA_CHI^2_MI	RF	0.98	0.94	0.97	0.90	0.99	0.96	0.98	0.90
	ADA	0.90	0.96	0.85	0.93	0.90	0.97	0.87	0.95
	LR	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

In the performance evaluation findings for the sequential feature selection procedures, the Logistic Regression classifier received perfect accuracy scores (1.00). This demonstrates that the Logistic Regression model performed flawlessly for various feature selection tactics. To be more explicit, when the feature selection approach was "MI_ CHI^2 _ANOVA," the Logistic Regression classifier earned an accuracy score of 1.00 for the 60 percent feature retention scenarios.

The Logistic Regression classifier achieved a high quality score of 1.00 in the 60% case using the feature selection method " MI ANOVA CHI2." In the 60% case, the Logistic Regression classifier obtained a high score of 1.00 while using the feature selection method " CHI2 ANOVA MI ". When tested with the "CHI2 MI_ ANOVA" feature selection method, the Logistic Regression classifier achieved a high score of 1.00 in the 60% example. This illustrates that the Logistic Regression model using the chi-square, mutual information, and ANOVA characteristics achieved perfect classification accuracy for predicting vaccine efficacy. The Logistic Regression classifier earned perfect accuracy ratings of 1.00 in both the 60% and 20% situations. This was also true for the feature selection procedures "ANOVA_ MI_ CHI2" and "ANOVA_ CHI2 _ MI". This suggests that the variables selected using the ANOVA, mutual information, and chi-square criteria were extremely relevant and useful in predicting vaccine efficacy, and that the classification performance was flawless.

4.4.2.4. Important features produced from Sequential feature selection

The study used feature selection sequences, as shown in Figure (4.15), to see how good they were at identifying specific genes for analysis. The following six strategies are investigated: MI_CHI2_ANOVA, MI_ANOVA_CHI2, CHI2_MI_ANOVA, CHI2_ANOVA_MI, ANOVA_MI_CHI2, and ANOVA_CHI2_MI are all examples of ANOVA. The results show that genes including ['LINC00641', 'XIST', 'ZFC3H1', 'C2orf88', 'TRAPPC10', 'ID2', 'TUBB1', 'SUZ12', 'NFE218LK', 'PPBP', and 'YWHAG'] are involved. Furthermore, for (20%) and (60%), the same list of very accurate genes was created.

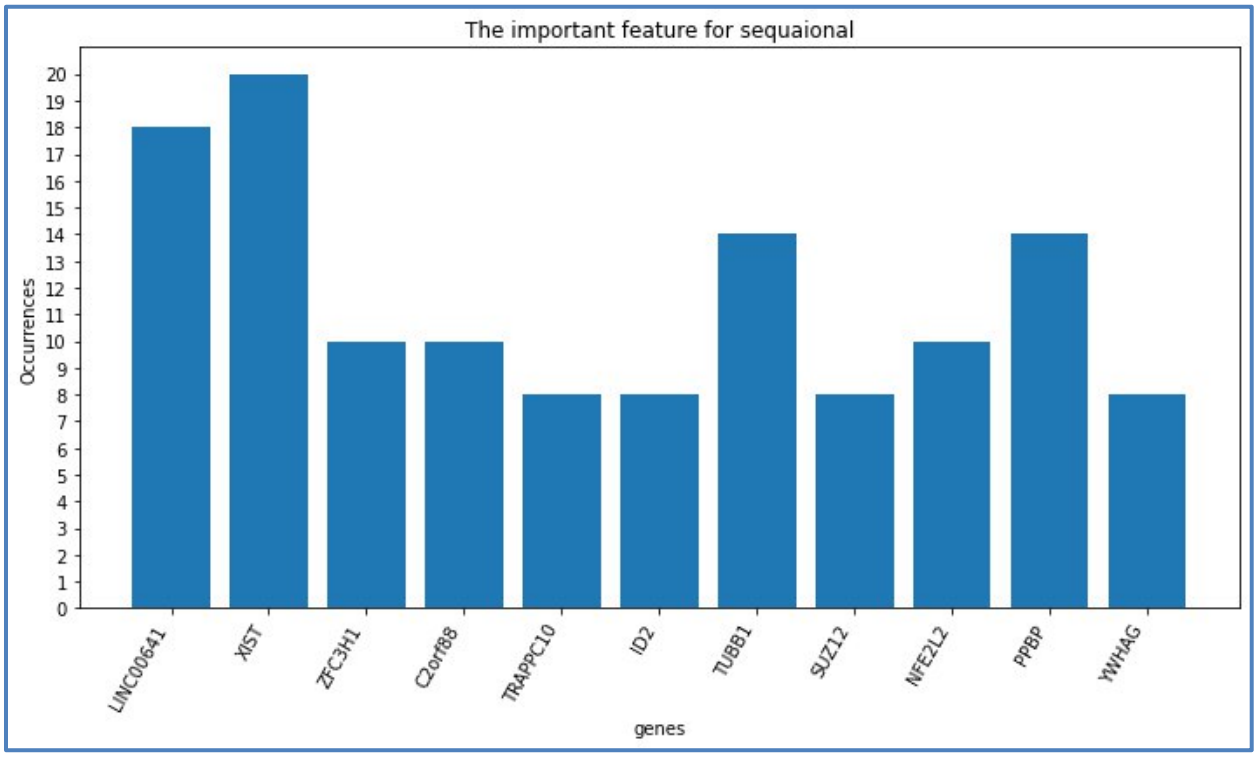


Figure (4.15): Important features from Sequential feature selection.

4.5.3. Hybrid Feature Selection

In the hybrid feature selection process, the analysis of variance (ANOVA), chi-square (CHI^2), and mutual information (MI) are combined with the Random Forest (RF) feature selection algorithm. The input is obtained by the RF classifier from each of the feature selection methods.

4.5.3.1. The Result of the hybrid Feature Selection

To choose features, this study applies hybrid feature selection. Both approaches are used in this solution to improve feature selection and classification model accuracy. The Random Forest (RF) feature selection approach was combined with mutual information (MI), analysis of variance (ANOVA), and chi-square (CHI^2) as illustrated in Table (4.6). The MI_ RF approach is used to choose 596 features from 15,818 features (60 percent scenario) and 506 features from 5,273 features (20 percent scenario). ANOVA_ RF chose 577 (60 percent scenario) and 499 (20 percent scenario) attributes from a total of 15,818 and 5,273 features. CHI^2_ RF chose 548 (60 percent scenario) and 477 (20 percent scenario) features from a total of 15,818 and 5,273 features. The hybrid strategy

leverages the complementary nature of these parallel feature selection methods with the Random Forest algorithm to select more relevant and informative features for improved classification performance.

Table (4.6): Number of Genes Selected using Hybrid Feature Selection**.**

Feature Selection methods		Feature Selection percent	The Number of Selected Genes	
1.	***MI_RF***	60%	*MI: 15,818*	*RF:* 596
		20%	*MI: 5273*	*RF:* 506
2.	***ANOVA_RF***	60%	*ANOVA: 15,818*	*RF:* 577
		20%	*ANOVA: 5273*	*RF:* 499
3.	***CHI^2 _RF***	60%	*CHI^2: 15,818*	*RF:* 548
		20%	*CHI^2: 5273*	*RF:* 477

4.5.3.2. The Result of the Normalization

Normalization assures that all features of hybrid feature selection are comparable. This stage is critical to avoiding one trait dominating the analysis. To apply Min-Max normalization to the dataset, equation (2.1) from Chapter Two is used , Section 2.3.2. This normalization approach restricted feature values to 0 to 1. Because it reduces feature scale bias and allows for fair comparison, this normalization method is important for classification problems.

Figure 4.16 compares the dataset before and after normalization to highlight the influence of Min-Max normalization for MI_ RF (20%). Before normalization (as shown in Figure 4.17), the amplitude and range of the ANOVA_ RF (20%) feature values in the dataset vary greatly. Figure 4.18 depicts the normalized dataset, with all feature values scaled for the CHI^2_ RF (20%). This normalization method removes any potential biases introduced by the original feature scales, ensuring that each feature chosen in the parallel feature selection stage contributes equally to the analysis. Furthermore, normalization improves analytical precision by ensuring a consistent and standardized representation of characteristics.

hyprid feature slction (MI_Randam forst)

	AASDHPPT	ABCF1	ABCF2	ABCG1	ACAD9	...	ZNF738	ZNF791	ZNF862	ZRANB2	ZSWIM1
0	1718	3352	3442	937	1053	...	1481	1988	3433	8680	721
1	1619	3514	2802	1042	883	...	1402	2550	3731	8319	812
2	1877	4021	3451	690	1125	...	1531	2067	3639	10563	587
3	2134	4824	3066	1162	3383	...	1569	3417	4076	11092	611
4	1205	4094	2672	826	1224	...	917	2694	3381	6786	518

	AASDHPPT	ABCF1	ABCF2	...	ZNF862	ZRANB2	ZSWIM1
0	0.222448	1.000000	0.427958	...	0.557076	0.309845	0.380560
1	0.340274	0.261729	0.706213	...	0.434770	0.274558	0.347611
2	0.591671	0.213339	0.515202	...	0.713745	0.582965	0.546952
3	0.284408	0.549548	0.471580	...	0.308096	0.433518	0.476112
4	0.449467	0.133810	0.378387	...	0.212871	0.260288	0.177924

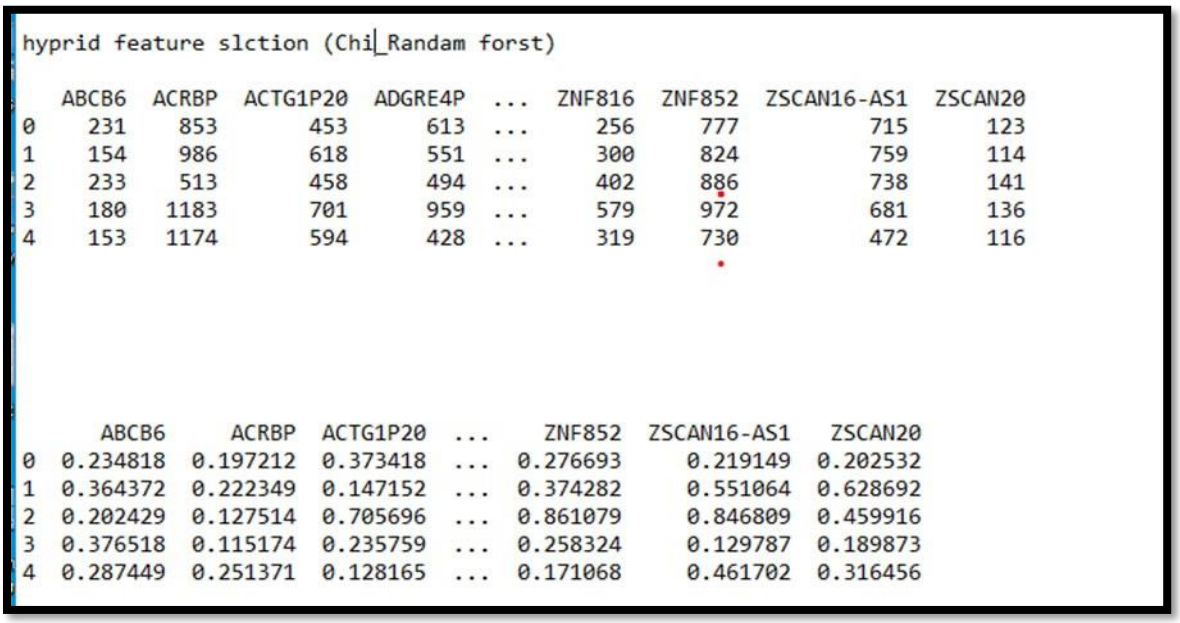
Figure (4.16): Normalization of the data using the min-max algorithm for MI_ RF (20%).

hyprid feature slction (ANOVA_Randam forst)

	AANAT	ABCC1	ABI2	ABLIM3	ACBD3	...	ZNF853	ZRSR2	ZSCAN20	ZSCAN30	ZUFSP
0	72	6762	2926	147	3150	...	459	3105	123	1184	723
1	60	6507	3168	247	3729	...	423	2905	114	1153	806
2	80	5775	3255	91	3915	...	473	3518	141	1223	1010
3	139	6651	3459	284	4932	...	343	3633	136	1055	1317
4	121	6094	2337	477	3597	...	449	3593	116	809	974

	AANAT	ABCC1	ABI2	...	ZSCAN20	ZSCAN30	ZUFSP
0	0.543353	0.703324	0.299250	...	0.202532	0.187739	0.630675
1	0.341040	0.410782	0.401500	...	0.628692	0.370881	0.241718
2	0.514451	0.644205	0.694433	...	0.459916	0.397701	0.392638
3	0.167630	0.344295	0.393999	...	0.189873	0.177778	0.515337
4	0.277457	0.162803	0.310699	...	0.316456	0.272797	0.085890

Figure (4.17): Normalization of the data using the min-max algorithm for ANOVA_ RF (20%).



hyprid feature slction (Chi_Randam forst)

	ABCB6	ACRBP	ACTG1P20	ADGRE4P	...	ZNF816	ZNF852	ZSCAN16-AS1	ZSCAN20
0	231	853	453	613	...	256	777	715	123
1	154	986	618	551	...	300	824	759	114
2	233	513	458	494	...	402	886	738	141
3	180	1183	701	959	...	579	972	681	136
4	153	1174	594	428	...	319	730	472	116

	ABCB6	ACRBP	ACTG1P20	...	ZNF852	ZSCAN16-AS1	ZSCAN20
0	0.234818	0.197212	0.373418	...	0.276693	0.219149	0.202532
1	0.364372	0.222349	0.147152	...	0.374282	0.551064	0.628692
2	0.202429	0.127514	0.705696	...	0.861079	0.846809	0.459916
3	0.376518	0.115174	0.235759	...	0.258324	0.129787	0.189873
4	0.287449	0.251371	0.128165	...	0.171068	0.461702	0.316456

Figure (4.18): Normalization of the data using the min-max algorithm for CHI^2_ RF (20%).

4.5.3.3. The Result of the Classification Model

The Random Forest technique uses a reduced feature set from sequential feature selection algorithms to classify the dataset. The Random Forest classifier uses 100 estimators (n estimators) by default, with a maximum depth of 0, to balance performance with overfitting avoidance. This configuration aids the Random Forest model in generalizing to new data. Logistic Regression, a prominent classification method, is used for feature selection and assessing the classification accuracy of the reduced feature set. Using specific features, logistic regression models are developed and validated. The model's performance will be evaluated using key metrics.

The Adaboost technique generates a highly accurate model by combining the outputs of multiple weak classifiers. The maximum number of boosting rounds before the algorithm stops is determined by n estimators. Adaboost enhances dataset categorization by employing several classifiers. By including them into the classification step, then compare the effectiveness of the Random Forest, Logistic Regression, and Adaboost classifiers in predicting vaccination efficacy categories using the reduced feature set from feature selection.

Following feature selection, three distinct classification models (Random Forest,

Logistic Regression, and Adaboost) are examined and contrasted. The dataset is frequently split in half, with one half utilized for training and the other for testing (70:30). The accuracy of these models is used to assess their effectiveness. These metrics allow us to see how successfully the algorithms classify the effectiveness of vaccines. Table 4.7 shows the categorization model results for Random Forest, Logistic Regression, and Adaboost.

In several of the situations given in Table 4.7, the hybrid feature selection approaches achieve an accuracy score of 1.00. This indicates that the classification performance in these circumstances is flawless. The MI_ RF feature selection strategy produced an accuracy score of 1.00 for both the Adaboost (ADA) and Logistic Regression (LR) classifiers in the 20% feature selection scenario. This signifies that the Random Forest collaborated perfectly with the mutual information-based feature selection to generate ideal classification accuracy. The accuracy of the ANOVA_ RF feature-selection approach's 60 percent feature-selection scenario employing the Random Forest (RF) and Logistic Regression (LR) classifiers is 1.00. This means that the Random Forest is able to correctly categorize the vaccine's efficacy using the features selected using the ANOVA criterion.

Table (4.7): Performance Evaluation Results for Hybrid Part .

Feature Selection Method	Classifier	Accuracy		Precision		Recall		F1-score	
		(20%)	(60%)	(20%)	(60%)	(20%)	(60%)	(20%)	(60%)
MI_ RF	RF	0.96	0.96	0.93	0.93	0.97	0.97	0.95	0.95
	ADA	**1.00**	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98
	LR	**1.00**	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98
ANOVA_RF	RF	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
	ADA	0.96	0.96	0.95	0.93	0.95	0.97	0.95	0.95
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**
CHI^2 _RF	RF	0.96	0.98	0.93	0.97	0.97	0.99	0.95	0.98
	ADA	0.92	0.96	0.90	0.93	0.90	0.97	0.90	0.95
	LR	0.98	**1.00**	0.97	**1.00**	0.99	**1.00**	0.98	**1.00**

Finally, utilizing the CHI^2_ RF feature selection strategy, the Logistic Regression (LR) classifier got an accuracy score of 1.00 in the 60% feature selection scenario. This suggests that the Random Forest functioned well in unison with the CHI-selected features to appropriately categorize the dataset. Accuracy values of 1.00, the highest attainable, indicate that the hybrid feature selection approach performed flawlessly in these situations. It stresses the ability of the selected features to accurately discern between different levels of vaccine efficacy, as obtained through the integration of parallel feature selection and Random Forest. As a result, it is obvious that the proposed methodology has the potential to provide valuable insights into how various immunizations alter the antibody response and the genetic immune system.

4.4.3.4. Important features produced from Hybrid feature selection

The significant features produce a result of key genes (see Figure (4.19)) from the feature hybrid selection sequence utilizing the ANOVA_RF and CHI^2_RF methods, with the MI_RF techniques generating the same result for (20%) and (60%). This provides the most relevant properties of genes ['LINC00641', 'EIF1AX-AS1', 'TRIM52', 'NEU3', 'PAXBP1', 'ZFC3H1', 'LOC101929516', 'SNAPC1', 'RNF219', 'XIST', 'NFE2L2', 'MIR25', 'SIAH1].

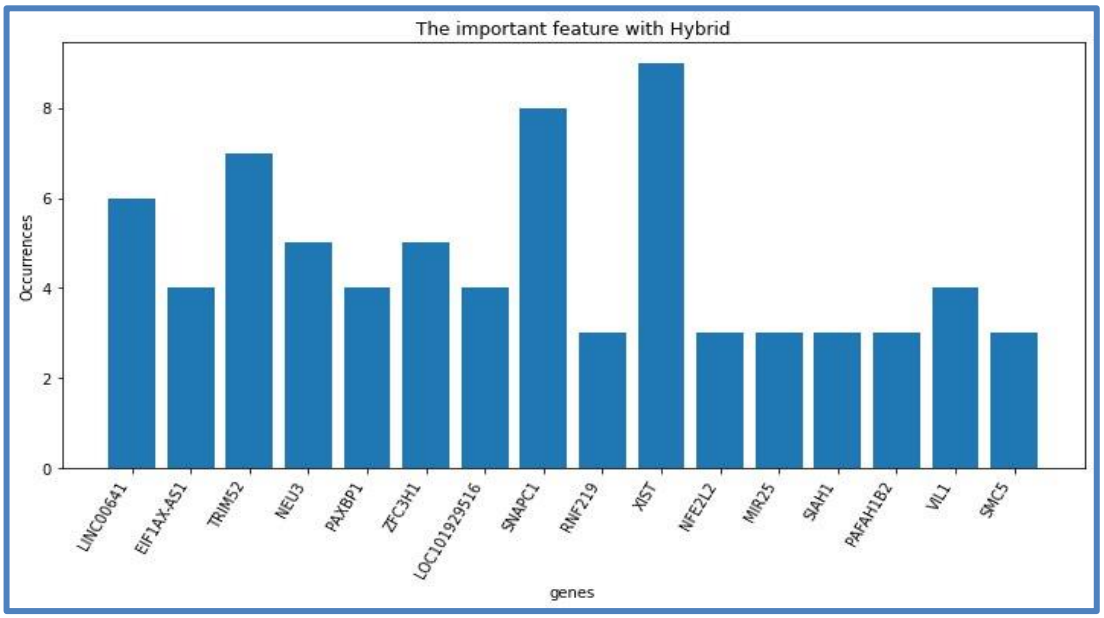


Figure (4.19): Important features from Hybrid feature selection**.**

4.5. CNN Classification Model Stage

For predicting gene expression data, a Convolutional Neural Network (CNN) model is proposed. The Keras Python library, which provides a high-level interface for creating neural networks, is used to implement the CNN model. This allows for rapid prototyping and experimentation with various topologies. The CNN model architecture is made up of eight convolutional layers, eight LeakyReLU activation layers, seven max-pooling levels, one fully connected layer, and three dense layers.

The CNN model begins with a convolutional layer with a 3x1 kernel and 16 distinct filters. Following this layer is a LeakyReLU activation function, which brings non-linearity into the network. The feature maps are downsampled using a max-pooling layer. The procedure is repeated with a second convolutional layer with a 3x1 kernel and 32 filters. Another max-pooling layer is employed to lower the dimensionality even further. A third convolutional layer is then deployed, with a kernel size of 3x1 and 64 filters. Following that is a LeakyReLU activation function and another max-pooling layer. The third max-pooling layer's output is flattened to form a one-dimensional feature vector. Following that, the flattened features are passed through a sequence of completely connected layers.

The first completely connected layer contains 256 filters with ReLU activation. A dropout layer is added after the first completely linked layer to prevent overfitting. Following that, another dropout layer is added, followed by a second fully linked layer with 512 filters and a ReLU activation function. Finally, there is a third completely connected layer with 1024 filters and a ReLU activation algorithm.

The final fully connected layer is made up of two neurons that represent the two classes in the gene dataset and uses a softmax activation function for prediction. To compute the loss, categorical cross-entropy is used as the loss function, while accuracy is employed as the evaluation metric. The Adam optimizer with a learning rate of 0.001 is used to optimize the model. The gene expression data is fed into the CNN model for

50 epochs during the training procedure. During training, the model learns to extract relevant characteristics from input data and generate accurate classification predictions.

4.5.1. Performance Evaluation For Parallel Part

To improve accuracy score, three feature selections (MI (20%), ANOVA (20%), and MI (60%)) are used in parallel feature selection. These methods are enhanced by including a CNN classification model with a 70:30 ratio between training and testing. Including convolutional neural networks (CNNs), a deep learning-based technique, had the potential to improve classification accuracy by using CNNs' ability to capture complex patterns and correlations in data. The MI (20%), ANOVA(20%), and MI(60%) approaches were used to select features, which were then input into a CNN model.

Deep learning algorithms using feature-selected datasets can improve performance, and CNN classification results can explain why. We may compare the accuracy scores obtained by the CNN classification model with the starting results of the parallel feature selection procedures to see how effectively the CNN model increases system accuracy. This activity is taken to study the potential benefits of combining CNN's powerful learning capabilities with parallel feature selection methodologies. The findings generated by the CNN classification model will provide insight into the usefulness of the parallel technique in predicting vaccination success based on antibody response and genetic immune system traits.

Figures 4.20, 4.21, and 4.22 show the accuracy and validation accuracy tendencies for the MI (60%), MI(20%), and ANOVA(20%) feature selection approaches. The evolution of the CNN model is graphically displayed from the beginning of training to the finish. The accuracy and validation accuracy curves indicate that three alternative feature selection techniques all converge to an accuracy of 1.00 over time. The results demonstrate the effectiveness of the CNN classification model

when used with parallel feature selection procedures to improve overall system accuracy. According to the data, the CNN model consistently predicted the vaccine's effectiveness based on antibody response and hereditary immune system features.

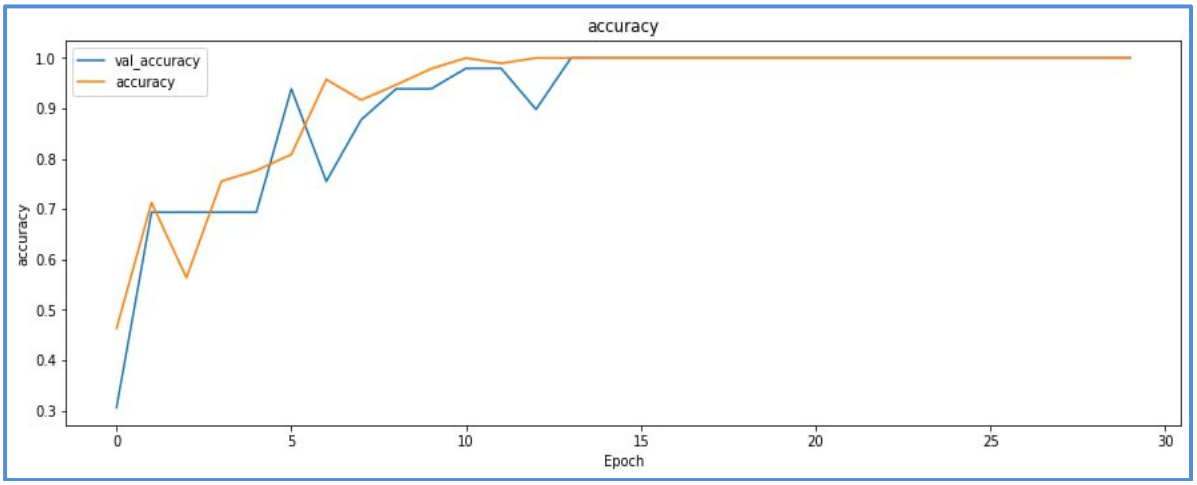


Figure (4.20): Accuracy and Val_ Accuracy for MI (60%).

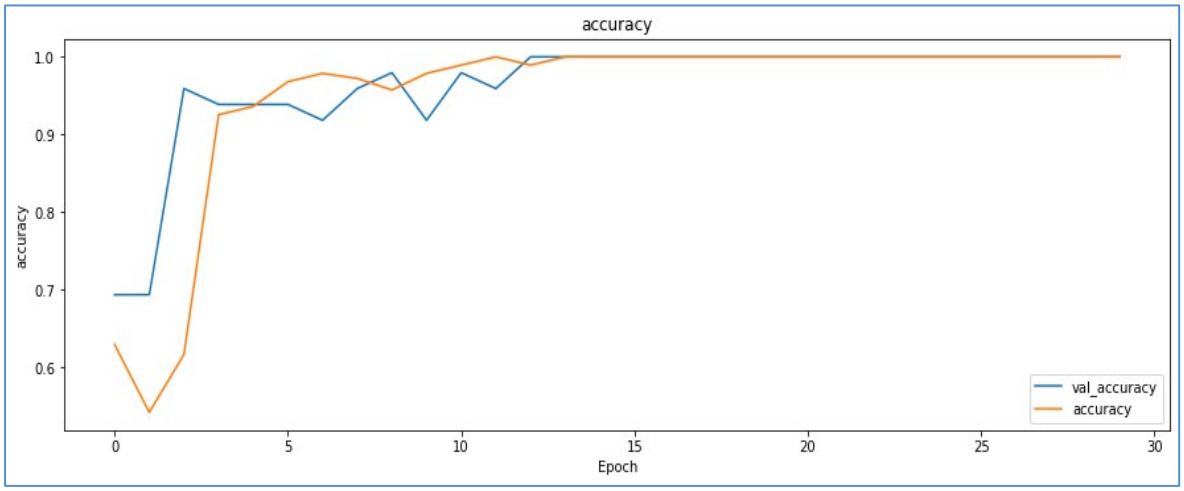


Figure (4.21): Accuracy and Val_ Accuracy for MI (20%).

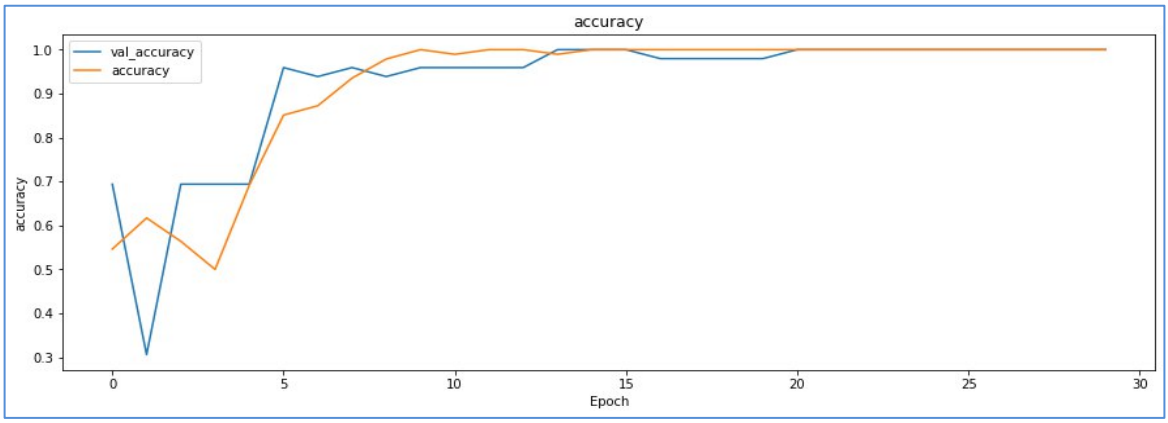


Figure (4.22): Accuracy and Val_ Accuracy for ANOVA(20%).

4.5.2. Performance Evaluation for Sequential Part

In the sequential feature selection phase, the CNN classification algorithm is utilized to evaluate additional strategies for improving accuracy. MI_ CHI^2_ ANOVA(20%), MI_ ANOVA_ CHI^2 (20%), CHI^2_ MI_ ANOVA(20%), and CHI^2_ ANOVA_ MI(20%) are chosen as the best for use in CNN classification. Seventy percent of the data is used to train the CNN model, with the remaining thirty percent used to assess its effectiveness. The purpose of this strategy is to improve overall system precision by using the benefits of deep learning.

If the CNN classification method is utilized, the system's accuracy may improve when compared to results achieved with other classifiers. CNNs, with their hierarchical design and feature learning capabilities, are an excellent choice for improving classification accuracy because they can capture complex relationships and patterns in data. By evaluating the performance of the CNN model on the features acquired using the MI_ CHI^2_ ANOVA(20%), MI_ ANOVA _CHI^2 (20%), CHI^2_ MI_ ANOVA (20%), and CHI^2_ ANOVA_ MI (20%) feature selection methods, it is possible to learn about how deep learning approaches affect the classification of vaccine efficacy. This investigation will reveal how well sequential feature selection collaborates with CNN to improve the system's accuracy and prediction skills.

As evidenced by accuracy, recall, and F1-score values of 1.00 across all techniques, the CNN model exhibited high precision and recall, resulting in balanced

performance and dependable predictions. This suggests that the features selected performed a good job of capturing the important components and patterns associated with vaccine efficacy, resulting in accurate forecasts free of false positives and negatives. The accuracy and validation accuracy plots presented in Figures 4.23, 4.24, 4.25, and 4.26 further verify the CNN model's good performance on the chosen features. As indicated by the constant convergence and high accuracy values during the training, the CNN model successfully learnt the underlying patterns and generated correct predictions for the vaccine's efficacy.

These findings imply that when sequential feature selection approaches are paired with CNN classification, good performance in consistently estimating immunization efficacy is attained. The CNN model's ability to leverage deep learning features and extract difficult features from the chosen feature subsets contributes to its high accuracy. These findings illustrate the potential of using cutting-edge deep learning techniques to investigate the genetics underpinning vaccine responses and immunological responses.

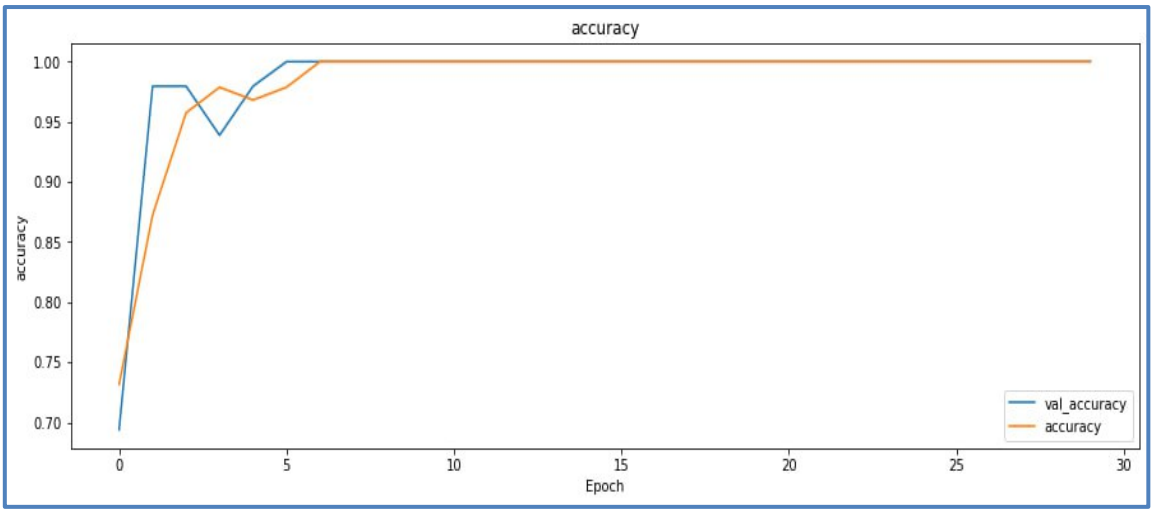


Figure (4.23): Accuracy and Val_ Accuracy for MI_ CHI^2_ANOVA(20%).

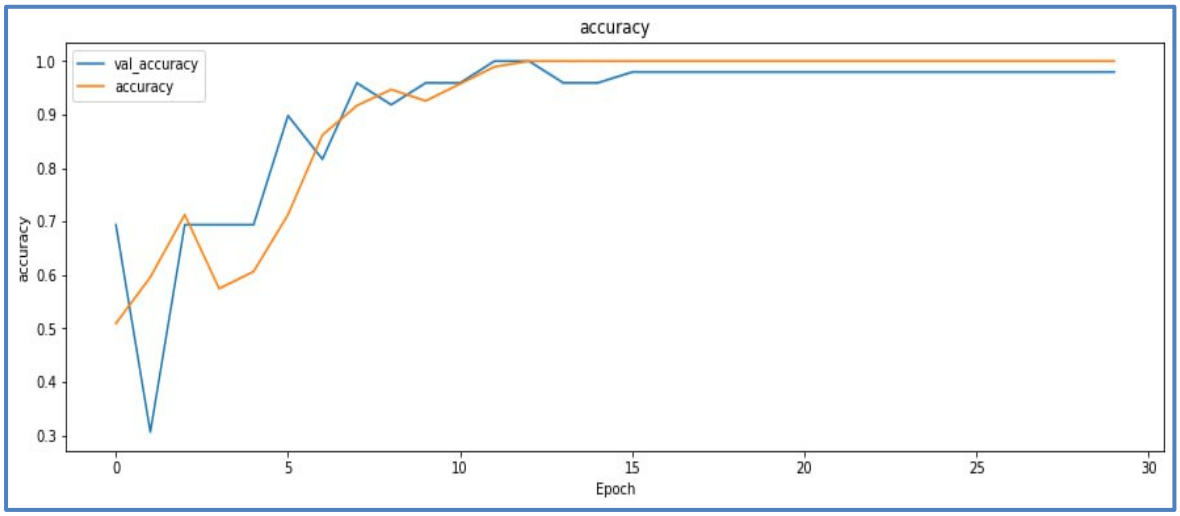


Figure (4.24): Accuracy and Val_ Accuracy for MI_ANOVA_ CHI^2(20%).

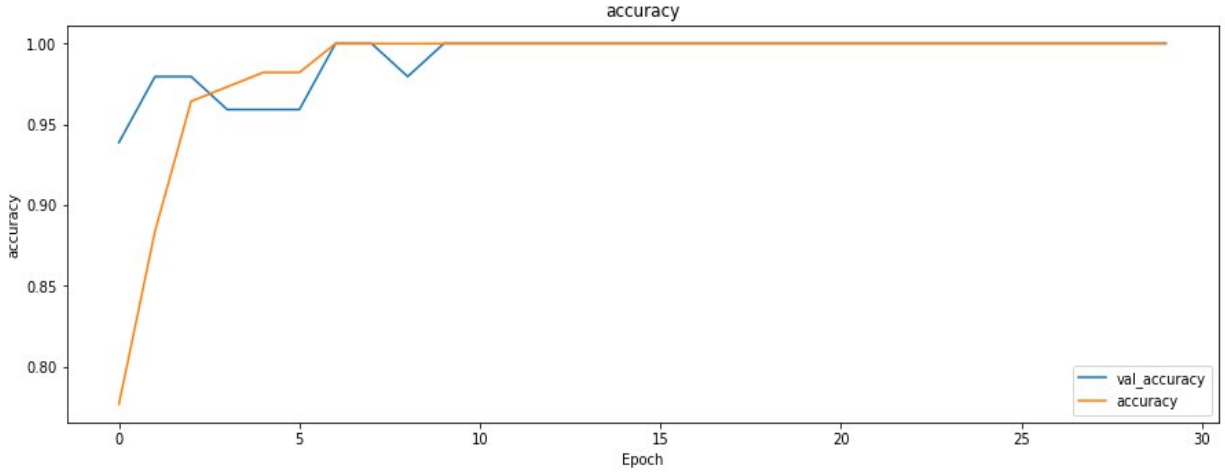


Figure (4.25): Accuracy and Val_ Accuracy for CHI^2_MI_ANOVA(20%).

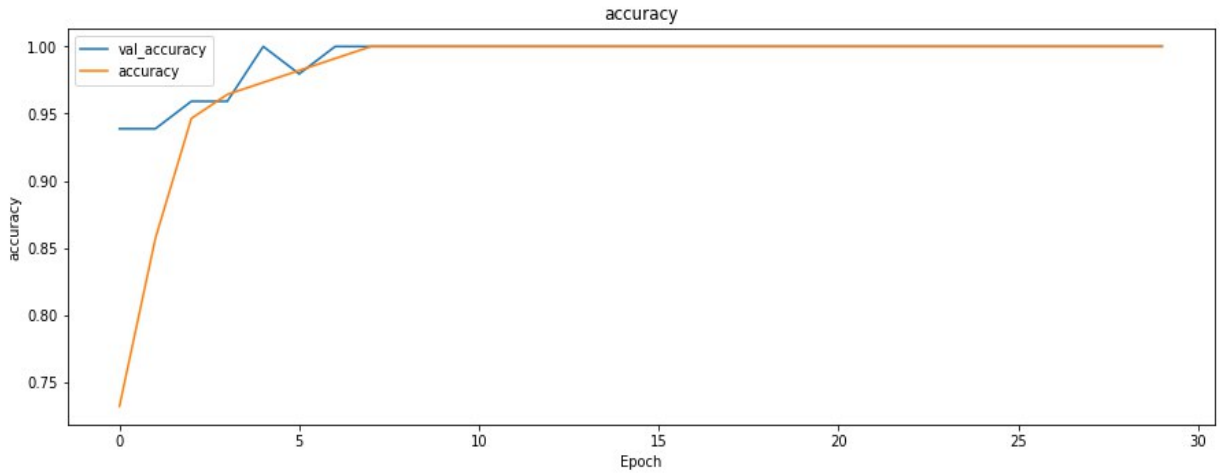


Figure (4.26): Accuracy and Val_ Accuracy for CHI^2 _ANOVA_MI(20%).

4.5.3. Performance Evaluation For Hybrid Part

The hybrid feature selection accuracy ratings were below 1.00 in several instances. These illustrations use the feature selection methods MI_ RF (60%), ANOVA_ RF (20%), and CHI2_RF (20%). The accuracy of the classification system is improved further by feeding these manually picked characteristics into a Convolutional Neural Network (CNN) classifier. The dataset is split 70:30 for training and testing the CNN classifier, with 70% of the data used for training and 30% for testing. The goal is to improve the system's precision by capturing complex relationships and patterns within the selected characteristics by leveraging the capability of CNN, which excels at image and pattern recognition tasks.

Even if the accuracy ratings generated by the hybrid feature selection approaches did not reach 1.00, using CNN enables for further refinement of classification performance. Using CNN allows for a more in-depth analysis of the selected features, which may lead to the discovery of new insights and correlations, resulting in better accuracy. In such cases, the usage of CNN signifies a multi-stage technique for feature selection and classification, with the goal of improving the system's accuracy. Combining the benefits of hybrid feature selection methods with CNN might considerably increase the accuracy of the system's predictions, leading to a better understanding of how different vaccinations affect the antibody response and the genetic immune system.

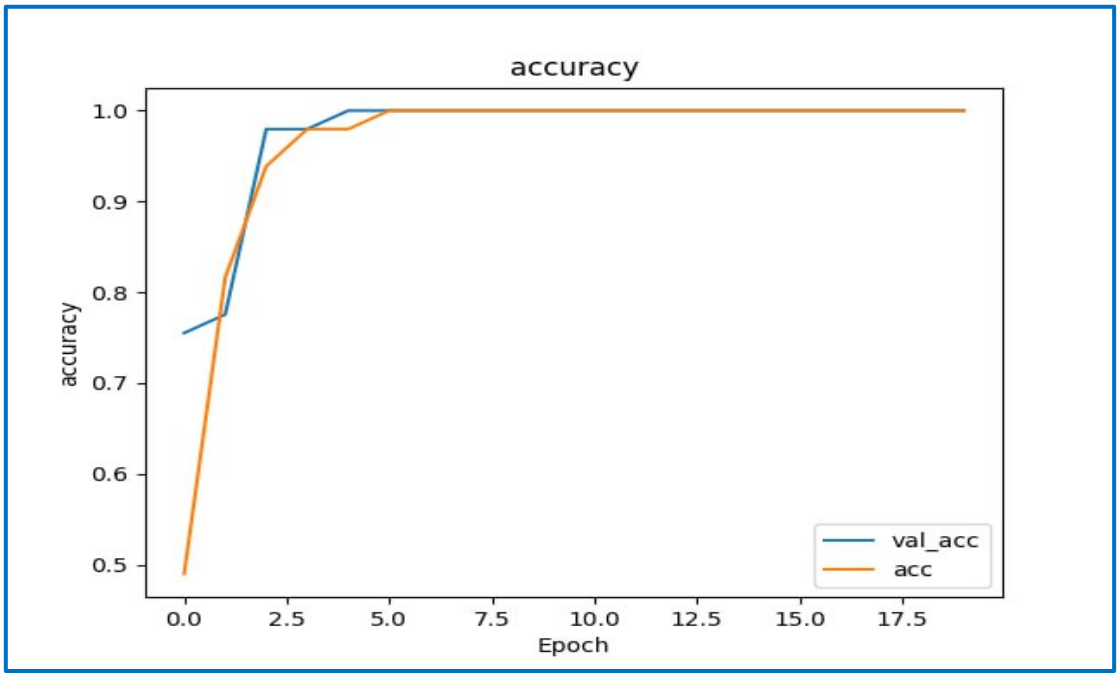


Figure (4.27): Accuracy and Val_ Accuracy for MI_ RF (60%).

4.6. Predicted Genes Names Classification Model

The Classification Model-predicted gene names are displayed in Table (4.8) and Figure (4.28). This diagram depicts all of the predicted and identified genes in this dataset. Anticipated gene names are required for understanding the genomic landscape and identifying immune response and antibody production. By assessing projected gene names, researchers can find genetic markers and pathways that influence vaccine efficacy. These projected gene names help researchers identify potential biomarkers or genetic elements for further study, as well as indicators of immune response and vaccine efficiency.

Table (4.8): The Names of Predicted Genes for GSE201535 Dataset.

GENE	Biological Name	Biological Description
LINC00641	(long intergenic non-protein-coding41)	It is a non-coding RNA molecule that does not produce a protein.
XIST	(X-inactive-specific transcript)	It is an RNA molecule that plays a crucial role in X chromosome inactivation, a process by which one of the two X chromosomes in female cells is inactivated to achieve dosage compensation with males.
SMC5	(structural maintenance of chromosomes 5)	It is a gene that encodes a component of the SMC5/6 complex, Mutations in SMC5 have been associated with genetic disorders and cancer.
C2orf88	(Chromosome 2 open reading frame 88)).	It is research is needed to elucidate its role in cellular processes or its potential associations with diseases.
NFE2L2	(Nuclear Factor, Erythroid 2 Like 2)	It is Tau proteins are associated with neurodegenerative diseases, such as Alzheimer's disease, and tubulin proteins are essential components of microtubules involved in various cellular processes..
ZFC3H1	(Zinc Finger CCCH-Type Containing 1)	It encodes a protein that contains multiple zinc finger domains of the CCCH type.
TSSK4	(Testis-Specific Serine Kinase 4)	TSSK4 plays a role in spermatogenesis, the process of sperm cell development.
SPARC	(Secreted Protein Acidic and Rich in Cysteine)	SPARC is found in various tissues and has been associated with several diseases, including cancer. It plays a role in modulating the extracellular matrix.
TRAPPC10	(Trafficking Protein Particle Complex 10)	It is a gene that encodes a subunit of the TRAPP (TRAnsport Protein Particle) complex. The TRAPP complex is involved in intracellular vesicle trafficking and plays a role in protein transport between

		different compartments of the cell.
ID2	(Inhibitor of DNA Binding 2)	It has been implicated in various biological processes, including neurogenesis, hematopoiesis, and cancer development.
TUBB1	(Tubulin Beta 1 Class VI)	TUBB1 is specifically involved in the formation of microtubules and contributes to various cellular processes.
YWHAG	(Tyrosine 3-Monooxygenase/Tryptophan 5-Monooxygenase Activation Protein Gamma)	It proteins play essential roles in numerous cellular processes, including signal transduction, cell cycle control, apoptosis, and protein-protein interactions.
SNAPC1	(Small Nuclear RNA Activating Complex, Polypeptide 1)	SNAPC1 plays a crucial role in the regulation of snRNA transcription and is required for proper splicing and mRNA processing.
CXCL5	(C-X-C Motif Chemokine Ligand 5)	It belongs to the CXC chemokine family, which are small proteins involved in immune responses and inflammation.
DDX11L10	(DEAD/H-Box Helicase 11 Like 10)	DDX11L10 is expressed in multiple tissues, including the brain, heart, and skeletal muscle. accumulated mutations or deletions that render them non-coding.
NFE2L2	(Nuclear Factor, Erythroid 2 Like 2),	It is involved in cellular defense against oxidative stress and controls the expression of numerous genes involved in detoxification, antioxidant defense, and anti-inflammatory processes.
RYBP	(Ring1 and YY1 Binding Protein)	RYBP is known to be involved in various cellular processes, including embryonic development, cell proliferation, and tumorigenesis
ZNF14	(Zinc finger protein 14)	Zinc finger proteins are a class of proteins that can bind to specific DNA sequences and play a role in gene expression regulation.

PPBP	**(Pro-platelet Basic Protein)**	**also known as CXCL7, is a chemokine involved in the regulation of platelet function and hematopoiesis. It promotes platelet activation, aggregation.**
VIL1	**(Villin 1)**	**VIL1 is mainly expressed in the small intestine and colon.**
PAFAH1B2	**(Platelet-activating factor acetylhydrolase 1b, catalytic subunit 2)**	**PAFAH1B2 is primarily expressed in the brain and is involved in neuronal development and function.**
PAXBP1	**(PAX3 and PAX7 Binding Protein 1)**	**PAXBP1 is thought to play a role in the regulation of gene expression during development, but its exact function is still being studied..**
NEU3	**(Neuraminidase 3)**	**Dysregulation of NEU3 expression or activity has been associated with several diseases, including cancer and neurodegenerative disorders.**
TRIM52	**(tripartite motif-containing 52)**	**which is involved in various cellular processes, including immune response .**
FAM123C	**(Family with Sequence Similarity 123C)**	**It is located on chromosome 15**
CCDC103	**(Coiled-Coil Domain Containing 103)**	**It is a human gene that is located on chromosome 17**
EIF1AX-AS1	**(relatively newly discovered lncRNA)**	**studies have suggested that it may function as an oncogene by interacting with other cellular components and modulating gene expression.**

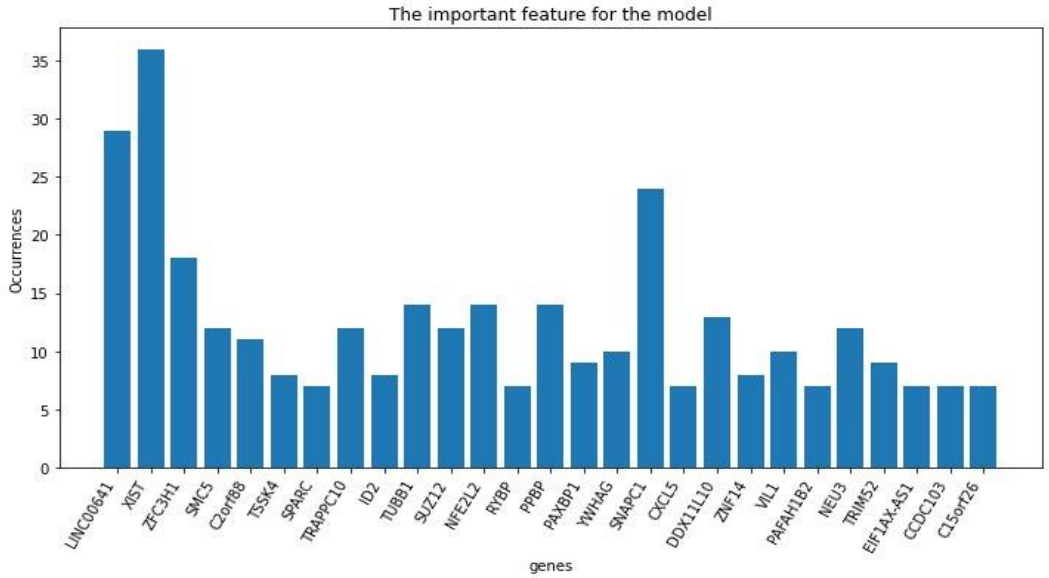


Figure (4.28): The Predicted Gene Names for the model**.**

CHAPTER FIVE

CONCLUSIONS AND FUTUREWORKS

5.1 Conclusions

Several important inferences can be made from the findings of this thesis utilizing machine learning models and feature selection techniques:

1. By comparing the results of parallel, sequential, and hybrid feature selection approaches, we are able to identify a group of genes with useful information. During the parallel feature selection phase, genes like:
   - Parallel feature selection phase: [ZFC3H1, NEU3, C1RL-AS1, SNAPC1, XIST, CCDC103, C15orf26, ATP13A5-AS1, DDX11L10, SMC5, FAM222B, and LINC00641]
   - Sequential feature selection phase: [LINC00641, XIST, ZFC3H1, C2orf88, TRPPC10, ID2, TUBB1, SUZ12, NFE218LK, PPBP, and YWHAG] .
   - Hybrid feature selection phase: [LINC00641, EIF1AX-AS1, TRIM52, NEU3, PAXBP1, ZFC3H1, LOC101929516, SNAPC1, RNF219, XIST, NFE2L2, MIR25, SIAH1, PAFAH1B2, VIL1, and SMC5]
2. Comparing the accuracy of several machine learning models that used sequential, parallel, and hybrid methods for selecting features yielded outstanding overall results.
   - Parallel feature selection phase: Logistic Regression classifier achieved an accuracy score of 1.00, for ANOVA (60%) and For both the (20%) and (60%) feature selection CHI^2 .
   - Sequential feature selection phase: Logistic Regression classifier achieved an accuracy score of 1.00, for MI_CHI^2_ANOVA, MI_ANOVA_CHI^2, CHI^2_MI_ANOVA and CHI^2_MI_ANOVA for (60%).As for both the (20%) and (60%) feature selection ANOVA_MI_CHI^2 and ANOVA_CHI^2_ MI .

- Hybrid feature selection phase: LR and ADA classifier achieved an accuracy score of 1.00, for MI_ RF for (20%).As for both the ANOVA_RF and CHI^2 _RF feature selection with LR classifier for (60%) ,while RF classifier with ANOVA_RF for (60%) .

3. When paired with parallel feature selection procedures such as MI (60 %), MI (20 %), and ANOVA (20 %), the CNN model shown excellent accuracy tendencies. The CNN model's performance on features obtained from various feature selection methods, such as MI_ CHI^2_ ANOVA(20 %), MI_ ANOVA_ CHI^2(20 %), CHI^2_ MI_ ANOVA(20 %), and CHI^2_ ANOVA_ MI(20 %), demonstrated the effectiveness of sequential feature selection in improving system accuracy and predictive abilities. However, when using feature selection techniques such as MI_ RF (60 %), ANOVA_ RF (20 %), and CHI^2 _RF(20 %) in certain situations when hybrid feature selection is utilized, accuracy ratings are improved to 1.00.
4. Genes found utilizing parallel, sequential, and hybrid feature selection methods offered light on the variables involved in immunization efficacy; however, the sequential approach (60%) showed to be the most effective of the three.

5.2 Future Works

Significant discoveries addressing the effects of different vaccines on antibody response and genetic immunity have been revealed by this study, but there are still many questions that may be answered with further study.

1. Exploring the efficacy of different machine learning methods in making vaccination efficacy predictions, like Gradient Boosting Machines (GBM), and Neural Networks.
2. Examining the effects of using alternate feature selection methods, such as Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE), on model performance and feature relevance.

3. In order to assure the generalizability of the findings and to improve the dependability of the models, it is important to validate them using larger and more diverse datasets.
4. Learning more about the genetic pathways that contribute to vaccination efficacy and immune response through in-depth examination and interpretation of the selected features and anticipated gene names.

It is anticipated that improvements in personalized medicine and vaccine production would result from addressing these future research objectives, specifically the prediction of antibody response and genetic immunological variables.

References

[1] A. Adadi, M. Lahmer, and S. Nasiri, "Artificial Intelligence and COVID-19: A Systematic umbrella review and roads ahead," *Journal of King Saud University - Computer and Information Sciences*, no. xxxx, 2021, doi: 10.1016/j.jksuci.2021.07.010.

[2] A. Kumar, P. K. Gupta, and A. Srivastava, "A review of modern technologies for tackling COVID-19 pandemic," *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, vol. 14, no. 4, 2020, doi: 10.1016/j.dsx.2020.05.008.

[3] J. Li *et al.*, "Identification of genes related to immune enhancement caused by heterologous ChAdOx1–BNT162b2 vaccines in lymphocytes at single-cell resolution with machine learning methods," *Front Immunol*, vol. 14, Mar. 2023, doi: 10.3389/fimmu.2023.1131051.

[4] M. Sallam, "Covid-19 vaccine hesitancy worldwide: A concise systematic review of vaccine acceptance rates," *Vaccines*, vol. 9, no. 2. 2021. doi: 10.3390/vaccines9020160.

[5] K. Kricorian, R. Civen, and O. Equils, "COVID-19 vaccine hesitancy: misinformation and perceptions of vaccine safety," *Hum Vaccin Immunother*, vol. 18, no. 1, 2022, doi: 10.1080/21645515.2021.1950504.

[6] J. Groslambert, E. Prokhorova, and I. Ahel, "ADP-ribosylation of DNA and RNA," *DNA Repair (Amst)*, vol. 105, 2021, doi: 10.1016/j.dnarep.2021.103144.

[7] L. Lorente *et al.*, "DNA and RNA Oxidative Damage and Mortality of Patients With COVID-19," *American Journal of the Medical Sciences*, vol. 361, no. 5, 2021, doi: 10.1016/j.amjms.2021.02.012.

[8] J. Li, S. Fong, R. K. Wong, R. Millham, and K. K. L. Wong, "Elitist Binary Wolf Search Algorithm for Heuristic Feature Selection in High-Dimensional Bioinformatics Datasets," *Sci Rep*, vol. 7, no. 1, 2017, doi: 10.1038/s41598-017-04037-5.

[9] J. Shen *et al.*, "Deep learning approach for cancer subtype classification using

high-dimensional gene expression data," *BMC Bioinformatics*, vol. 23, no. 1, Dec. 2022, doi: 10.1186/s12859-022-04980-9.

[10] T. Yu, Z. Huang, and Z. Pu, "Identification of Potential Diagnostic Biomarkers and Biological Pathways in Hypertrophic Cardiomyopathy Based on Bioinformatics Analysis," *Genes (Basel)*, vol. 13, no. 3, 2022, doi: 10.3390/genes13030530.

[11] J. Ahmad, H. Farman, and Z. Jan, "Deep Learning Methods and Applications," in *SpringerBriefs in Computer Science*, 2019. doi: 10.1007/978-981-13-3459-7_3.

[12] H. Z. Almarzouki, "Deep-Learning-Based Cancer Profiles Classification Using Gene Expression Data Profile," *J Healthc Eng*, vol. 2022, 2022, doi: 10.1155/2022/4715998.

[13] T. S. Qaid, H. Mazaar, M. Y. H. Al-Shamri, M. S. Alqahtani, A. A. Raweh, and W. Alakwaa, "Hybrid Deep-Learning and Machine-Learning Models for Predicting COVID-19," *Comput Intell Neurosci*, vol. 2021, 2021, doi: 10.1155/2021/9996737.

[14] S. I. Mallah *et al.*, "COVID-19: breaking down a global health crisis," *Ann Clin Microbiol Antimicrob*, vol. 20, no. 1, p. 35, Dec. 2021, doi: 10.1186/s12941-021-00438-7.

[15] Md. J. Ali, A. B. Bhuiyan, N. Zulkifli, and M. K. Hassan, "The COVID-19 Pandemic: Conceptual Framework for the Global Economic Impacts and Recovery," in *Towards a Post-Covid Global Financial System*, Emerald Publishing Limited, 2022, pp. 225–242. doi: 10.1108/978-1-80071-625-420210012.

[16] "Covid-19 Coronavirus Pandemic," *worldometers*.

[17] T. P. Velavan and C. G. Meyer, "The COVID-19 epidemic," *Tropical Medicine & International Health*, vol. 25, no. 3, pp. 278–280, Mar. 2020, doi: 10.1111/tmi.13383.

[18] S. A. J. Zaidi, S. Tariq, and S. B. Belhaouari, "Future prediction of covid-19

vaccine trends using a voting classifier," *Data (Basel)*, vol. 6, no. 11, Nov. 2021, doi: 10.3390/data6110112.

[19] Z. Zhang, "Genomic Beneets and Potential Harms of COVID-19 Vaccines Indicated from High-Performance Genomic Biomarkers Genomic Benefits and Potential Harms of COVID-19 Vaccines Indicated from High-Performance Genomic Biomarkers," 2022, doi: 10.21203/rs.3.rs-1918598/v1.

[20] H. K. Lee *et al.*, "Heterologous ChAdOx1-BNT162b2 vaccination in Korean cohort induces robust immune and antibody responses that includes Omicron," *iScience*, vol. 25, no. 6, 2022, doi: 10.1016/j.isci.2022.104473.

[21] K. Rezaee, G. Jeon, M. R. Khosravi, H. H. Attar, and A. Sabzevari, "Deep learning-based microarray cancer classification and ensemble gene selection approach," *IET Syst Biol*, vol. 16, no. 3–4, pp. 120–131, May 2022, doi: 10.1049/syb2.12044.

[22] O. Abdelwahab, N. Awad, M. Elserafy, and E. Badr, "A feature selection-based framework to identify biomarkers for cancer diagnosis: A focus on lung adenocarcinoma," *PLoS One*, vol. 17, no. 9 September, Sep. 2022, doi: 10.1371/journal.pone.0269126.

[23] S. Liu and W. Yao, "Prediction of lung cancer using gene expression and deep learning with KL divergence gene selection," *BMC Bioinformatics*, vol. 23, no. 1, Dec. 2022, doi: 10.1186/s12859-022-04689-9.

[24] M. M. Hatmal *et al.*, "Side effects and perceptions following covid-19 vaccination in jordan: A randomized, cross-sectional study implementing machine learning for predicting severity of side effects," *Vaccines (Basel)*, vol. 9, no. 6, Jun. 2021, doi: 10.3390/vaccines9060556.

[25] D. Papadopoulos *et al.*, "Predictive Factors for Neutralizing Antibody Levels Nine Months after Full Vaccination with BNT162b2: Results of a Machine Learning Analysis," *Biomedicines*, vol. 10, no. 2, Feb. 2022, doi: 10.3390/biomedicines10020204.

[26] N. Koul and S. S. Manvi, "Ensemble Feature Selection from Cancer Gene

Expression Data using Mutual Information and Recursive Feature Elimination," in *Proceedings of 2020 3rd International Conference on Advances in Electronics, Computers and Communications, ICAECC 2020*, 2020. doi: 10.1109/ICAECC50550.2020.9339518.

[27] L. Chen *et al.*, "Identifying COVID-19-Specific Transcriptomic Biomarkers with Machine Learning Methods," *Biomed Res Int*, vol. 2021, 2021, doi: 10.1155/2021/9939134.

[28] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, "Improved handwritten digit recognition using convolutional neural networks (Cnn)," *Sensors (Switzerland)*, vol. 20, no. 12, pp. 1–18, Jun. 2020, doi: 10.3390/s20123344.

[29] C. Garcia-Pintos, F. Riet-Correa, and A. Menchaca, "Effect of Foot-and-Mouth Disease Vaccine on Pregnancy Failure in Beef Cows," *Front Vet Sci*, vol. 8, 2021, doi: 10.3389/fvets.2021.761304.

[30] S. Bogdanović-Vasić *et al.*, "Protection of health workers employed in a tertiary health institution from hepatitis b virus infection," *Srp Arh Celok Lek*, vol. 148, no. 11–12, 2020, doi: 10.2298/SARH200419059B.

[31] H. A. Safar, A. S. Mustafa, and T. D. McHugh, "COVID-19 vaccine development: What lessons can we learn from TB?," *Annals of Clinical Microbiology and Antimicrobials*, vol. 19, no. 1. 2020. doi: 10.1186/s12941-020-00402-x.

[32] M. Pacenti *et al.*, "Measles virus infection and immunity in a suboptimal vaccination coverage setting," *Vaccines (Basel)*, vol. 7, no. 4, 2019, doi: 10.3390/vaccines7040199.

[33] H. Singh *et al.*, "Development of the India COVID-19 vaccine tracker," *Indian Journal of Medical Research*, vol. 155, no. 5–6, 2022, doi: 10.4103/ijmr.ijmr_3500_21.

[34] M. Herper, "Covid-19 vaccine from Pfizer and BioNTech is strongly effective, early data from large trial indicate," 2020.

[35] E. M. Agency, “COVID-19 vaccines safety update,” 2022.

[36] M. Voysey *et al.*, “Single-dose administration and the influence of the timing of the booster dose on immunogenicity and efficacy of ChAdOx1 nCoV-19 (AZD1222) vaccine: a pooled analysis of four randomised trials,” *The Lancet*, vol. 397, no. 10277, pp. 881–891, Mar. 2021, doi: 10.1016/S0140-6736(21)00432-3.

[37] D. of H. and S. Care, M. T. MP, and T. R. H. S. J. MP, “One year anniversary of UK deploying Oxford-AstraZeneca vaccine,” *GOV.UK*, 2022.

[38] G. Rahman and C. C. Wen, “Omega Network Pseudorandom Key Generation Based on DNA Cryptography,” *Applied Sciences (Switzerland)*, vol. 12, no. 16, 2022, doi: 10.3390/app12168141.

[39] J. F. Cárdenas-García, “The Central Dogma of Information,” *Information (Switzerland)*, vol. 13, no. 8, 2022, doi: 10.3390/info13080365.

[40] G. P. McGuire, C. V. Luna, E. M. Staehling, and M. E. Stroupe, “From COVID-19 to the Central Dogma,” *Am Biol Teach*, vol. 84, no. 7, 2022, doi: 10.1525/abt.2022.84.7.410.

[41] T. Vaiyapuri, Liyakathunisa, H. Alaskar, E. Aljohani, S. Shridevi, and A. Hussain, “Red Fox Optimizer with Data-Science-Enabled Microarray Gene Expression Classification Model,” *Applied Sciences (Switzerland)*, vol. 12, no. 9, 2022, doi: 10.3390/app12094172.

[42] T. wei Miao *et al.*, “High expression of SPP1 in patients with chronic obstructive pulmonary disease (COPD) is correlated with increased risk of lung cancer,” *FEBS Open Bio*, vol. 11, no. 4, 2021, doi: 10.1002/2211-5463.13127.

[43] P. Wang *et al.*, “The transcriptional characteristics of NADC34-like PRRSV in porcine alveolar macrophages,” *Front Microbiol*, vol. 13, 2022, doi: 10.3389/fmicb.2022.1022481.

[44] A. K. Al-Mashanji and S. Z. Al-Rashi, “Computational Methods for Preprocessing and Classifying Gene Expression Data- Survey,” in *4th Scientific International Conference Najaf, SICN 2019*, 2019. doi:

10.1109/SICN47020.2019.9019349.

[45] A. S. Ahmed, Z. K. Obeas, B. A. Alhade, and R. A. Jaleel, "Improving prediction of plant disease using k-efficient clustering and classification algorithms," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 3, 2022, doi: 10.11591/ijai.v11.i3.pp939-948.

[46] Y. Jiang, "Using Decision Tree Classification and AdaBoost Classification to Build the Abnormal Data Monitoring System of Financial Accounting in Colleges and Universities," *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/1467195.

[47] N. Suga *et al.*, "Matrix Factorization-Based RSS Interpolation for Radio Environment Prediction," *IEEE Wireless Communications Letters*, vol. 10, no. 7, 2021, doi: 10.1109/LWC.2021.3069979.

[48] G. Huang, "Missing data filling method based on linear interpolation and lightgbm," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1754/1/012187.

[49] S. A. J. Zaidi, S. Tariq, and S. B. Belhaouari, "Future prediction of covid-19 vaccine trends using a voting classifier," *Data (Basel)*, vol. 6, no. 11, 2021, doi: 10.3390/data6110112.

[50] H. Henderi, "Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer," *IJIIS: International Journal of Informatics and Information Systems*, vol. 4, no. 1, pp. 13–20, Mar. 2021, doi: 10.47738/ijiis.v4i1.73.

[51] N. D. S. S. Kiran Relangi, A. Chaparala, and R. Sajja, "Effective Groundwater Quality Classification Using Enhanced Whale Optimization Algorithm with Ensemble Classifier," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 1, pp. 214–223, 2023, doi: 10.22266/ijies2023.0228.19.

[52] J. Ravničan *et al.*, "A Prestudy of Machine Learning in Industrial Quality Control Pipelines," *Informatica (Slovenia)*, vol. 46, no. 2, 2022, doi:

10.31449/inf.v46i2.3938.

[53] K. T. Fang and C. H. Ping, "Using Machine Learning to Explore the Crucial Factors of Assistive Technology Assessments: Cases of Wheelchairs," *Healthcare (Switzerland)*, vol. 10, no. 11, 2022, doi: 10.3390/healthcare10112238.

[54] C. Li, S. Wei, X. Xu, and X. Qu, "Modelling of Critical Acceleration for Regional Seismic Landslide Hazard Assessments by Finite Element Limit Analysis," *Front Earth Sci (Lausanne)*, vol. 10, 2022, doi: 10.3389/feart.2022.830371.

[55] W. Zhongxin, S. Gang, Z. Jing, and Z. Jia, "Feature Selection Algorithm Based on Mutual Information and Lasso for Microarray Data," *Open Biotechnol J*, vol. 10, no. 1, pp. 278–286, Nov. 2016, doi: 10.2174/1874070701610010278.

[56] H. Zhou, X. Wang, and R. Zhu, "Feature selection based on mutual information with correlation coefficient," *Applied Intelligence*, vol. 52, no. 5, pp. 5457–5474, Mar. 2022, doi: 10.1007/s10489-021-02524-x.

[57] B. Thakur, N. Kumar, and G. Gupta, "Machine learning techniques with ANOVA for the prediction of breast cancer," *International Journal of Advanced Technology and Engineering Exploration*, vol. 9, no. 87, 2022, doi: 10.19101/IJATEE.2021.874555.

[58] M. Kumar, N. K. Rath, A. Swain, and S. K. Rath, "Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor," in *Procedia Computer Science*, 2015. doi: 10.1016/j.procs.2015.06.035.

[59] M. Alassaf and A. M. Qamar, "Improving Sentiment Analysis of Arabic Tweets by One-way ANOVA," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, 2022, doi: 10.1016/j.jksuci.2020.10.023.

[60] K. Chadaga, S. Prabhu, N. Sampathila, R. Chadaga, S. Swathi, and S. Sengupta, "Predicting cervical cancer biopsy results using demographic and epidemiological parameters: a custom stacked ensemble machine learning

approach," *Cogent Eng*, vol. 9, no. 1, 2022, doi: 10.1080/23311916.2022.2143040.

[61] S. K. Dey, K. M. M. Uddin, H. M. H. Babu, M. M. Rahman, A. Howlader, and K. M. A. Uddin, "Chi2-MI: A hybrid feature selection based machine learning approach in diagnosis of chronic kidney disease," *Intelligent Systems with Applications*, vol. 16, Nov. 2022, doi: 10.1016/j.iswa.2022.200144.

[62] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved Chi-square for Arabic text classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 2, pp. 225–231, Feb. 2020, doi: 10.1016/j.jksuci.2018.05.010.

[63] T. Elemam and M. Elshrkawey, "A Highly Discriminative Hybrid Feature Selection Algorithm for Cancer Diagnosis," *Scientific World Journal*, vol. 2022, 2022, doi: 10.1155/2022/1056490.

[64] T. Agbele, B. Ojeme, and R. Jiang, "Application of local binary patterns and cascade AdaBoost classifier for mice behavioural patterns detection and analysis," in *Procedia Computer Science*, Elsevier B.V., 2019, pp. 1375–1386. doi: 10.1016/j.procs.2019.09.308.

[65] J. K. Tsai and C. H. Hung, "Improving adaboost classifier to predict enterprise performance after covid-19," *Mathematics*, vol. 9, no. 18, Sep. 2021, doi: 10.3390/math9182215.

[66] B. C. Kim, J. Kim, I. Lim, D. H. Kim, S. M. Lim, and S. K. Woo, "Machine learning model for lymph node metastasis prediction in breast cancer using random forest algorithm and mitochondrial metabolism hub genes," *Applied Sciences (Switzerland)*, vol. 11, no. 7, Apr. 2021, doi: 10.3390/app11072897.

[67] V. Monfared and A. Hashemi, "Prediction Analysis of Preterm Neonates Mortality using Machine Learning Algorithms via Python Programming", doi: 10.1101/2023.01.20.524905.

[68] H. Fei *et al.*, "Cotton Classification Method at the County Scale Based on Multi-Features and Random Forest Feature Selection Algorithm and Classifier,"

Remote Sens (Basel), vol. 14, no. 4, Feb. 2022, doi: 10.3390/rs14040829.

[69] J. X. Liang, J. F. Zhao, N. Sun, and B. J. Shi, "Random Forest Feature Selection and Back Propagation Neural Network to Detect Fire Using Video," *J Sens*, vol. 2022, 2022, doi: 10.1155/2022/5160050.

[70] M. Ram, A. Najafi, and M. T. Shakeri, "Classification and biomarker genes selection for cancer gene expression data using random forest," *Iran J Pathol*, vol. 12, no. 4, pp. 339–347, Sep. 2017, doi: 10.30699/ijp.2017.27990.

[71] A. S. M. Shafi, M. M. I. Molla, J. J. Jui, and M. M. Rahman, "Detection of colon cancer based on microarray dataset using machine learning as a feature selection and classification techniques," *SN Appl Sci*, vol. 2, no. 7, 2020, doi: 10.1007/s42452-020-3051-2.

[72] J. Bowlee, "Logistic Regression for Machine Learning," *Machine Learning Mastery*, 2016.

[73] R. D. Joshi and C. K. Dhakal, "Predicting type 2 diabetes using logistic regression and machine learning approaches," *Int J Environ Res Public Health*, vol. 18, no. 14, 2021, doi: 10.3390/ijerph18147346.

[74] E. Y. Boateng and D. A. Abaye, "A Review of the Logistic Regression Model with Emphasis on Medical Research," *Journal of Data Analysis and Information Processing*, vol. 07, no. 04, 2019, doi: 10.4236/jdaip.2019.74012.

[75] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep Learning-Based Text Classification," *ACM Computing Surveys*, vol. 54, no. 3. 2021. doi: 10.1145/3439726.

[76] Y. Zhao, Q. Chen, W. Cao, W. Jiang, and G. Gui, "Deep Learning Based Couple-like Cooperative Computing Method for IoT-based Intelligent Surveillance Systems," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, 2019. doi: 10.1109/PIMRC.2019.8904229.

[77] H. Greenspan, B. Van Ginneken, and R. M. Summers, "Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New

Technique," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5. 2016. doi: 10.1109/TMI.2016.2553401.

[78] F. J. Díaz-Pernas, M. Martínez-Zarzuela, D. González-Ortega, and M. Antón-Rodríguez, "A deep learning approach for brain tumor classification and segmentation using a multiscale convolutional neural network," *Healthcare (Switzerland)*, vol. 9, no. 2, Feb. 2021, doi: 10.3390/healthcare9020153.

[79] F. Alrasheedi, X. Zhong, and P. C. Huang, "Padding Module: Learning the Padding in Deep Neural Networks," *IEEE Access*, vol. 11, pp. 7348–7357, 2023, doi: 10.1109/ACCESS.2023.3238315.

[80] C. Edwin Singh and S. Maria Celestin Vigila, "WOA-DNN for Intelligent Intrusion Detection and Classification in MANET Services," *Intelligent Automation and Soft Computing*, vol. 35, no. 2, pp. 1737–1751, 2023, doi: 10.32604/iasc.2023.028022.

[81] S. A. Alazawi and M. N. Al Salam, "Evaluation of LMT and DNN Algorithms in Software Defect Prediction for Open-Source Software," in *Advances in Intelligent Systems and Computing*, 2021. doi: 10.1007/978-981-15-7527-3_19.

[82] J. Baek and Y. Choi, "Deep neural network for predicting ore production by truck-haulage systems in open-pit mines," *Applied Sciences (Switzerland)*, vol. 10, no. 5, 2020, doi: 10.3390/app10051657.

[83] V. Sharma, M. Gupta, A. K. Pandey, D. Mishra, and A. Kumar, "A Review of Deep Learning-based Human Activity Recognition on Benchmark Video Datasets," *Applied Artificial Intelligence*, vol. 36, no. 1, 2022, doi: 10.1080/08839514.2022.2093705.

[84] J. Sui, M. X. Liu, J. H. Lee, J. Zhang, and V. Calhoun, "Deep learning methods and applications in neuroimaging," *Journal of Neuroscience Methods*, vol. 339. 2020. doi: 10.1016/j.jneumeth.2020.108718.

[85] D. Jung and Y. Choi, "Systematic review of machine learning applications in mining: Exploration, exploitation, and reclamation," *Minerals*, vol. 11, no. 2. 2021. doi: 10.3390/min11020148.

[86] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Trans Neural Netw Learn Syst*, pp. 1–21, Jun. 2021, doi: 10.1109/tnnls.2021.3084827.

[87] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mech Syst Signal Process*, vol. 151, 2021, doi: 10.1016/j.ymssp.2020.107398.

[88] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "Evolving Deep Convolutional Neural Networks for Image Classification," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, 2020, doi: 10.1109/TEVC.2019.2916183.

[89] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif Intell Rev*, vol. 53, no. 8, 2020, doi: 10.1007/s10462-020-09825-6.

[90] R. A. Pratiwi, S. Nurmaini, D. P. Rini, M. N. Rachmatullah, and A. Darmawahyuni, "Deep ensemble learning for skin lesions classification with convolutional neural network," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 3, 2021, doi: 10.11591/ijai.v10.i3.pp563-570.

[91] M. A. Gómez-Guzmán *et al.*, "Classifying Brain Tumors on Magnetic Resonance Imaging by Using Convolutional Neural Networks," *Electronics (Switzerland)*, vol. 12, no. 4, 2023, doi: 10.3390/electronics12040955.

[92] V. H. Phung and E. J. Rhee, "A High-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets," *Applied Sciences (Switzerland)*, vol. 9, no. 21, 2019, doi: 10.3390/app9214500.

[93] H. Gu, Y. Wang, S. Hong, and G. Gui, "Blind channel identification aided generalized automatic modulation recognition based on deep learning," *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2934354.

[94] P. Xu, Z. Guo, L. Liang, and X. Xu, "MSF-net: Multi-scale feature learning network for classification of surface defects of multifarious sizes," *Sensors*, vol. 21, no. 15, 2021, doi: 10.3390/s21155125.

[95] S. A. Singh, T. G. Meitei, and S. Majumder, "Short PCG classification based on deep learning," in *Deep Learning Techniques for Biomedical and Health Informatics*, 2020. doi: 10.1016/B978-0-12-819061-6.00006-9.

[96] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, "Improved handwritten digit recognition using convolutional neural networks (Cnn)," *Sensors (Switzerland)*, vol. 20, no. 12, 2020, doi: 10.3390/s20123344.

[97] R. D. Rakshit, D. R. Kisku, P. Gupta, and J. K. Sing, "Cross-resolution face identification using deep-convolutional neural network," *Multimed Tools Appl*, vol. 80, no. 14, 2021, doi: 10.1007/s11042-021-10745-y.

[98] A. Kost, W. A. Altabey, M. Noori, and T. Awad, "Applying neural networks for tire pressure monitoring systems," *SDHM Structural Durability and Health Monitoring*, vol. 13, no. 3, 2019, doi: 10.32604/sdhm.2019.07025.

[99] S. H. Wang, J. Hong, and M. Yang, "Sensorineural hearing loss identification via nine-layer convolutional neural network with batch normalization and dropout," *Multimed Tools Appl*, vol. 79, no. 21–22, 2020, doi: 10.1007/s11042-018-6798-3.

[100] X. Zhang, Y. Wang, N. Zhang, D. Xu, and B. Chen, "Research on scene classification method of high-resolution remote sensing images based on RFPNet," *Applied Sciences (Switzerland)*, vol. 9, no. 10, 2019, doi: 10.3390/app9102028.

[101] Z. Chen, Q. Xue, Y. Wu, S. Shen, Y. Zhang, and J. Shen, "Capacity prediction and validation of lithium-ion batteries based on long short-term memory recurrent neural network," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3025766.

[102] D. Yi, J. Ahn, and S. Ji, "An effective optimization method for machine learning based on ADAM," *Applied Sciences (Switzerland)*, vol. 10, no. 3, 2020, doi: 10.3390/app10031073.

[103] W. E. L. Ilboudo, T. Kobayashi, and K. Sugimoto, "Robust Stochastic Gradient Descent with Student-t Distribution Based First-Order Momentum," *IEEE*

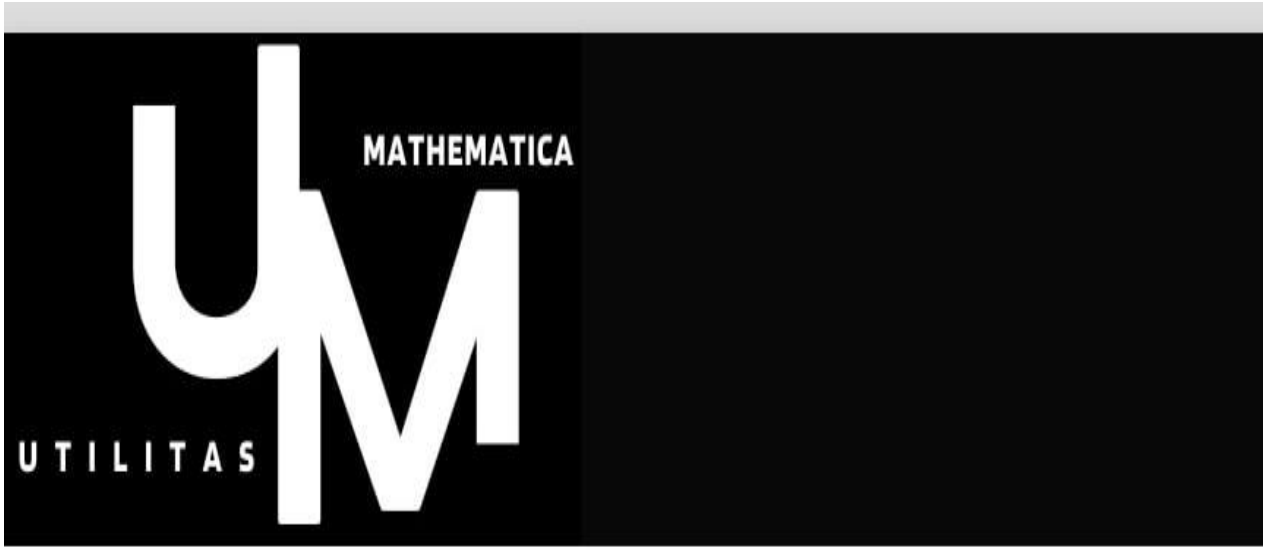
Trans Neural Netw Learn Syst, vol. 33, no. 3, 2022, doi: 10.1109/TNNLS.2020.3041755.

[104] R. Tiwari, "Stabilizing the training of deep neural networks using Adam optimization and gradient clipping," *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 07, no. 01, 2023, doi: 10.55041/ijsrem17594.

[105] N. Ali, D. Neagu, and P. Trundle, "Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets," *SN Appl Sci*, vol. 1, no. 12, 2019, doi: 10.1007/s42452-019-1356-9.

Appendix A

The Published Paper



Effective of Various Vaccines on Antibody Response and Genetic Immune Using Deep Learning Method

Nuha Hisham Mohammed
Department of Information Technology, University of Babylon, Babylon, Iraq, Iraq

Sura Zaki Al-Rashid
Department of Information Technology, University of Babylon, Babylon, Iraq, Iraq

PDF

PUBLISHED

2023-06-15

Appendix B

The Accepted Paper

International Conference
on Engineering Science
and Advanced Technology

IEEE Iraq Section

NORTHERN TECHNICAL UNIVERSITY
Technical Engineering College / Mosul

Date: 15 June 2023

FORMAL ACCEPTANCE AND INVITATION LETTER

Paper ID: **1570897905**
Date: **21-22 June, 2023**

Dear respected author(s):

Nuha Husham
Software department, College of Information Technology, University of Babylon

Sura zaki
Software department, College of Information Technology, University of Babylon

It is our pleasure to inform you that based on the reviewers feedback of (ICESAT2023), your submitted paper entitled: " **Effects of Different Vaccines on Genetic Immune and Antibody Response Using Machine Learning Methods** " has been accepted for oral presentation in this conference.
Note: This letter should serve in obtaining your institute permission and fund to attend the meeting. Also, the letter should help in obtaining a visa to attend the meeting, if required.
Thanks for your interest to participate in this large worldwide venue.

Prof. Dr. Aylaa Al-Attar
CONFERENCE CHAIR

Asst. Prof. Dr. Majid K. Najim
CONFERENCE CO-CHAIR

ICESAT

www.icesat.org
info@icesat.org
+964 770 111 7489

المستخلص

في مجال علم المناعة والوقاية من الأمراض، تعتبر اللقاحات ضرورية لنقل المناعة ضد الكائنات الدقيقة والسموم الضارة. عادةً ما تتألف هذه العوامل المعززة للمناعة من بروتينات أو ببتيدات تعرف بالمستضدات، والتي تحفز إنتاج الأجسام المضادة للدفاع ضد الغزاة المحتملين. فهم مكونات اللقاحات أمر بالغ الأهمية لتقدم هذا المجال ولضمان الصحة العامة. هذا العمل يتناول مشكلة تحديد الجينات المعلومة المتعلقة باستجابة اللقاح. من خلال أساليب مختارة لاختيار السمات بشكل متوازي وتتابعي وهجين، ثم يتم مقارنة فعاليتها في اختيار الجينات ذات الصلة. تسلط مرحلة اختيار السمات المتوازية والهجينة الضوء على ١٦ جينًا، بينما تميز اختيار السمات التتابعي ١١ جينًا.

بالإضافة إلى ذلك، يتم تقييم أداء نماذج تعلم الآلة المختلفة باستخدام السمات المختارة. أسفرت مرحلة اختيار السمات المتوازية عن نتائج مثيرة، حيث حققت نمذجة التحليل اللوجستي درجة دقة مثالية بنسبة ١,٠٠ عند استخدام اختبار الانحدار التحليلي للتباين (٦٠٪) وكلا اختيارات CHI2 عند ٢٠٪ و ٦٠٪. وبالمثل، في مرحلة اختيار السمات التتابعي، حققت نمذجة التحليل اللوجستي أيضًا درجة دقة مثالية بنسبة ١,٠٠ لـ ٦ مسارات عند ٦٠٪. وعلاوة على ذلك، أسفرت مرحلة اختيار السمات الهجينة عن درجة دقة بنسبة ١,٠٠ باستخدام نمذجة التحليل اللوجستي ونمذجة ADA مع اختبار MI_RF (20%) وأيضًا مع اختبار التحليل اللوجستي ونمذجة RF مع اختبار ANOVA_RF عند (٦٠٪). وأخيرًا، اختبار CHI2_RFعند (٦٠٪) مع نمذجة التحليل اللوجستي أيضًا يحقق درجة دقة بنسبة ١,٠٠.

بالإضافة إلى ذلك، لتحسين الكفاءة والدقة، تم دمج نموذج شبكة العصبي الاصطناعي المتسلسل (CNN) مع تقنيات اختيار السمات المتوازية مثل MI (20%) و MI (60%) و

ANOVA (٢٠٪). هذا الدمج يكشف عن تحسين ملحوظ في الدقة. أظهر أداء نموذج CNN على السمات المحصل عليها من مختلف أساليب اختيار السمات، بما في ذلك أربع مسارات بنسبة ٢٠٪، وأكد أن اختيار السمات التتابعي يعزز دقة النظام وقدراته التنبؤية. وأساليب اختيار السمات الهجينة، مثل MI_RF (٦٠٪) و ANOVA_RF (٢٠٪) و CHI2_RF (٢٠٪)، تعزز دقة النظام إلى ١,٠٠.

للتحقق من موثوقية النموذج وقدرته على التعميم، تم استخدام مجموعة من البيانات المستقلة لم تكن قد تم رؤيتها مسبقًا من قبل النظام للغرض من الاختبار. تضمنت هذه الطريقة اختيار عشوائي لنسبة ١٠٪ من مجموعة البيانات الأصلية، والتي كانت جديدة تمامًا بالنسبة للنظام. تم الاحتفاظ بهذه المجموعة الجديدة بشكل منفصل عن بيانات التدريب، والتي بلغت ٩٠٪ من مجموعة البيانات الأصلية. تم تقسيم بيانات التدريب إلى جزء يبلغ ٧٠٪ لتدريب النموذج وجزء يبلغ ٣٠٪ لاختبار النموذج. تم بعد ذلك تقييم أداء النموذج على البيانات الجديدة التي بلغت نسبتها ١٠٪ وأظهرت النتائج دقة ونجاح النموذج.

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة بابل
قسم برمجيات تكنولوجيا المعلومات

تحليل الاستجابة المناعية والأجسام المضادة الوراثية باستخدام نموذج التعلم الآلي للقاحات مختلفة

رسالة مقدمة الى

مقدم إلى مجلس كلية تكنلوجيا المعلومات- جامعة بابل في استيفاء جزئي لمتطلبات درجة الماجستير في تكنولوجيا المعلومات / البرمجيات

من قبل

نهى هشام محمد عبد الرحمن

باشراف

الاستاذ المساعد الدكتورة سرى زكي ناجي

1445 هـ **2023 م**