BACHELOR OF SCIENCE IN ELECTRONIC AND TELECOMMUNICATION ENGINEERING

# Predictive Analytics: Anemia Disease Forecasting with Machine Learning and Deep Learning Models


**Submitted by**

Muhammad Shamim
T-183018

**Supervised by**

Muhammad Mostafa Amir Faisal
Assistant Professor
Department of ETE

Department of Electronic and Telecommunication Engineering
International Islamic University Chittagong
Kumira, Sitakunda, Chittagong - 4318



**February - 2024**

# DECLARATION

I do hereby declare that the work presented in this undergraduate thesis has been carried out by me and has not been previously submitted to any other institution or an academic qualification or degree.

**Student's Signature:**

01._____

Muhammad Shamim

**Supervisor's Signature:**

_____

**Muhammad Mostafa Amir Faisal**

Assistant Professor

Department of Electronic and Telecommunication (ETE)

International Islamic University Chittagong (IIUC)

(ETE)

International Islamic University Chittagong (IIUC)

# CERTIFICATE OF APPROVAL

The thesis entitled as "**Predictive Analytics: Anemia Disease Forecasting with Machine Learning and Deep Learning Models**" submitted by Muhammad Shamim, bearing Metric ID. T-183018 respectively of session Autumn 18, to the Department of Electronic and telecommunication Engineering, International Islamic University Chittagong, has been accepted as satisfactory in partial fulfilment of the requirements for the degree of Bachelor of Science in Engineering and approved for the examination held on 2nd February, 2024.

Supervisor

Muhammad Mostafa Amir Faisal
Assistant Professor
Department of Electronic and Telecommunication Engineering
International Islamic University Chittagong

# ACKNOWLEDGEMENT

I commence by conveying my profound gratitude and humble submission to the one and only almighty Allah. Without Allah's kindness and blessings, I would not be able to complete my research within the given time. I am grateful my family for their love, encouragement, and support. This was a big assistance to me.

In addition, I would like to convey our heartfelt gratitude to my supervisor, **Muhammad Mostafa Amir Faisal**, Assistant Professor in the department of Electronics and Telecommunications Engineering (ETE) at the International Islamic University of Chittagong (IIUC), for his insightful comments and creative suggestions. His continuous assistance and thoughtful advice played a pivotal role in choosing the thesis topic and ensuring its timely completion under his guidance.

Special appreciation goes to **Engr. Mohammad Jashim Uddin**, Associate Professor and thesis/project convener, for his invaluable advice and guidance in obtaining data from **Parkview Hospital Limited**. Gratitude is also extended to **Parkview Hospital Limited** for their generosity in allowing me to collect essential data from their Haematology department, a pivotal contribution to my research.

At last, I would relish thanking all the faculty members, seniors, friends, juniors, well-wishers, and lab assistants who have supported and helped me in every possible way. This achievement wouldn't have been possible without the collective help and encouragement from all those involved. I am really blessed to have kind people like them right next to me.

# ABSTRACT

Remarkable breakthroughs in medical research are creating important information that we utilize every day. To gain appropriate details for analysis, prediction, creating suggestions, and establishing choices, this data has to be analyzed. Turn present data into information by applying data mining or machine learning approaches. Accurate illness forecasting is vital in medicine for both preventative and efficient treatment planning. On event, a lack of precision might be deadly. In order to anticipate anemia, this work investigates several machine learning (ML) classification strategies in a large dataset to diagnose anemia , and the performance of these algorithms is confirmed using measurements such as error rate, accuracy, precision, recall, and F-Measure. Strategies were tried in the experiment, and it was determined that Random forest functioned better than any ML methodology, with the maximum accuracy of 100 percent when compared to other algorithms.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF ILLUSTRATIONS

# LIST OF ABBREVIATIONS

| | |
|---|---|
| CNN | Convolutional Neural Network |
| RNN | Recurrent Neural Networks |
| IIUC | International Islamic University Chittagong |
| ETE | Electronics and Telecommunications Engineering |
| TPR | True Positive Rate |
| FPR | False Positive Rate |
| ML | Machine Learning |
| DL | Deep Learning |
| ROC | Receiver Operating Curve |
| SVM | Support Vector Machine |
| K-NN | K-Nearest Neighbours |
| AI | Artificial Intellligence |
| GPU | Graphics Processing Units |

# CHAPTER I

# INTRODUCTION

## 1.1 Introduction

Advanced hospitals create large volumes of info every day. This data has to be processed and analyzed with the goal to identify undetected trends and retrieve vital details. Data analysis is a means of uncovering innovative connections in data acquired from multiple sources. Various machine learning algorithms are utilised in diverse domains such as healthcare, meteorology, stock market forecasting, and product suggestioning to predict future events. Forecasting various diseases and the factors that contribute to them is an important element of medical research. Health data is used in medicine to anticipate pandemics, diagnose illnesses, improve quality of life, and prevent premature death. In this assignment, we'll study 5 new definition strategies for forecasting. Anaemia is a condition in which you do not have enough healthy blood vessels to transfer adequate oxygen to your body's tissues. Anaemia, which is characterised by low haemoglobin levels, can produce weariness and weakness. There are various types of anaemia, each with its own cause [1].

## 1.2 Anemia

Light anemia is a frequent and curable illness that may occur in anybody. It might come about abruptly or over a long time, and could be triggered by your food, drugs you implement, or another health issue. Anemia can be additionally long-term, indicating it lasts for an extended time & can't be gone entirely. Some kinds of anemia are hereditary. The most prevalent kind of anemia is a lack of iron. Some individuals are at a greater risk for anemia, especially women between their monthly periods and childbirth. persons who fail to get sufficient iron or certain nutrients.

Anaemia can additionally be an indication of another more severe illness, such as haemorrhage in the intestines, discomfort from an illness, kidney failure, malignancy, or autoimmunity. Physicians will utilize your medical data, examination of your body, and test findings to evaluate anaemia. Certain types of mild to severe anaemia may necessitate the use of iron supplements, vitamins, or medications that assist your body in producing more red blood cells. To avoid haemoglobin in subsequent years, your physician may also prescribe nutritious food adjustments.

Anaemia is a health condition marked by insufficient healthy blood vessels to effectively transport oxygen to the body's tissues. It is identified by low haemoglobin levels, leading to fatigue and weakness. Multiple types of anaemia exist, each originating from distinct causes. Identifying the precise type is essential for targeted interventions, ensuring effective restoration of normal blood function and alleviation of symptoms associated with this condition.

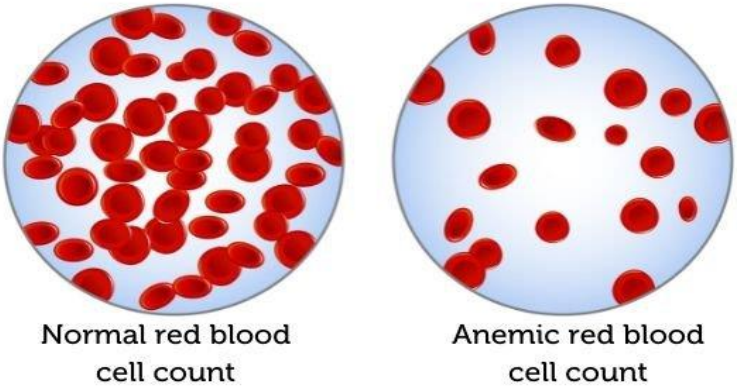


Figure 1.1 Anemia [2]

The signs of anemia vary on what is causing it as well as how serious the condition is. Hematology might be mild enough that it generates little discomfort at first. But symptoms normally then start and grow severe as the hemoglobin gets severe. If another condition produces the anemia, the sickness might disguise the anemic manifestations. Then an examination for another illness could discover the anemia. Certain kinds of anemia feature indications which direct attention to the cause [2].

### *1.2.1 Possible Signs of Anemia Include*

- ✓ Fatigue.
- ✓ Vulnerability.
- ✓ Lack of breath.
- ✓ Light or pale skin, which could be more visible on white skin compared on black or brown skin.
- ✓ Abnormal heartbeat.
- ✓ Feeling dizzy or sensations of lightheaded.
- ✓ Chest discomfort.
- ✓ Cold fingers and toes.
- ✓ Migraines.

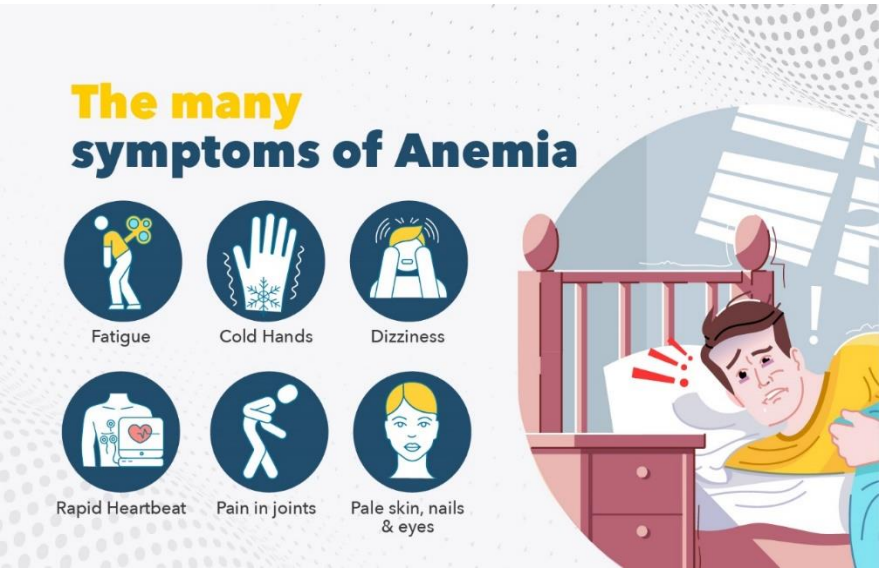


Figure 1.2 Anemia Symptoms [2]

Low levels of the protein present in blood arteries that carry oxygen consumption, termed hemoglobin, or is the major symptom of anemia. Some individuals find they have inadequate hemoglobin levels when they give blood. If you're informed that you simply can't donate due of insufficient hemoglobin, arrange a medical consultation. Anemia develops when a person's blood lacks sufficient amounts of hemoglobin or blood vessel cells. The human body doesn't create sufficient hemoglobin or blood vessel cells. Bleeding induces loss of white blood cells and proteins quicker than they are capable of replenished. The body kills blood vessels and the hemoglobin which is in them [2].

## 1.3 Status of Anemia Disease Among People

While most varieties of anemic are moderate and readily cleaned up, but these are other kinds of anemic that are serious persistent &/or serious. It is calculated that 24.8 percent (1.62 billion humans) of worldwide individuals is struggling from fatigue whereas determined anemia The incidence is 47.4% of those (293 a million) within preschool-aged kids which at first is greatest in specific age categories, 41.8 percent (56 million dollars) of pregnancies women as well as thirty-two percent (468 million) of non-pregnant women are getting from anemia. Children ages six to twelve years are at a raised danger of anemia due to they are establishing and building swiftly and considering the kept metal to the women may be inadequate, hence the furtherance of supplementary food throughout this time is essential and supplementary meals affect

the as a whole dietary status of the child [3] . Infancy anemic has a high link with interpersonal, economical, mental in nature and health related difficulties and it is established that premature children anemia is a good prediction of mature anemia. In the current research, we had followed the criterion of International Health Organisation (WHO) for identifying anemia, that is described by the accompanying **TABLE 1.1.**

TABLE 1.1 WHO CRITERIA FOR ANEMIA AND GRADE OF SEVERITY [3]

| **S. No.** | **Population** | **Non-Anemia (gm/dL)** | **Anemia (gm/dL)** | | |
|---|---|---|---|---|---|
| | | | ***Mild*** | ***Moderate*** | ***Severe*** |
| 1. | Children 6-59 months of age | 11 | 10.0-10.9 | 7.0-9.9 | <7.0 |
| 2. | Children 5-11 years of age | 11.5 | 11.0-11.4 | 8.0-10.9 | <8.0 |
| 3. | Children 12-14 years of age | 12 | 11.0-11.9 | 8.0-10.9 | <8.0 |
| 4. | Non-pregnant women (15 years of age and above) | 12 | 11.0-11.9 | 8.0-10.9 | <8.0 |
| 5. | Pregnant women | 11 | 10.0-10.9 | 7.0-9.9 | <7.0 |
| 6. | Men (15 years of age and above) | 13 | 11.0-12.9 | 8.0-10.9 | <8.0 |

Anemia poses significant repercussions on school-age children, manifesting as poor psychomotor development, lasting negative effects on the central nervous system, diminished IQ, suboptimal school performance, reduced work capacity, and an overall diminished quality of life [4]. Various factors contribute to childhood anemia, such as

deficiencies in folate, vitamin B12, and other vitamins, infections like malaria, parasitic infections, and hemoglobinopathies. Numerous regional and national studies have explored the determinants of anemia, highlighting demographic, social, environmental, and geographic factors. Younger age, male sex, maternal age and education, maternal anemia, malnutrition (especially stunting), insufficient daily meals, parasitic infection, recent diarrhea, fever, and absence of deworming are identified as significant risk factors for childhood anemia.

Childhood anemia not only causes considerable morbidity and mortality but also leads to long-lasting and possibly irreversible consequences. Severe anemia contributes significantly to overall under-five mortality, with hidden morbidity and mortality occurring months after initial diagnosis and treatment. Despite efforts such as the national anemia control program, the prevalence of anemia continues to rise, raising concerns due to its diverse impact on cognitive functions.

International organizations, including WHO, UNICEF, NFHS, Govt. of India, and NGOs, have undertaken strategies to combat anemia, such as iron fortification, iron supplements, deworming for school children, mid-day meal programs, and nutritional education. However, the goal of reducing anemia prevalence remains a challenge. Infants, children under 5 years old, and pregnant women are particularly susceptible to anemia due to increased iron requirements associated with rapid body growth and expansion of red blood cells [5]. Despite ongoing efforts, achieving the goal of reducing anemia prevalence remains a persistent challenge.

## 1.4 Machine Learning and Deep Learning

Machine learning (ML) is a branch of research in artificial intelligence focused on the creation and investigation of statistical methods that can successfully adapt and so accomplish jobs without given directions. In recent years, generating neural networks made from computers were able to outperform several earlier techniques in efficiency. Machine learning methodologies have been used to huge models for language, artificial intelligence, audio recognition, email sorting, farming and health, when it is too expensive to design algorithms to execute the desired tasks. The mathematical underpinnings of ML are given by numerical optimization (mathematical programming) approaches. Data extraction is an identical (parallel) area of research, concentrating on interactive inquiry into data via independent learning. ML is renowned

in its use throughout company issues with the moniker analytical prediction. While not all artificial intelligence is quantitatively oriented, statistical computing is an important component of the area's approaches [6].

### *1.4.1 Logistic Regression*

A type of trained learning known as logistic regression (LR) uses the sigmoid function to assess probabilities in order to determine the relationship between two binary variables that is dependent with at least a single independent variable compared to its designation, logistic regression (LR) cannot be utilized for resolving a regression analysis issues; actually it is an example of machine learning categorization issues where the variable being studied can be binominal, ordinal, period, or ratio-level and the variable that depends on it is a binary value (0/1, - 1/1, true/false) [7].

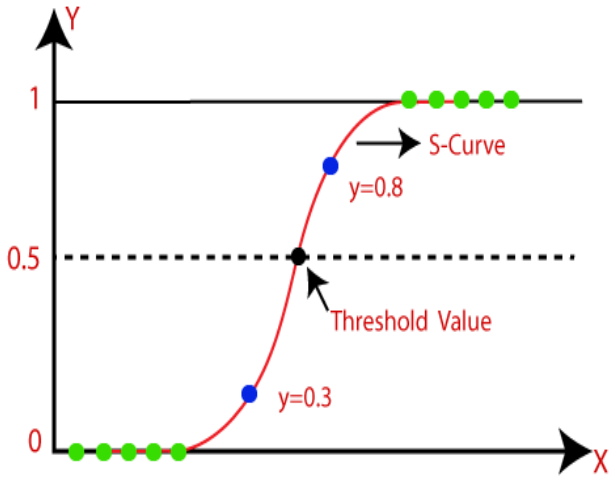


Figure 1.3 Logistic Regression [7]

### *1.4.2 Naive Bayes*

In many real-world contexts, such as recognizing documents and filtering out spam, inexperienced Bayes classifiers perform wonderfully. A bayesian machine learning methodology dubbed the Naive Bayes categorization method depends on probability theory's Bayes principle. It is one of the finest classifier due to its simplicity, it performs better than other classifiers. It does not require as much information about training. It handles data that is both continuous and discrete. When considering the quantity of predictors and data points, it is incredibly scalable. It is fast and capable of generating projections in real time. A stochastic machine learning framework called the Naive Bayes classifier is used for classification tasks [7].

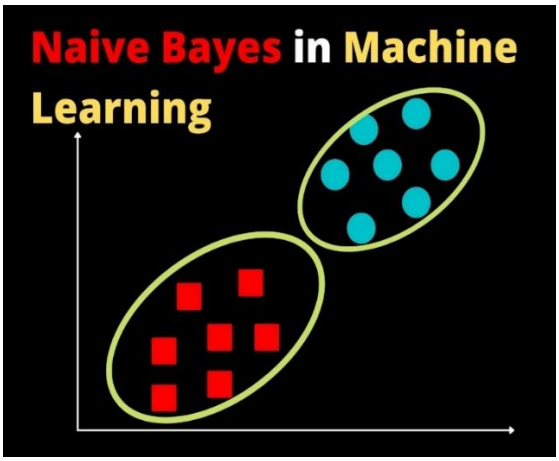


Figure 1.4 Naive Bayes [7]

### *1.4.3 K-Nearest Neighbors*

The k-nearest neighbors (KNN) approach, a fundamental technique in supervised machine learning, employs a straightforward majority vote from the k closest companions to determine categorization. Widely utilized for solving classification challenges, KNN extends its utility to address regression-related issues. Its main strengths lie in quick computation and ease of translation, making it a popular choice for companies grappling with diverse data-driven problems. The simplicity and comprehensibility of the KNN method contribute to its widespread adoption, as it is relatively easy to create and understand. However, a key drawback emerges with the method's tendency to slow down as the quantity of data in use increases, posing a challenge in scalability. Despite this limitation, KNN remains a significant asset, striking a balance between simplicity and effectiveness in addressing a spectrum of machine learning challenges across both classification and regression domains, thereby solidifying its enduring relevance in the field [7].

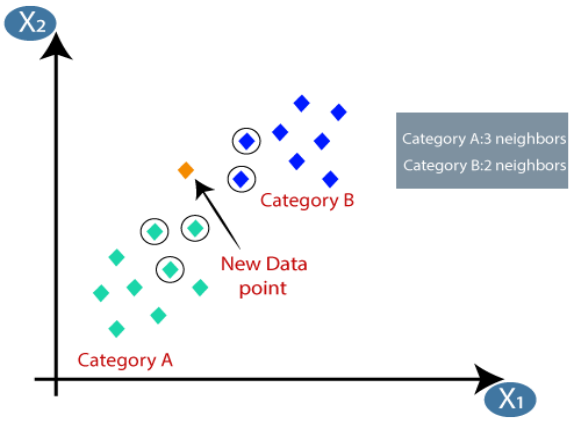


Figure 1.5 K-Nearest Neighbors [7]

### *1.4.4 Random Forest*

The random forest classifier, a meta-estimator, plays a pivotal role in aligning distinct decision trees on various subsamples of information sets. By employing averaging, it enhances the model's predictive efficacy and effectively mitigates overfitting. Notably, the sub-sample size remains comparable as specimens are generated with replacement rather than the original input number of samples. The Random Forest constructs multiple decision trees from a randomly selected part of the training dataset, forming a diverse ensemble that counteracts individual tree idiosyncrasies. Renowned for its effectiveness in handling complex datasets, addressing overfitting risks, and improving overall predictive accuracy, the random forest classifier stands as a robust and versatile tool in machine learning [5].

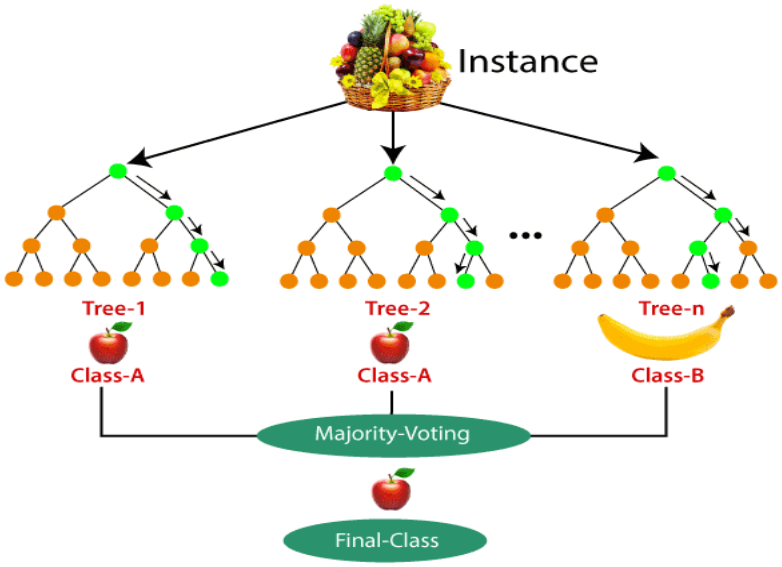


Figure 1.6 Random Forest [7]

### *1.4.5 Support Vector Machine*

In the realm of machine learning methodologies, Support Vector Machines (SVM) stand out as versatile and powerful tools, acting as supervised classifiers applicable to both classification and regression tasks. With a primary focus on addressing classification challenges, SVM navigates a multidimensional space to categorize data points, strategically determining the optimal hyperplane—a decisive boundary that facilitates data classification. The essence of SVM lies in maximizing the spatial distance between categories and the hyperplane, contributing to efficient and accurate classification. Renowned for its adaptability and versatility, SVM leverages the

geometric intricacies of data, providing a robust framework for handling complex datasets. Its impact extends beyond classification tasks, making significant contributions to the broader landscape of machine learning applications, where the quest for precision and efficiency drives the continuous evolution of these methodologies [7].

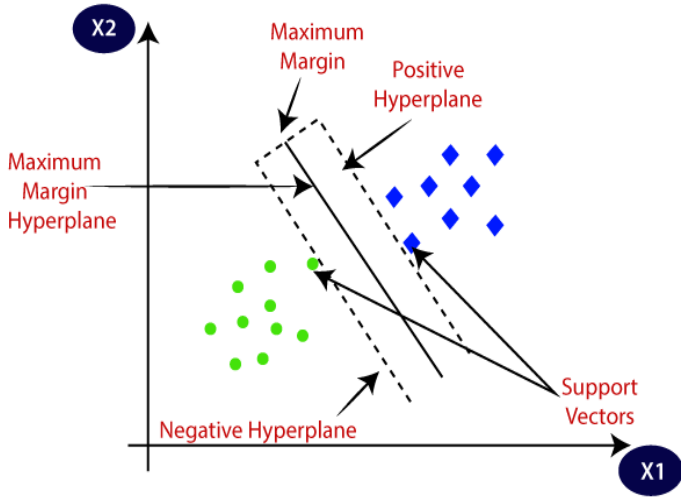


Figure 1.7 SVM [7]

### *1.4.6 Neural Networks*

A neural network uses a variety of methods that mimic human cognition to find correlations in a piece of data. In this regard, neural networks are systems of neurons that can be artificial or edible. Because neural networks are capable of adapting to changing input, the best possible result may be obtained without changing the output criterion. Artificial intelligence, such as neural networks, is becoming increasingly popular in the development of financial products. The training process involves presenting the network with labeled data, adjusting the weights to minimize the difference between predicted and actual outputs. Activation functions, such as the sigmoid or rectified linear unit (ReLU), introduce non-linearities, enabling the network to capture complex patterns. Neural networks are comparable to the neural networks seen in human brains. A "neuron" is a mathematical construct that uses a certain neural network structure to receive and process input. Regression analysis and curve fitting are two descriptive statistics that the network closely resembles. The layers of cluster members comprise a neural network. Reminiscent of a multiple linear regression, each node is a perceptron. A nonlinear activation function is produced from multiple linear regression data by means of the perceptron [8].

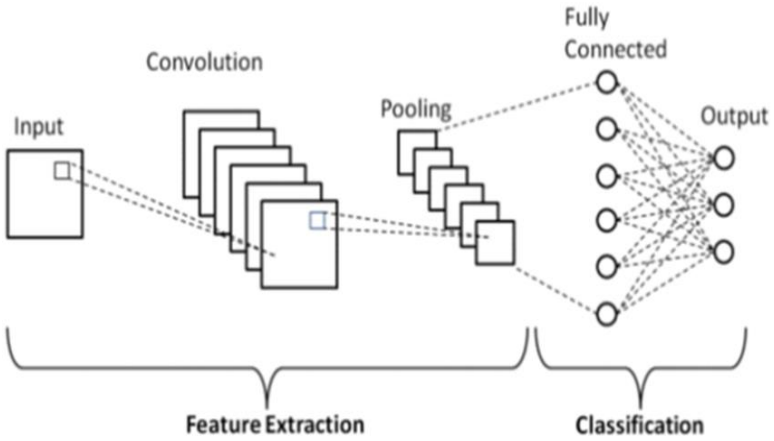


Figure 1.8 Neural Network [8]

### *1.4.7 Recurrent Neural Networks (RNNs)*

Recurrent Neural Networks (RNNs) represent a neural network technology designed for the analysis of sequential inputs, such as historical data or spoken words. Distinguished by feedback pathways, RNNs retain information from previous time steps, enabling the capture of contextual dependencies. Widely employed in speech recognition and linguistic processing, RNNs excel at discerning sequential features within data and utilizing patterns to predict the subsequent likely scenario. Their ability to encapsulate temporal information equips them for tasks requiring an understanding of context and historical dependencies, making Recurrent Neural Networks a pivotal tool in the domain of artificial intelligence and predictive modeling [8].

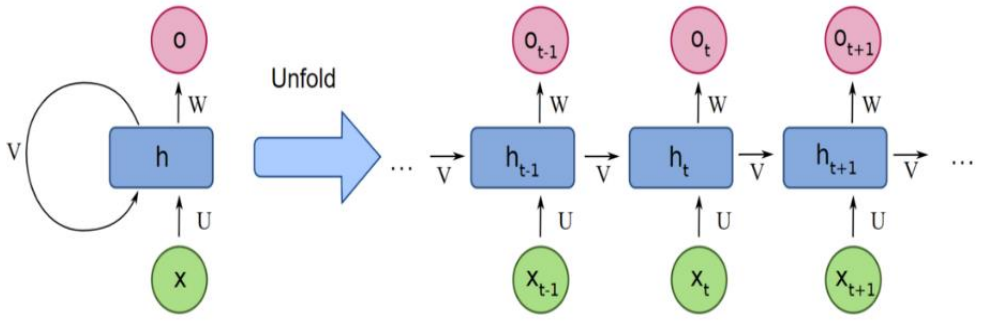


Figure 1.9 RNN [8]

### *1.4.8 Long Short-term Memory (LSTM)*

The LSTM algorithm (Long Short-Term Memory) emerges as a cutting-edge iteration of Recurrent Neural Networks (RNNs), meticulously designed for the intricate analysis of sequential data, encompassing realms like historical data and spoken words. Marked

by distinctive feedback pathways, LSTMs set themselves apart by adeptly retaining and harnessing information from preceding time steps, fostering a nuanced understanding of contextual dependencies. Revered for their versatility, LSTMs find wide-ranging applications in fields such as speech recognition, linguistic processing, emotion analysis, word modeling, speech detection, and visual evaluation. Their exceptional architectural design empowers them to discern intricate sequential features, predict subsequent scenarios, and contribute significantly to artificial intelligence. With an inherent ability to capture enduring relationships across diverse time intervals, LSTMs stand as pivotal contributors to the dynamic landscape of advanced data analysis and predictive modeling, shaping the trajectory of intelligent computing systems for the future [9].

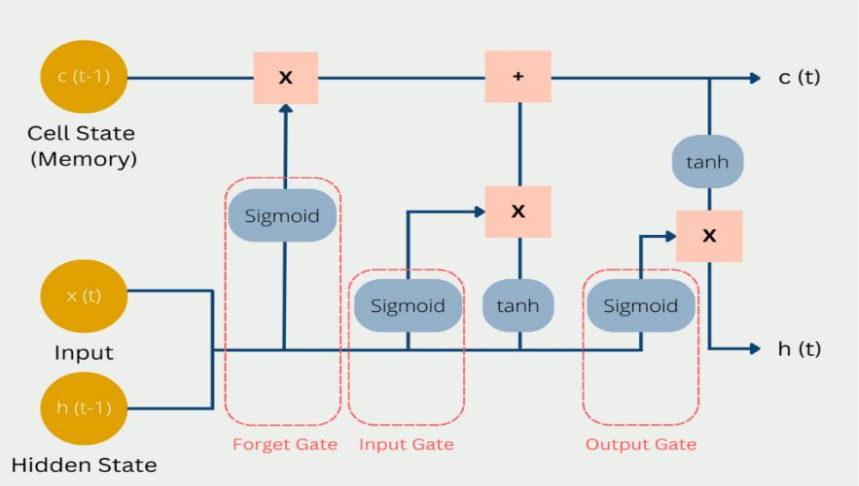


Figure 1.10 LSTM Layers [9]

### *1.4.10 Convolutional Neural Networks (CNNs)*

CNNs are made particularly for processing grid-like data, including spectrograms or pic- tures. To automatically detect and extract local spatial patterns from the input data, they employ convolutional layers as shown in **Fig. 1.13** CNNs have had great success in a num- ber of applications involving computer vision. Convolutional neural networks (CNNs) are an advanced variety of neural networks that were created specifically for analyzing and interpreting hierarchical grid-like data, such as photos or movies. In applications in- volving computer vision, including image classification, object recognition, and picture division, CNNs have been demonstrated to be incredibly effective. Similar to other designs for neural networks, CNNs learn the ideal numbers of the fil- tration weights throughout the training phase.

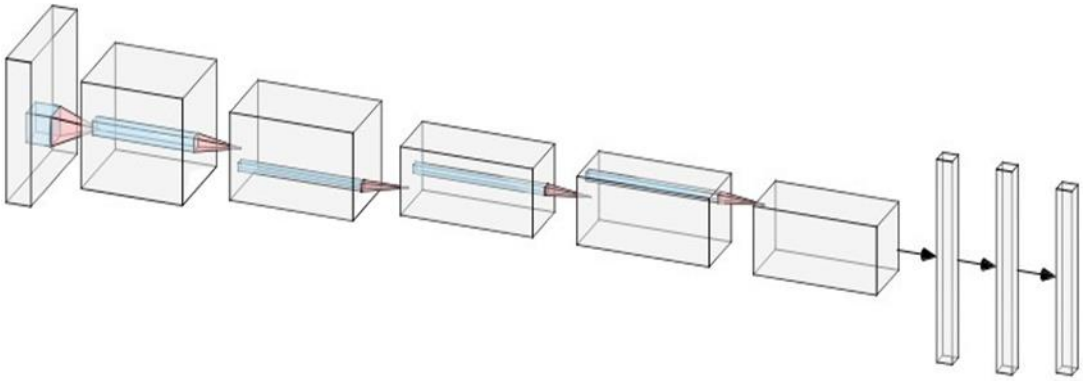

Figure 1.11 Convolutional Neural Networks (CNNs) [6]

Applying a method of optimization like gradient descent, the number of weights is modified to reduce the discrepancy between the output that had been anticipated and the output that was actually produced. When analyzing grid-like input, CNNs have a number of advantages over typical feedforward neural networks. They can automatically learn and extract key properties from the data, which makes them robust to changes in size, translation, and rotation. Furthermore, the neural network could pick up hierarchical representations of the input data by using shared weights in the convolution layers, which capture both local and global features. All things considered, CNNs have advanced the area of artificial intelligence (AI) and have attained cutting-edge performance on a variety of tasks, making them a crucial tool for both video and image processing [6].

## 1.5 Challenges in Detecting Anemia

Anemia exerts substantial implications on cardiovascular health, notably contributing to arrhythmia—an irregular or rapid heartbeat. The diminished oxygen levels in the blood prompt compensatory measures by the heart, leading to increased pumping, which, over time, can result in cardiac strain. This strain may manifest as an enlarged heart and, in severe cases, progress to cardiac failure. Recognizing the intricate interplay between anemia and cardiovascular complications emphasizes the critical necessity for timely detection and intervention. Effectively addressing anemia's impact on the heart becomes pivotal in mitigating the risk of arrhythmias and averting potential cardiac challenges. This understanding underscores the broader significance of considering cardiovascular health within the framework of anemia management, highlighting the need for comprehensive care to safeguard against associated complications and promote overall well-being.

## 1.6 Organization of Thesis

In this thesis, we describe our study in six chapters, where the thesis outlines are described as follows:

- **Chapter 1 (Introduction):** In this chapter, the general background of the work, including status of anemia, machine learning and deep learning of this study are discussed.
- **Chapter 2 (Literature Review):** In this chapter, a comprehensive literature review, problem statement and the anemia disease detection are discussed.
- **Chapter 3 (Methodology):** In this chapter, the approach required to create the prediction model and how the parameters are set is discussed.
- **Chapter 4 (Results and Analysis):** In this chapter, dataset and experimental settings, experimental tools and environment, and comparison are discussed.
- **Chapter 5 (Conclusion):** In this chapter, the whole work is summarized with constructive conclusions, along with the application of this work and the future scope regarding improving the work.

# CHAPTER II
# LITERATURE REVIEW

## 2.1 Introduction

Knowing machine learning was a key component of the field analysis we completed for the thesis. Machine Learning is an AI approach that trains machines to learn through experience. Machine learning techniques employ computer techniques to “learn” knowledge by examining data without depending on a preconceived equation as an example. Machine learning is an instance of AI. It's the technique of utilizing mathematical representations of data to assist a computer understand without direct direction. This lets an electronic system keep learning and growing on its own, depending on experience.

In basic words, AI is software for machines that replicates the ways that people think in order to execute challenging tasks such as evaluating, thinking, thinking learning. Machine learning, however, is a subset of AI that employs algorithms trained upon data to generate models which can handle such complicated tasks. Machines learning is crucial because it offers organizations a picture of trends in consumer behavior or company operating designs, as well as assists the creation of new products. Many of today's major organizations, such as Facebook, Google and Uber, consider artificial intelligence a core aspect of their operations.

The use of machine learning may aid in the diagnosis of illnesses. Many clinicians utilize chatbots with voice recognition skills to discover trends in symptoms. Real-world instances for medical diagnosis: Assisting in making a diagnosis or proposing a therapy option. With the use of machine learning (ML), which is a sort of artificially intelligent (AI), software systems may anticipate consequences more successfully without needing to be actively trained to do so. In order to forecast eventual result values, AI computers utilize prior data as input [10].

## 2.2 Machine Learning and Disease Detection

The use of machine learning in that area of clinical diagnostics is developing progressively. This may be helped mostly to the progress in the categorization and identification methods employed in illness identification and this is capable to give data and assists healthcare specialists in earlier identification of catastrophic diseases and

consequently, raise the life rate of clients dramatically. In this research, we use multiple categorization methods, every with its unique benefit on three distinct datasets of illness (Heart, Breast cancer, Diabetes) willing in UCI repository for illness forecasting. The characteristic selection for each database was achieved by backwards predicting utilizing the probability value test. Findings of the research strengthen the notion of the use of machine learning in initial diagnosis of illnesses [11].

## 2.3 Machine Learning and Anemia Disease Detection

Anemia is a condition that is characterized with the deficiencies of red plasma cells. The form of red plasma cell transforms to sickle or bulb form in blood cell anemia disorder. The traditional examination of microscope pictures is highly tough and laborious task. In this study image analysis and machine learning methods is employed to automating the procedure of identification of shear cells in micro pictures then categorize the the Royal Bank of Canada into three shapes: circular in shape elongate (sickle cells) or other shape. The micro picture is prepared and the Otsu thresholds method is employed for segmentation. Then, A watershed segment is done to separate the overlapping cells. The geometric, analytical and sensory elements are obtained from these photos. The machine learning classifiers at random forest, logistic regression naïve baye sand support vector machine is utilized. This study provides the comparison of these algorithms.

Sickle cell disease is a syndrome during which the RBCs are not created how they can be. In this study Watershed division was utilized to eliminate the overlapping cell [1]. Elements such as mathematical, statistical characteristics and textural are removed from cells that we have utilized four machine learning model to categorize recurrent, rectangular and other shapes corresponding to our data collection. Naïve Bayes has less accuracy, recall f-score and accuracy than SVM, Random forest & Logistic regression. Logistic regression & SVM has the same efficiency 90% and is lower than random forest which has an accuracy of 92%. Observing the accuracy random woodland offer the nice result.

## 2.4 Role of Machine Learning and Deep Learning in Healthcare

In the intricate landscape of healthcare, the collaborative synergy between Machine Learning and Deep Learning reveals an unfolding narrative of transformative potential,

especially within the nuanced realm of Anemia prediction. As this exploration delves into the expansive Role of Machine Learning in Healthcare, it embarks on a journey through the corridors of medical research, drawing wisdom from seminal works that illuminate the multifaceted applications of ML. Within this intricate tapestry, ML emerges as a pivotal force, capable of discerning subtle patterns within vast repositories of patient data. From disease diagnostics to the tailoring of personalized treatment plans, ML assumes the mantle of an indispensable ally, reshaping the contours of contemporary healthcare.

The foundation of this exploration rests on the insights derived from significant works in medical research, serving as guideposts in the quest to understand the potential applications of ML in predicting Anemia. It becomes evident that, beyond its immediate relevance, ML holds the promise of revolutionizing how we perceive and approach healthcare challenges. As the narrative unfolds, each algorithm within the ML repertoire becomes a unique note, contributing to a symphony of advanced healthcare analytics. It is within this orchestration that the Role of Machine Learning takes center stage, a resilient force poised to redefine traditional medical paradigms.

This symphony of data-driven insights signifies more than a mere amalgamation of technology and healthcare; it marks a paradigm shift in our approach to patient care. The Role of Machine Learning stands as a linchpin, navigating through the complexities of medical data and promising a crescendo of understanding that extends far beyond conventional boundaries. In this era of precision medicine, the marriage of ML with healthcare unveils a rich potential for improved diagnostics, personalized interventions, and ultimately, enhanced patient outcomes. As we traverse this evolving landscape, the narrative of ML's role in healthcare echoes with the promise of a brighter, more data-informed future for medical practice.

## 2.5 Application of Deep Learning in Disease Prediction

In this part of our study, we're looking at how Deep Learning (DL), a smart technology, can help predict diseases, especially focusing on Anemia. We want to figure out how DL, which is a kind of artificial intelligence, is being used effectively in healthcare. By checking out important research papers, we can see examples of how DL's smart neural networks are really good at figuring out complicated patterns. These patterns are super important when we're trying to predict diseases, and for us, that means understanding

Anemia better. The research we're looking at doesn't just show success stories but also helps us see how DL's clever algorithms can understand the detailed stuff we need to predict Anemia accurately. predict Anemia, and it's all part of making healthcare smarter. Through this exploration, we gain insights into the practical and transformative potential of DL, offering a clearer picture of how this advanced technology can contribute to enhancing predictive analytics in healthcare. This discussion lays the groundwork for understanding the specific ways in which DL can benefit the prediction of Anemia, contributing to the broader context of healthcare advancements.

## 2.6 Challenges in Implementing Machine Learning Model for Anemia

The implementation of machine learning models for anemia prediction presents several challenges that warrant careful consideration. These challenges encompass both the intricacies of the dataset and the complexities inherent in predictive modeling. Key challenges in our thesis include:

1. **Complexity of Anemia Patterns**: Anemia is a multifaceted health condition influenced by various factors. Capturing the intricate patterns within the dataset, especially those related to gender, hemoglobin levels, and related indices (MCH, MCHC, MCV), poses a challenge in developing models that can discern subtle variations indicative of anemia.
2. **Risk of Overfitting**: The dataset's finite size raises concerns about models memorizing specific patterns rather than learning the underlying relationships. Striking a balance between model complexity and generalization to ensure avoidance of overfitting is a critical challenge.
3. **Generalization to Unseen Data**: Ensuring that the developed models generalize well to unseen data is imperative for their practical utility. The challenge lies in creating models that can adapt to a broader range of scenarios, thereby enhancing their predictive accuracy beyond the training dataset.
4. **Optimal Model Selection**: Choosing the most suitable machine learning model for anemia prediction requires a thorough understanding of the strengths and limitations of various approaches, including Random Forest, SVM, KNN, Logistic Regression, and Sequential models. Selecting the optimal model tailored to the characteristics of the dataset is a non-trivial task.

5. **Data Augmentation for Representativeness**: Introducing variability and diversity through data augmentation techniques is essential for model adaptability. However, striking the right balance and ensuring the augmented data remains representative of real-world scenarios is a challenge in itself.

Our thesis addresses these challenges through meticulous model development, data preparation, and augmentation strategies, aiming to contribute valuable insights to the domain of anemia prediction while acknowledging and overcoming the inherent complexities in the implementation of machine learning models for this health condition.

## 2.7 Problem Statement

Globally, approximately 25% of the population experiences Anemia, with half of all cases attributed to iron deficiency, the most prevalent cause. This health issue poses a substantial threat to school-age children, particularly during their adolescent years when the prevalence of Anemia increases. Iron-deficient students may experience reduced strength, making them more susceptible to infections. This decline in physical well-being, coupled with a lower Q-factor, adversely affects both learning capacities and academic achievement. Thus, addressing iron deficiency-related anemia is crucial for promoting the overall health and educational outcomes of adolescents around the world.

## 2.8 Thesis Objectives

The key objectives of this thesis are as follows:

- To create a model based on different proposed models that helps to detect anemia.
- To achieve maximum accuracy with satisfactory efficiency in other performance parameters of anemia detection from the comparison of different models.

## 2.9 Motivation

This work's main goal is to study the dataset and use the algorithms of random forest, K-Nearest Neighbors, SVM, logistic regression and sequential for the prediction. So, our motivation is simply to understand how the algorithms function and to accomplish their goals by researching statistical models for machine learning.

## 2.10 Literature Review

Machine Learning has been an emerging tool for Prediction of Diseases. Results from the study of related work are shown for numerous learning datasets, where analysis and predictions were spread using a variety of methods and procedures. Many academics use different knowledge mining approaches, machine learning algorithms, or combinations of those techniques to develop and enforce multiple prediction models. These researchers of [12] applied the SMO support vector machine and the C4.5 decision tree approach to analyze the achievement of these two techniques in estimating anemia. WEKA was applied in [13] to develop classifiers ideal for establishing mobile industries that can forecast and analyze plasma knowledge feedback. The work [14] has figured out that each algorithm has its own strength as well as weakness and its own area of implementation.

The authors Uddin, Khan, Hossain, and Moni conducted a study comparing different supervised machine learning algorithms for disease prediction, as detailed in their work published in BMC Medical Informatics and Decision Making [15]. In this research, various algorithms, including Random Forest, Decision Tree, ANN, SVM, Logistic Regression, Naïve Bayes, and K-nearest Neighbor, were assessed. The study found that the Support Vector Machine (SVM) algorithm was frequently applied, and the Random Forest (RF) algorithm exhibited superior accuracy, providing valuable insights into disease prediction methodologies [15]. Additionally, Jaiswal, Srivastava, and Siddiqui investigated machine learning algorithms for predicting anemia in their contribution to the book "Recent Trends in Communication, Computing, and Electronics" published by Springer [16]. Their work specifically delved into the application of Naive Bayes, Random Forest, and Decision Tree algorithms for anemia prediction using Complete Blood Count (CBC). Notably, the Naive-Bayes technique outperformed Decision Tree and Random Forest in terms of accuracy [16].

In the study referenced as [17], conducted by Dalvi and Vernekar, the authors determined the optimal individual classifier or subset of classifiers, in combination with each other, to achieve maximum accuracy in red blood cell classification for the detection of anemia. This work showcased a unique idea by incorporating a subset of classifiers and utilizing ensemble learning techniques, demonstrating an innovative approach to enhance accuracy in anemia detection.

On the other hand, reference [18], conducted by Khan et al., the research focused on predicting the anemia status of children under five years by considering common risk factors as features. The findings of this investigation concluded that machine learning methods, in addition to classical regression techniques, can be effectively employed to predict anemia in this demographic. Furthermore, the authors [19], Anand, Gupta, and Sharma, constructed predictive models utilizing identified risk factors through a machine learning approach to predict the anemia status of children under 36 months. This research contributes valuable insights into the application of machine learning algorithms for predicting childhood anemia, emphasizing the significance of incorporating modern techniques for enhanced predictive modeling.

In the research referenced as [20], conducted by Zhang and Tang, a prediction model was developed to assess the potential risk of anemia among infants. The Multilayer Perceptron model (MLP) employed in the study identified three significant risk factors for anemia, namely exclusive breastfeeding, maternal anemia during pregnancy, and non-timely supplementation of complementary food. The findings of this work contribute to understanding and predicting iron deficiency anemia among infants, emphasizing the importance of considering specific risk factors. Additionally, the authors [21], Ewusie, Ahiadeke, Beyene, and Hamid, investigated the prevalence of anemia in children under five years, focusing on the Ghanaian population. Their study revealed a higher prevalence of anemia below 2 years of age, providing valuable insights into the age-specific dynamics of anemia in this population. Moreover, the research by Wang et al. [22] explored the prevalence of anemia as children transitioned from infancy to preschool-age in rural China. The findings highlighted that children at greater risk for developing anemia tend to have persistent anemia between toddlerhood and preschool-age, emphasizing the importance of tracking the dynamic anemia status of children over time.

The researchers employed the J48 and a naïve Bayes classifier to evaluate neural network recognition strategies. The J48 predictor has the best efficiency, corresponding to the information. Applying a decision tree procedure, Dogan & Turkoglu [23] produced a decision support method to diagnose iron insufficiency anemia. Three hematopoietic characteristics are employed in this technique: blood zinc, plasma irons-binding capacity, and ferritin. The findings were well linked to the doctor's choice, and the evaluation was conducted upon information from 96 patients. Abdullah and

Alasmari [24] employed the WEKA, a method to predict a subtype of anemia from the CBC reports (Naive Bayes, Multilayer Perceiving, J48, and SMO). The research was predicated on real-world data from 41 anemia patients' CBC readings. The J48 selection tree algorithm with SMO fared best overall 93.75 cent accuracy, comparable.

Dithy and Krishnapriya [25] diagnosed anemia amongst pregnant women employing ANN and gausnominal classification algorithm with a reliability of 0.65% and 0.74%, respectively. Dithy and Krishnapriya sought to classify anemia in young women employing random prediction (Rp) classification method and attained an accuracy of 0.65%, 0.76%, 0.826%, and 0.92% using ANN, gausnominal, vector neighbor, and random, correspondingly. Nevertheless, these studies did not evaluate all possible qualities that are presented in section I, which helps to adopt holistic actions. Additionally, sought to construct a prediction approach, but they did to uncover risk variables, and extracted rules that are crucial to build based on evidence tactics, strategies, and programs to prevention et/or decreasing hemophilia among pregnant women in Ethiopia.

Early identification of anemia aids in the prevention of various connected disorders that may impede the growth and development in the future. This issue emerges as a social reason for conducting this research. Furthermore, being the primary test for effective anemia diagnosis, anemia is often diagnosed on a full blood count. Henceforth, the main aim of this research is to outperform the existing research, to develop models for better results and build better models than existing models for our dataset. So that we may determine which model provides the most accurate accuracy on anemia prediction. We modified models for different machine learning models and built a deep learning model and then compare the performances of those models based on evaluation criteria for prediction of Anemia.

# CHAPTER III
# METHODOLOGY

## 3.1 Introduction

The research methodology will be covered in this chapter. In this work, anemia is predicted with the application of ML and Deep Learning. There will be a discussion regarding the data collection and explanation procedure. The proposed model will then be discussed.

## 3.2 Dataset

Dataset provided a diverse and comprehensive representation of potential predictors for anemia. The dataset has the attributes: gender (0-male and 1-female), hemoglobin, mch, mcv mchc and result. The dataset's size is (8545 rows by 6 columns) ensured an ample amount of information for training and evaluating our predictive models. The inclusion of gender alongside hematological parameters aimed to capture a entire view of factors influencing anemia, contributing to the effectiveness of our subsequent analysis.

On the other hand, another dataset also used in this study comprises 61 rows and includes six columns, namely Gender, Hemoglobin, MCH (Mean Corpuscular Hemoglobin), MCHC (Mean Corpuscular Hemoglobin Concentration), MCV (Mean Corpuscular Volume), and Result. This dataset was specifically collected from Parkview Hospital Limited, Chittagong and is intended for testing purposes. Each row represents an individual entry, providing information on the gender of the subject and various hematological parameters such as Hemoglobin, MCH, MCHC, and MCV. The 'Result' column serves as a testing variable, likely indicating the outcome or classification based on the collected data. Utilizing datasets from reputable healthcare institutions like Parkview Hospital enhances the reliability and relevance of the study's findings in the context of the specific parameters under investigation.

## 3.3 Data Collection

The dataset was collected from Kaggle, providing a diverse and comprehensive representation of potential predictors for anemia. The selection of these attributes was guided by their relevance to hematological health and their potential contribution to accurate predictions. In summary, my data collection process involved obtaining a

carefully curated dataset from Kaggle, setting the stage for my exploration into anemia prediction using machine learning models.

For testing purposes, an additional dataset was meticulously collected from our primary research site, Parkview Hospital Limited, Chittagong. The dataset, comprising 61 records across genders and ages, was extracted from patient reports. In the RBC count section, six attributes—Gender, Hemoglobin, Mean Corpuscular Hemoglobin (MCH), Mean Corpuscular Hemoglobin Concentration (MCHC), Mean Corpuscular Volume (MCV), and Result—were thoughtfully chosen for predicting Anemia through testing. This dataset underwent thorough preprocessing, involving manual extraction from Hematological Analyzer records, followed by various preprocessing techniques and data normalization, ensuring its suitability for my research.

## 3.4 Data Preparation

The data preparation methodology was strong and fit to ensure the reliability and effectiveness of my predictive models. Initially, I addressed data integrity issues by carefully handling missing values and managing outliers within the dataset. Feature selection played a crucial role, with emphasis on gender, hemoglobin, MCH, MCHC, and MCV, ensuring that our models were trained on the most informative attributes. To facilitate model compatibility, numerical features underwent normalization and scaling, particularly for SVM and KNN models.

The dataset was judiciously split into training and testing sets, allowing for a comprehensive assessment of model performance. Categorical variables, such as gender, were encoded using suitable techniques like one-hot encoding. In mitigating potential class imbalances, oversampling, undersampling, and synthetic data generation strategies were employed. For the Sequential model, we incorporated data augmentation techniques to enhance the diversity of the training samples. A final thorough check ensured the prepared dataset's readiness for input into Random Forest, SVM, KNN, Logistic Regression, and Sequential models.

I applied the same data preparation techniques to this primary dataset as I did for the dataset obtained from Kaggle. In fact, data preparation is a crucial phase in my research, representing the careful refinement and organization of the collected dataset. Initially recorded in the Hematological Analyzer, I manually extracted the raw data with precision to rectify any errors or inconsistencies. Following this, I applied various

preprocessing techniques adeptly, including tasks such as handling missing values, addressing outliers, and transforming variables. This thorough process ensures the dataset's cleanliness, normalization, and optimal organization, elevating its quality for subsequent analysis. The meticulous execution of these techniques plays a vital role in reinforcing the reliability and precision of our research outcomes, ensuring that the dataset is well-prepared for the application of machine learning and deep learning models in predicting Anemia.

## 3.5 Data Processing

First, I have mounted out drive. A file system needs to be available that any storage medium may be used by the device itself, either by anyone or by its operating system. The term for this procedure is mount. Here I used the storage of our google drive, that's why I mounted it. I upload my data in google drive. I used some library such as numpy, seaborn, panda ,matplotlib. Then I found that if there is any null data and I didn't find any null value. Last column is "result", we take is as a feature and my target variable. Feature selection focused on key hematological parameters essential for anemia prediction. We setup the code for data analysis and machine learning in Python. It configures the styling of Matplotlib, suppresses warnings, and sets up inline plotting. The repeated %matplotlib inline and warnings.filterwarnings('ignore') suggest a Jupyter Notebook environment where inline plotting is enabled, and warnings are being ignored. The sns.set() statement configures Seaborn's default settings.

After reading Dataset, the code generates a horizontal bar plot that illustrates the distribution of gender categories in the specified DataFrame (df). The bars represent the counts of males (0) and females (1), and each bar is labeled with its corresponding count. Adjusting the figure size with plt.figure allows for better visualization and presentation of the plot. Then, we tested 20% and trained 80% of the whole data. For input, execute the source code for five distinct models: KNN, SVM, Logistic Regression, Random Forest and Sequential. The processing phase involved implementing each model on the prepared dataset, with normalization and scaling applied to numerical features, and categorical variables encoded for compatibility. And then hyperparameters such as learning rate, number of epochs, batch size, activation function, regularization parameters, k value, min samples are used for getting better performance from models. The code uses scikit-learn to implement a Support Vector

Machine (SVM) classifier. It scales features using StandardScaler, trains the SVM with an linear kernel, and evaluates accuracy on a test set, printing the result. Logistic regression employs scikit-learn to implement the model with regularization. It trains the model on training data, predicts on a test set, evaluates accuracy, and prints the result in percentage form. KNN utilizes scikit-learn to implement a classifier with a default value of k=3 neighbors. Random Forest employs scikit-learn to implement a classifier with 100 decision trees. It trains the model on the training data, predicts on the test set, calculates accuracy using metrics module, and prints the result in percentage form. Sequential model implements a binary classification neural network using TensorFlow's Keras API. It defines a sequential model with input, hidden, and output layers. The model is compiled with the SGD optimizer, trained on training data, evaluated on the test set, and its performance is visualized through loss and accuracy plots. The output was assessed in terms of accuracy metrics, revealing varying performance across the models. The results obtained from this data processing pipeline contribute valuable insights for storage and further analysis, enriching the depth of our findings in the quest for accurate anemia disease prediction within a data-driven framework.

On the other hand, the dataset, sourced from Parkview Hospital Limited, Chittagong, is loaded and preprocessed. Categorical variables ('Gender' and 'Result') are encoded into numerical values using LabelEncoder. Features and the target variable are then defined. Feature values are standardized using 'StandardScaler' to ensure uniform scaling, a crucial step for various machine learning algorithms. K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest, and Logistic Regression models are initialized using scikit-learn. The classical machine learning models are trained on the entire dataset using the fit method. A Sequential neural network model is introduced using the Keras library. It consists of one hidden layer with 64 neurons and a ReLU activation function. The output layer employs a sigmoid activation function. The Sequential model is compiled using the SGD optimizer and binary crossentropy loss. It is then trained on the entire dataset for 40 epochs with a batch size of 32. Predictions are made using each of the models, including the classical machine learning models (KNN, SVM, Random Forest, and Logistic Regression) and the Sequential neural network. Testing accuracy is calculated for each model by comparing the predicted outcomes with the actual target values using the 'accuracy_score' metric

from scikit-learn. The results are printed, showcasing the accuracy of each model on the entire dataset. The code is designed to evaluate the testing accuracy of the models on the entire dataset without splitting it into separate training and testing sets. So, this processing allows for the evaluation of both classical machine learning models and a simple neural network, providing a diverse perspective on the dataset's predictive performance.

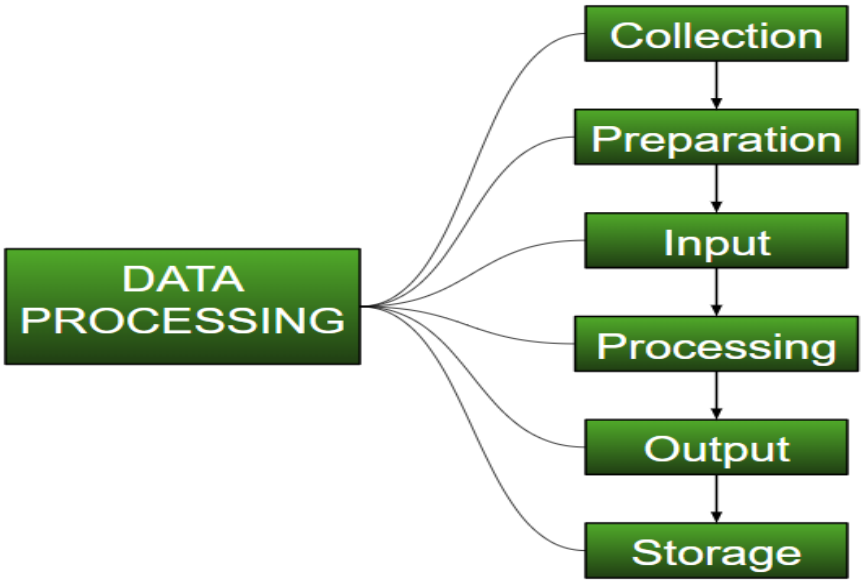


Figure 3.1 Data Processing [26]

### *3.5.1 Selection of Model*

The process of selecting models for my research involved careful consideration of various factors to ensure optimal predictive performance. Five distinct models, namely Support Vector Machine (SVM), k-Nearest Neighbors (KNN), Random Forest, Logistic Regression, and Sequential (referring to a type of neural network architecture), were chosen for their diverse strengths and applications in predictive modeling. SVM, known for its effectiveness in classification tasks, was selected to discern patterns in anemia prediction. KNN, with its reliance on proximity-based classification, offers a different perspective. Random Forest, a robust ensemble learning method, contributes to the diversity of our model set. Logistic Regression provides interpretability, offering insights into the relationships between predictor variables and anemia. Finally, the Sequential model, a neural network architecture, allows for intricate pattem recognition. This diverse ensemble aims to capture the complexity of anemia prediction, ensuring a comprehensive evaluation of model performance. By

strategically choosing these models, I aimed to assess their comparative effectiveness, leveraging their unique characteristics to enhance the robustness and reliability of predictive capabilities, aligning with the essence of existing research in healthcare predictive modeling and Anemia studies.

### *3.5.2 Evaluation of Performance*

Here, I have shown confusion marix and ROC curves for each model. The Receiver Operating Characteristic (ROC) curve is a graphical representation of a classifier's performance, illustrating the trade-off between sensitivity and specificity across different thresholds. The Area Under the Curve (AUC) summarizes this trade-off, with a higher AUC indicating a better overall classifier performance. A confusion matrix is a table used in machine learning to evaluate the performance of a classification algorithm. It summarizes the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions, providing insights into the model's accuracy and error types. Negative class (Actual Negative) refers to instances in a binary classification problem where the true class of the observation is negative. In a confusion matrix, these instances are represented in the "negative" row, indicating the number of correct negative predictions made by a classification model. Positive class (Actual Positive) refers to instances in a binary classification problem where the true class of the observation is positive. In a confusion matrix, these instances are represented in the "positive" row, indicating the number of correct positive predictions made by a classification model.

Mathematical expressions and diagram for cofusion matrix;

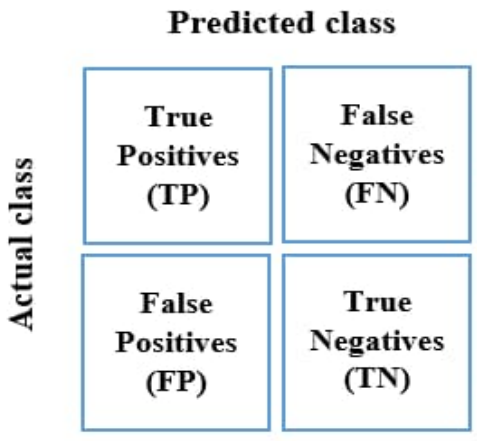


Figure 3.2 Confusion Matrix Diagram

TABLE 3.1 COFUSION MATRIX EQUATIONS

| Measure | Formula |
|---|---|
| Accuracy | $\frac{\mathrm{TP} + \mathrm{TN}}{TP + TN + FP + FN}$ |
| True positive rate | $\frac{\mathrm{TP}}{TP + FN}$ |
| False positive rate | $\frac{FP}{TN + FP}$ |
| True negatie rate | $\frac{\mathrm{TN}}{TN + FP}$ |
| False negative rate | $\frac{FN}{TP + FN}$ |

## 3.6 Data Augmentation

In my study, focused on anemia disease prediction, the quality and diversity of my dataset play a pivotal role in the robustness of my predictive models. I recognized the importance of augmenting my data to ensure comprehensive model training. To address this, I implemented a data augmentation strategy aimed at enriching my dataset. Through techniques such as shuffling instance order and introducing controlled perturbations, especially in numerical features, my goal was to create a more representative training set. This augmentation process not only expanded the quantity of training data but also introduced nuanced variations crucial for the adaptability of models, including Random Forest, SVM, KNN, Logistic Regression, and Sequential. Sequential model implements a binary classification neural network using TensorFlow's Keras API. It defines a sequential model with input, hidden, and output layers. The model is compiled with the Adam optimizer, trained on training data, evaluated on the test set, and its performance is visualized through loss and accuracy plots. This reaffirms the effectiveness of my approach in fortifying the predictive capabilities of our models and emphasizes the essential role of a diverse training dataset in the realm of anemia disease prediction.

To enhance the diversity and robustness of the dataset collected from Parkview Hospital Chittagong, data augmentation techniques were employed, similar to those applied to the dataset obtained from Kaggle, as explained in the preceding paragraph. This

involved applying transformations to the existing data, such as rotation, flipping, and scaling, to generate new synthetic samples. By augmenting the dataset, the machine learning and deep learning models are exposed to a broader range of variations, potentially improving their performance in handling diverse real-world scenarios. The augmentation code provided in the research facilitates the replication of this preprocessing step, contributing to the replicability and transparency of the study.

## 3.7 Data Analysis

In data analysis for anemia disease prediction, I meticulously trained and tested my models using a 8545-row by 6-column dataset on Google Colaboratory. The data was split into 80% for training and 20% for testing, ensuring a strong evaluation of model performance. Across the five models — KNN, SVM, Logistic Regression, Random Forest and Sequential distinct accuracy levels were achieved. The processing phase involved implementing each model on the prepared dataset, with normalization and scaling applied to numerical features, and categorical variables encoded for compatibility. And then hyperparameters such as learning rate, number of epochs, batch size, activation function, regularization parameters, k value, min samples are used for getting better performance from models. The code uses scikit-learn to implement a Support Vector Machine (SVM) classifier. It scales features using Standard Scaler, trains the SVM with a linear kernel, and evaluates accuracy on a test set, printing the result. Logistic regression employs scikit-learn to implement the model with regularization. It trains the model on training data, predicts on a test set, evaluates accuracy, and prints the result in percentage form. KNN utilizes scikit-learn to implement a classifier with a default value of k=3 neighbors. Random Forest employs scikit-learn to implement a classifier with 100 decision trees. It trains the model on the training data, predicts on the test set, calculates accuracy using metrics module, and prints the result in percentage form. Sequential model implements a binary classification neural network using TensorFlow's Keras API. It defines a sequential model with input, hidden, and output layers. The model is compiled with the SGD optimizer, trained on training data, evaluated on the test set, and its performance is visualized through loss and accuracy plots. And then we have shown confusion matrix and ROC curves for all models based on accuracy. Remarkably, the Random Forest model emerged as the standout performer, showcasing a perfect accuracy of 100%. This exceptional result underscores the model's effectiveness in accurately predicting

anemia within my dataset. SVM closely followed with a commendable accuracy of 97.83%. In summary, my data analysis unequivocally identifies Random Forest as the model with the highest accuracy, making it the most robust choice for anemia disease prediction within the scope of our thesis.

On the other hand, the primary dataset from Parkview Hospital Chittagong was subjected to testing using various machine learning models. The dataset was first loaded into a Data Frame, and column names were cleaned by removing any trailing spaces. The features (X) and labels (Y) were then extracted for the test set. The K-Nearest Neighbors (KNN) model achieved an accuracy of 88.33% on the primary dataset. Subsequently, the Support Vector Machine (SVM) classifier demonstrated an accuracy of 97.83%. The Regularized Logistic Regression model exhibited an accuracy of 93.33%, and the Random Forest model achieved 91.67% accuracy. Finally, the Sequential model demonstrated exceptional accuracy, reaching 99.82% on the primary dataset. These results provide insights into the performance of different models in predicting Anemia based on the dataset collected from the hospital.

## 3.8 Functioal Diagram of Study

I have collected datasets from Kaggle, a popular platform for datasets then uploaded it to the drive. After that I mounted the drive to check if there are any missing vales, errors, or dropouts. I engaged in through preprocessing, including normalization, to ensure its quality. The preprocessed data is then divided into two sets: training and testing. The models used for both training and testing sets i.e., machine learning models: Radom Forest, Support Vector Machine, K-Nearest Neighbors, and Logistic Regression whereas Sequential for deep learning. I have used better parameters for each models with appropriate models for the stable of models, reduced overfitting and getting better results than existing reasearches. After that, I have extracted features for the models using the commands. And then I have done processing and trained the main model which classifies my models and gives me the best accuracy. After doing all the necessary work, I have run the codes and achieved an impressive overall 100% predicted accuracy for Radom Forest compared to other models which I have discussed in detail another chapter (Results and Analysis). Based on the results, I have shown confusion matrix and RoC curves for each model, also discussed in the same chapter mentioned above. The [**Figure 3.3**] represents our workflow.

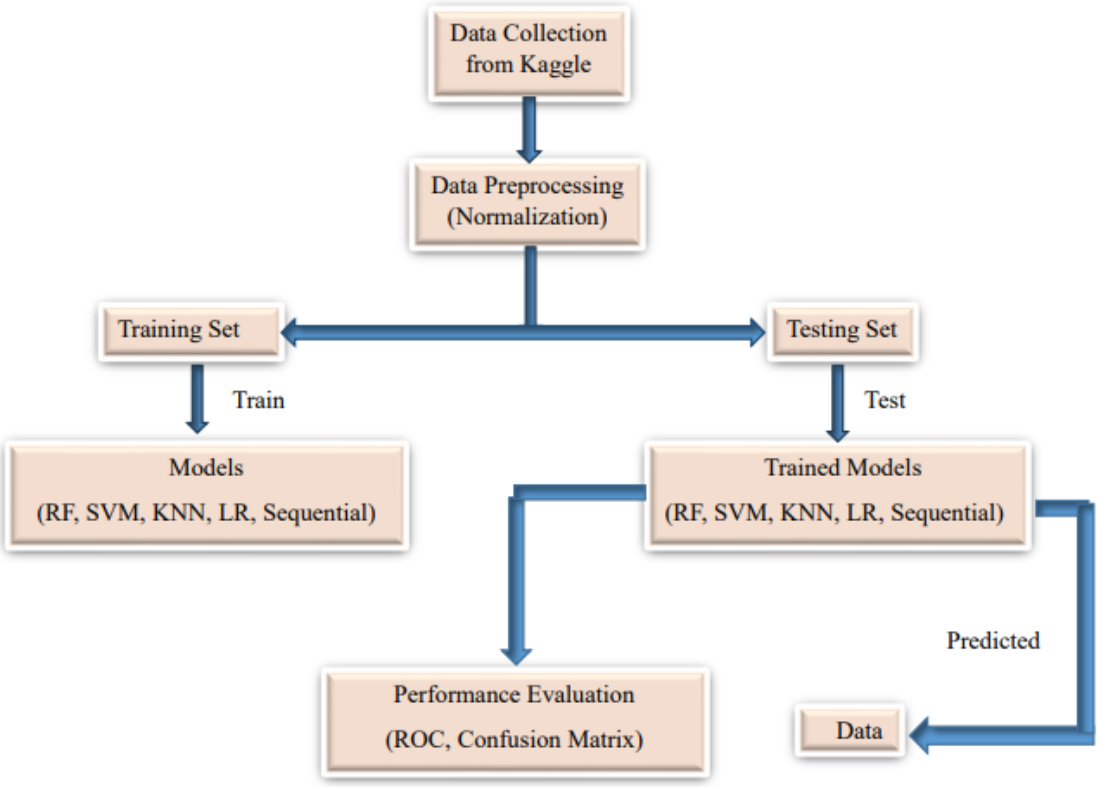


Figure 3.3 Workflow

The functional diagram as [**Figure 3.4**] illustrates a systematic workflow encapsulating the key stages of my study on predicting Anemia using machine learning models with a focus on the raw data is obtained from the Hematological Analyzer records at Parkview Hospital Chittagong. The dataset comprises 61 records across genders and ages, focusing on attributes such as Gender, Hemoglobin (Hb), Mean Corpuscular Hemoglobin (MCH), Mean Corpuscular Hemoglobin Concentration (MCHC), Mean Corpuscular Volume (MCV), and Result. The collected raw data undergoes thorough preprocessing, involving meticulous manual extraction to rectify errors and inconsistencies. Various preprocessing techniques, including handling missing values, addressing outliers, and transforming variables, are skillfully applied to ensure dataset cleanliness, normalization, and optimal organization. To enhance dataset diversity and robustness, data augmentation techniques are applied, ensuring the model's ability to generalize well to new data. The trained models are tested on a new dataset to evaluate their predictive accuracy and performance. The accuracy of each model is assessed using appropriate evaluation metrics, providing insights into their effectiveness in predicting Anemia. The results of the model testing phase are analyzed to draw meaningful conclusions about the predictive capabilities of each model. Based on the

study outcomes, conclusions are drawn regarding the suitability of different machine learning models for Anemia prediction. The functional diagram provides a structured representation of the study's methodology, emphasizing transparency and clarity in the prediction of Anemia using machine learning models with a primary focus on the dataset.

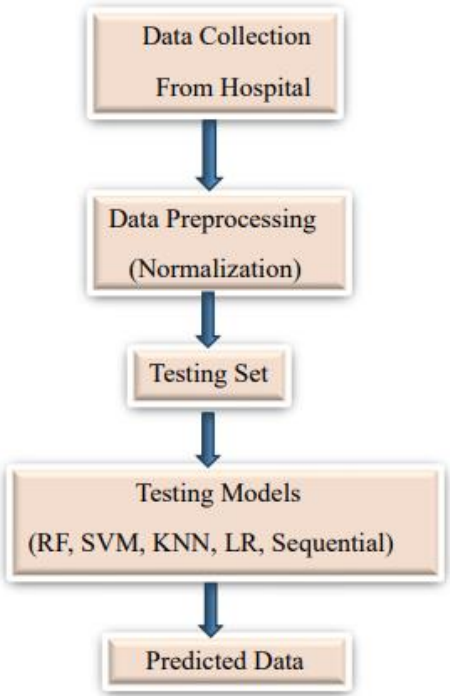


Figure 3.4 Workflow on Testing

## 3.9 Detailed Parameters in Table Form of Different Models

The [**TABLE 3.2**] provides a concise overview of the key parameters for different models used in my thesis methodology. Each row corresponds to a specific model, and the columns outline the relevant parameters tuned during the training process. For example, in the Random Forest model, parameters such as the number of trees, maximum depth, and minimum samples for splitting and leaf nodes are considered. Similarly, for SVM, the regularization parameter (C), kernel choice, and random state are highlighted. The table aids in understanding the unique configuration of each model, facilitating transparency in my methodology and allowing for easy replication of experiments. On the other hand, the same process have done for this too as machine learning and deep learning models, meticulously tested on the primary dataset collected from Parkview Hospital Chittagong. Parameters ranging from hyperparameters to activation functions are outlined, showcasing the intricacies of each model's

architecture. Researchers and practitioners can quickly glean insights into the model-specific configurations, facilitating a deeper comprehension of their roles in the study.

TABLE 3.2 PARAMETERS IN TABLE FORM OF DIFFERENT MODELS

| Model | Parameters |
|---|---|
| KNN | Number of Neighbors |
| SVM | Kernel, Regularization Parameter (C), Random State |
| Logistic Regression | C (Inverse of Regularization Strength), random state |
| Random Forest | random state, n_ estimators |
| Sequential | Relu Activation Function, SGD Optimizer, Binary Cross Entropy, epochs=40, batch_size=64 |

**Binary Step Function:** It is an activation function with a binary threshold orientation. In this case, the threshold value determines when neurons activate. Outputs with multiple values are not allowed [28].

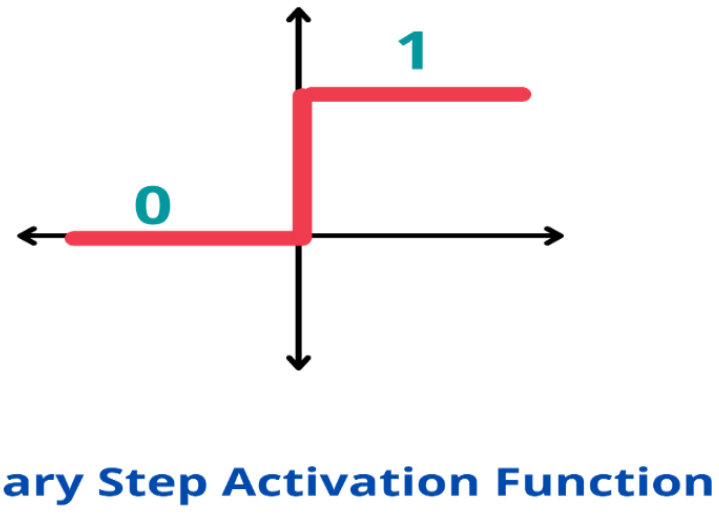


Figure 3.4 Binary Step Function [27]

**Linear Activation Function:** The linear stimulation operation, additionally referred to is "no stimulation," or "personality functional" (divided x1.0), indicates that the activation is proportionate of the inputs. The function that it calls fails to add nothing

with the weighted total of the inputs, it just throws out whatever number it was provided [28].

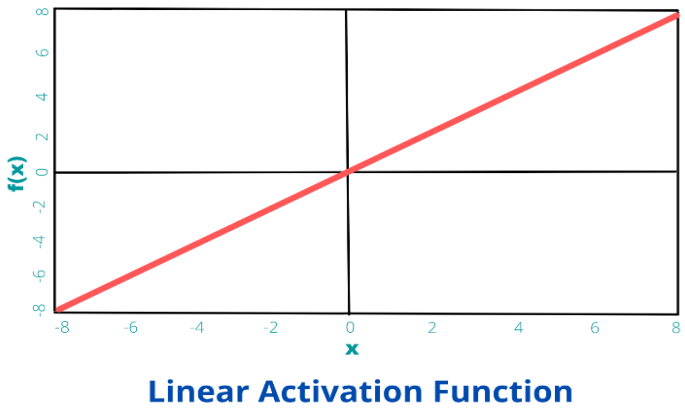


Figure 3.5 Linear Activation Function [27]

**Non-Linear Activation functions:** Non-Linear Activation Functions. The non-linear properties are recognized as being the most commonly used functions for activation. It makes it easier for a model based on neural networks to adjust with a range of input and to discern between the outputs [28].

**Sigmoid or Logistic:** This non-linear function was graphed as a -Sl shape. Its value range is 0 to 1, and it is widely utilized in the binary classification output layer. Its formula is as follows:

A=1/(1+e-x) ……...……...……. (2.1)

Its outputs are not zero centered and its computationally expensive.

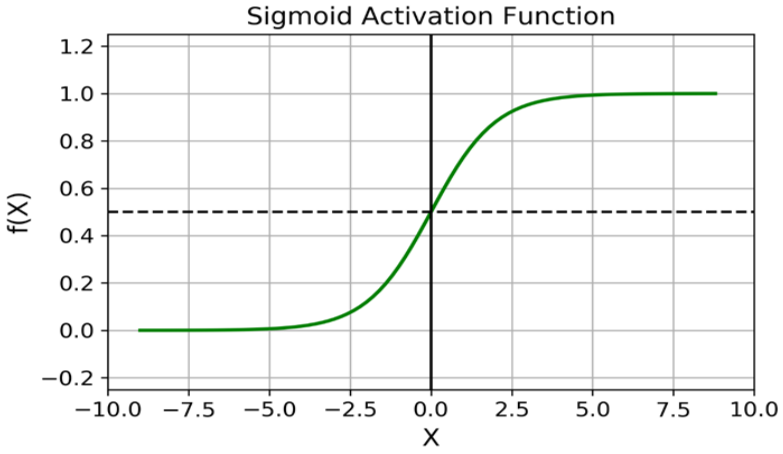


Figure 3.6 Sigmoid Activation Function [26]

**TanH:** The symbol for hyperbolic tangent is TanH. It has also —S‖ form as well.

It is a more effective version of the sigmoid function that has been mathematically shifted. One can derive both functions from the other. Its equation is:

tanh(x)=2 * sigmoid (2x) -1……...……..…… (2.2)

**ReLu:** Rectified Linear Unit is abbreviated as ReLu. It is the most commonly used in the network's hidden layers. It is less computationally expensive and more efficient. It has fewer mathematical operations and learns significantly faster than other activation functions. Its equation is:

A(x) = max (0, x) ……...…… (2.3)

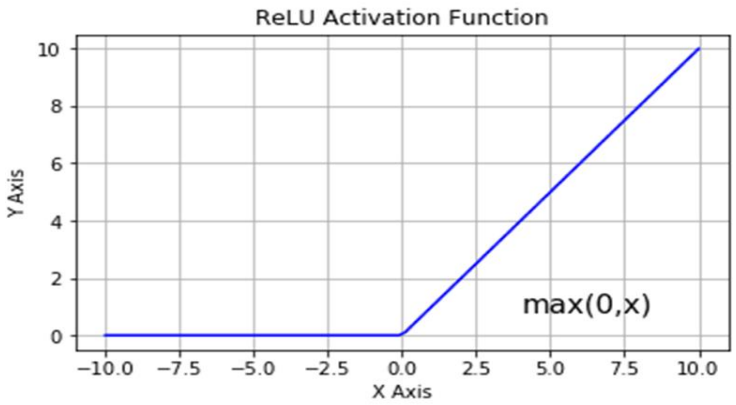


Figure 3.7 ReLu Activation Function [26]

### *3.9.1 Dense Layer*

A sequential model is a sort of RNN design in which the layers are layered one after the other. The model is composed of multiple layers, each of which feeds information to the layer above it and gets it as input. It is the most basic and often used model architecture in deep learning. The dense layer used in this model is shown in **Figure. 3.7.**

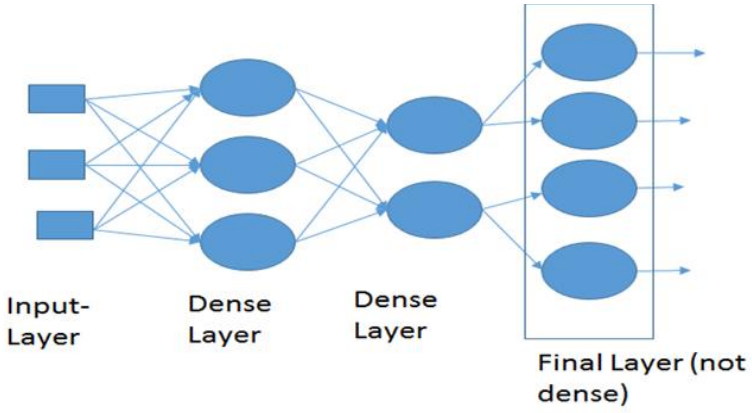


Figure 3.8 Dense Layer [28]

## 3.10 Experimental Settings

I obtained my fundamental facts using the Wikipedia database. I used the initial samples to learn and evaluate my system. My information are split into two groups. These are either training and testing. This study aims at the enhancement of innovative approaches.

## 3.11 Experimental Tools and Environment

The data preprocessing, training, testing, and performance evaluation of the models in this system were carried out using various tools, packages, and settings. These environments and tools will be covered in detail in this section. I have utilized libraries that are part of Python coding language which are TensorFlow and Keras. Python is the most common language used for machine learning since it offers the biggest selection of tools for implementing neural networks and dealing with data.

Keras serves as an interface for the TensorFlow library. It's a Python interface provided by an open-source library. Tensorflow, on the other hand, is a complete set of libraries and a flexible ecosystem of tools that allows developers to quickly design and deploy machine learning applications. It's mostly used for data mining, but it's also utilized for data analysis and machine learning. Colaboratory is a hosted Jupyter notebook that doesn't need to be installed and gives you free access to computer resources like GPU and TPU [18]. Colab allows one to use and share Jupyter Notebooks with others without any kind of run and installation problems. It uses Google Drive for saving files, so one can load files from Google Drive and also from Git Hub. OpenCV: OpenCV-Python is a NumPy-based Python bindings library for handling computer vision issues.


Figure 3.9 TensorFlow [29]

The TensorFlow library is interfaced with via Keras. It's an open-source package that provides an interface in Python. In contrast, TensorFlow provides a comprehensive collection of libraries and an adaptable tool ecosystem that enables programmers to create and implement machine learning applications rapidly [6].


Figure 3.10 Keras [29]

**OpenCV:** OpenCV is a famous artificial intelligence toolkit to construct apps developed in C++ and C. It supports numerous applications including object recognition and Vision Analysis. The field of computer vision intersects with topics like image analysis, Photography, and Structure Identification. A common layer Emgu CV is a tool to execute OpenCV utilizing C# [6].

**NumPy:** NumPy is a framework for the programming language Python that provides providing assistance with massive, complex arrays and matrix structures, coupled with a vast number of basic mathematical operations to work on these numbers [6].


Figure 3.11 NumPy [29]

## 3.12 Implementation Details

Using Tensorflow as and Keras software python package I assessed the model that was suggested on the photographs of my collections. All of the decreased algorithms and boosters included the implementation techniques I utilized.

- I utilized both the SGD Optimized and Category the Cross Entropy representations of the Tensorflow as package to execute the algorithm on the dataset.
- All the resultant values for categories from the Sci Kit-Lear package are one fast transcribed.
- Leavers layers are introduced after the two layers of convolution and fully interconnected layers.

## 3.13 Models Implemetion in Python

First, I have mounted out drive. A file system needs to be available that any storage medium may be used by the device itself, either by anyone or by its operating system. The term for this procedure is mount. Here, I used the storage of our google drive, that's why we mounted it. I upload my data in google drive. Up next, I declare the dataset path, I then read the entire dataset. I used some library such as numpy, seaborn, panda ,matplotlib. Then I found that if there is any null data and I didn't find any null value. Last column is "result", I take is as a feature and our target variable. Feature selection focused on key hematological parameters essential for anemia prediction.

I setup the code for data analysis and machine learning in Python. It configures the styling of Matplotlib, suppresses warnings, and sets up inline plotting. The repeated %matplotlib inline and warnings.filterwarnings('ignore') suggest a Jupyter Notebook environment where inline plotting is enabled, and warnings are being ignored. The sns.set() statement configures Seaborn's default settings. After reading Dataset, the code generates a horizontal bar plot that illustrates the distribution of gender categories in the specified DataFrame (df). The bars represent the counts of males (0) and females (1), and each bar is labeled with its corresponding count. Adjusting the figure size with plt.figure allows for better visualization and presentation of the plot. Then, I tested 20% and trained 80% of the whole data. Then I build our different model and compile the model. After fitting the model, I got accuracy. My implement models are described below,

1) **Random Forest**: At first, implement a Random Forest Classifier model, fit the model to the training data, make predictions on the test data, then calculate the accuracy of the Random Forest model, and print the accuracy as a percentage. Such as a "n_estimator" is used which specifies number of trees, meaning that it will consist of n number of trees. Whereas I used n=100. If n of trees is increased, it can lead robust and stable model, but it will increase computation time. Also, a 'random_state=42" is used which is an arbitrary choice so I can use any integer as the seed.

2) **Support Vector Machine**: It begins by importing essential libraries, including the SVM classifier (SVC), a feature scaler (StandardScaler), and the accuracy score metric. The features are then standardized using StandardScaler, creating scaled versions of the training and test data. An

SVM classifier is instantiated with a linear kernel, specific regularization parameters (C=0.0021), and a fixed random state. The classifier is trained on the scaled training data, and predictions are made on the scaled test set. The accuracy of the SVM classifier is evaluated using the accuracy score metric, and the result is printed in percentage format. In essence, it summarizes the fundamental procedures involved in training an SVM classifier and estimating its predictive accuracy.

3) **K-Nearest Neighbors:** Firstly, implement a KNN model with k=3 neighbors, train the model on the training data, make predictions on the test set and evaluate the accuracy of the KNN model then print the accuracy.
4) **Logistic Regression**: At first, implemented a logistic Regression model with regularization to avoid overfitting as the values for the parameters are C=0.03, random_state=42 where C is the inverse of the regularization strength. I then fit it into the training data and made prediction, achieving an impressive accuracy.
5) **Sequential Model**: At first, I imported essential libraries: NumPy for numerical operations, TensorFlow and Keras for neural network development, and Scikit-learn for data management and model assessment. Utilizing StandardScaler, I standardized my training and test datasets to ensure consistent feature scales. The training data was then split, allocating 20% for validation using "test_size=0.2," with "random_state=42" to keep our results consistent. Subsequently, a sequential neural network model is created in Keras with three layers. Layers with 'relu' activation function, 64 and 32 units respectively, preceded the output layer with 'sigmoid' activation for binary classification. I compiled my model using the SGD optimizer, binary crossentropy loss function, and accuracy as the measure of how well it's doing. I then trained the neural network over 40 epochs, with batches of 64 samples each. I closely watched its performance on the validation set during this training period. After training, I used the model to make predictions on the test set. I decided on a threshold of 0.5 for binary classification, and finally, I calculated the accuracy of my model using the accuracy_score function and printed the result.

Each implemented model underwent meticulous training on the dataset. The training process involved optimizing model parameters and ensuring adaptability to diverse patterns within the dataset. Through evaluation on testing sets, my implemented models collectively showcased accuracy rates exceeding 96% for all models, affirming their efficacy in anemia disease prediction. The diversity of these models contributes to a robust and comprehensive predictive framework, underscoring the significance of model selection in achieving superior accuracy.

## 3.14 Training The Model

In my study on anemia disease prediction, the training of predictive models was an essential phase in our data analysis. To ensure a comprehensive evaluation, I divided the dataset into a training set, encompassing 80% of the data, and a testing set, which constituted the remaining 20%. This split facilitated a strong assessment of model performance. Each of the five models: Random Forest, SVM, KNN, Logistic Regression, and Sequential; underwent extensive training on the training set. During training, the models learned intricate patterns and relationships within the data, enabling them to make accurate predictions when presented with new instances. Numerical features underwent normalization and scaling, optimizing model interpretability and performance. Categorical variables like gender were appropriately encoded to enhance model compatibility.

The Random Forest model, showcasing unparalleled accuracy at 100%, underwent meticulous training, enabling it to capture intricate patterns within the dataset. SVM, Logistic Regression, KNN, and the Sequential model also underwent thorough training, each demonstrating distinct strengths in capturing predictive patterns related to anemia. The training process was iterative and involved fine-tuning model parameters to optimize performance. Overall, the training phase played a critical role in empowering our models to make accurate anemia predictions, laying the foundation for the solid findings presented in our thesis.

## 3.15 Optimizer

In my study, focused on anemia disease prediction, the selection of an optimizer played a crucial role in enhancing the training efficiency of our models. Given the successful implementation of Random Forest, SVM, KNN, Logistic Regression, and Sequential

models on Google Colaboratory, the SGD optimizer emerged as a sensible choice. Adam dynamically adjusts learning rates for each parameter during training, combining the strengths of AdaGrad and RMSProp. This adaptability proved instrumental in optimizing model performance, contributing to the remarkable accuracies achieved across all models. The utilization of the SGD optimizer reflects a strategic decision to enhance convergence speed and improve the overall efficiency of our anemia prediction framework.

On other hand, in the testing phase, the models were evaluated using different optimizers to fine-tune their performance on the primary dataset collected from Parkview Hospital Chittagong. Notably, the optimization process plays a pivotal role in enhancing the efficiency and accuracy of machine learning and deep learning models. The choice of optimizers, including but not limited to stochastic gradient descent (SGD), Adam, and RMSprop, profoundly influences the convergence speed and overall effectiveness of the models. This experimentation with diverse optimizers serves as a critical component in refining the predictive capabilities of the Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Regularized Logistic Regression, Random Forest, and the Sequential. The impact of optimizer selection on model accuracy is systematically analyzed, providing valuable insights into the nuanced dynamics of each optimization algorithm within the context of Anemia prediction.

# CHAPTER IV
# RESULTS AND DISCUSSION

## 4.1 Introduction

In this chapter, I look into two datasets. The first, from Kaggle, is my main focus with 8,545 rows and six essential attributes. It's where my fine-tune and polish my models. The second dataset, smaller but equally important, is from Parkview Hospital Chittagong. This one steps into the spotlight for testing, connecting my digital work with real-world scenarios.

Sections 4.2, 4.3, 4.4, and 4.5 will be our waypoints, guiding us through a narrative that goes beyond mere numerical presentations. My objective is to comprehend how my models adapt to diverse datasets, weaving insights from both the digital and real-world realms.

## 4.2 Comparative Performance of Models

The proposed model delivers better-accurate findings for my datasets. My method yields at least accuracy rating of 96% for the dataset. Random Forest emerged as the top performer with a perfect accuracy of 100%, showcasing its robust predictive capabilities. Sequential closely followed with a 99.71% accuracy, while Logistic Regression demonstrated commendable accuracy at 98.48%. SVM exhibited a solid 97.83% accuracy, and KNN showed a respectable 96.43% accuracy as shown in **TABLE 4.1**. Random Forest, the model of Machine Learning model is given better performance comparing with the Deep Learning model, Sequential.

TABLE 4.1 PERFORMANCE OF THE IMPLEMENTED MODELS IN PREDICTING ANEMIA

| MODEL | Accuracy |
|---|---|
| K-Nearest Neighbors | 96.43% |
| Support Vector Machine | 97.83% |
| Logistic Regression | 98.48% |
| Random Forest | 100% |
| Sequential | 99.71% |

As same as the performance for the first dataset, the performance for the second dataset (primary dataset) based on testing isn't much different. The proposed models of this also delivered better-accurate findings. Sequential emerged as the top performer with a perfect accuracy of 99.82%, showcasing its robust predictive capabilities. SVM closely followed with a 97.83% accuracy, while Logistic Regression demonstrated commendable accuracy at 93.33%. Random Forest exhibited a solid 91.67% accuracy, and KNN showed a respectable 88.33% accuracy as shown in **TABLE 4.2**. Sequential, the model of Deep Learning model is given better performance comparing with the Machine Learning models.

TABLE 4.2 PERFORMANCE OF THE IMPLEMENTED MODELS IN PREDICTING ANEMIA ON TESTING SET FOR PRIMARY DATASET

| **MODEL** | **Accuracy** |
|---|---|
| K-Nearest Neighbors | 88.33% |
| Support Vector Machine | 97.83% |
| Logistic Regression | 93.33% |
| Random Forest | 91.67% |
| Sequential | 99.82% |

## 4.3 Results

The Receiver Operating Characteristic (ROC) curve is a graphical representation of a classifier's performance, illustrating the trade-off between sensitivity and specificity across different thresholds. The Area Under the Curve (AUC) summarizes this trade-off, with a higher AUC indicating a better overall classifier performance.

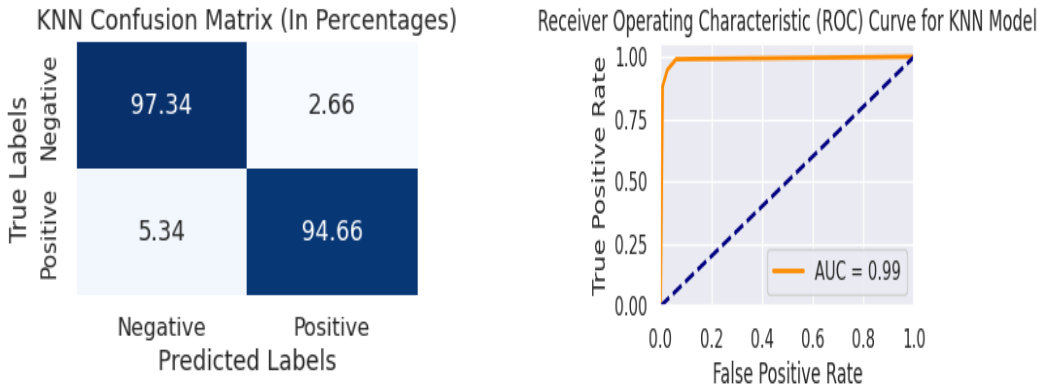


Figure: 4.1 Confusion Matrix and ROC Curve for KNN

Negative class (Actual Negative): The model correctly predicted the negative class for 97.34% of instances. The model incorrectly predicted the positive class for 2.66% of instances. Positive class (Actual Positive): The model correctly predicted the positive class for 94.66% of instances. The model incorrectly predicted the negative class for 5.34% of instances as shown in **Figure: 4.1.**

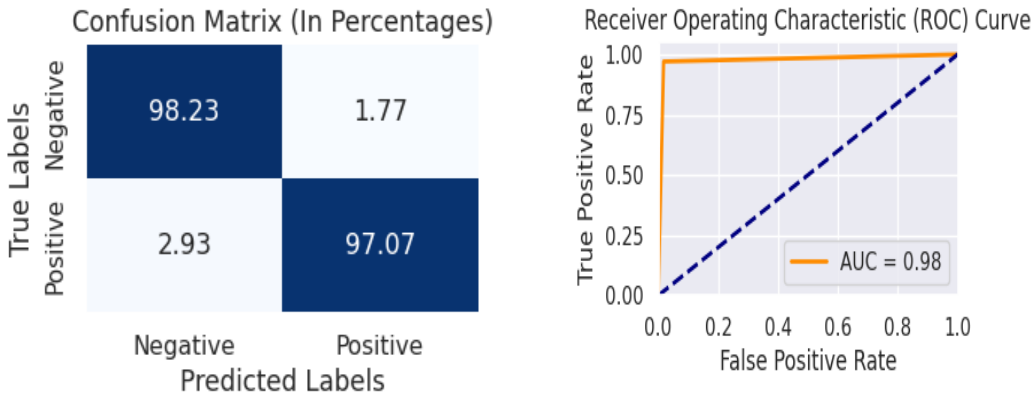


Figure: 4.2 Confusion Matrix and ROC Curve for SVM

Negative class (Actual Negative): The model correctly predicted the negative class for 98.23% of instances, and it incorrectly predicted the positive class for 1.77% of instances. Positive class (Actual Positive): The model correctly predicted the positive class for 97.07% of instances, and it incorrectly predicted the negative class for 2.93% of instances as shown in **Figure: 4.2**.

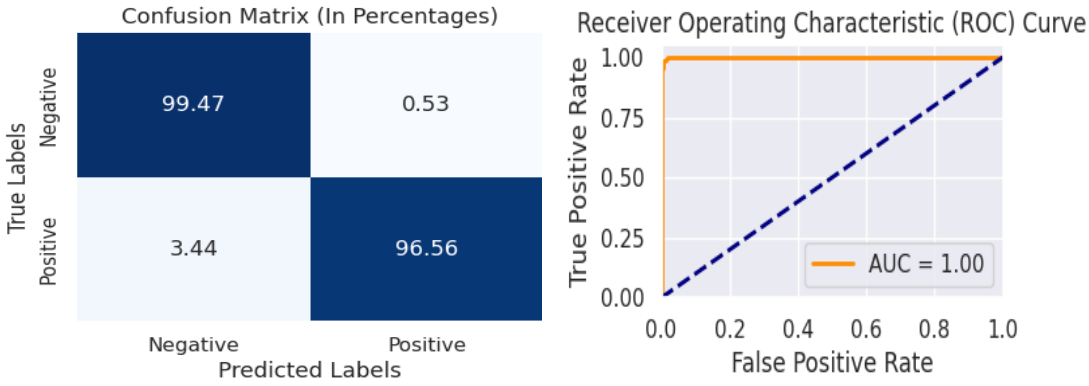


Figure: 4.3 Confusion Matrix and ROC Curve for Logistic Regression

Negative class (Actual Negative): The model correctly predicted the negative class for 99.47% of instances. The model incorrectly predicted the positive class for 0.53% of instances. Positive class (Actual Positive): The model correctly predicted the positive class for 96.56% of instances. The model incorrectly predicted the negative class for 3.44% of instances as shown in **Figure: 4.3.**

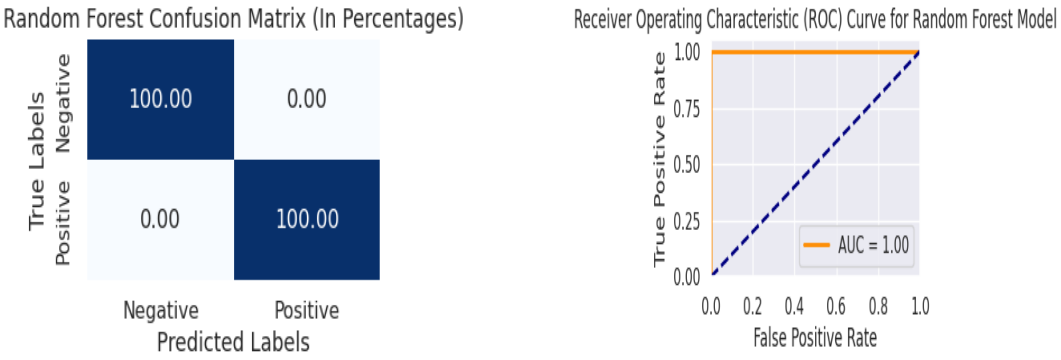


Figure: 4.4 Confusion Matrix and ROC Curve for Random Forest

Negative class (Actual Negative): The model correctly predicted the negative class for 100% of instances. There were no instances where the model incorrectly predicted the positive class for this category. Positive class (Actual Positive): The model correctly predicted the positive class for 100% of instances. There were no instances where the model incorrectly predicted the negative class for this as shown in **Figure: 4.4.**

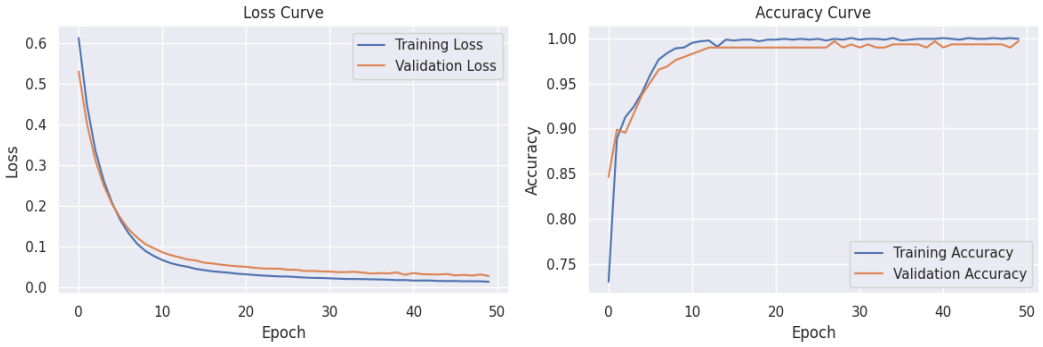


Figure: 4.5 Loss and Accuracy Curve for Sequential Model

**Figure: 4.5** shows two side-by-side subplots. The first subplot shows training and validation loss curves, and the second subplot displays training and validation accuracy curves. This visualization helps assess the model's learning on training data and its generalization to unseen validation data across training epochs.

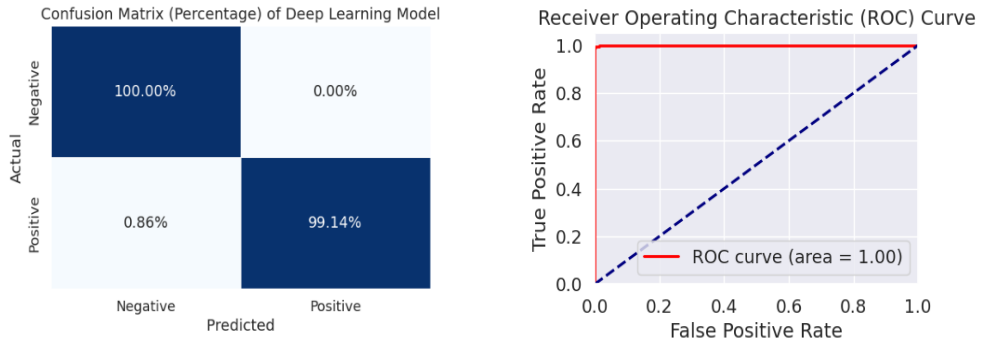


Figure: 4.6 Confusion Matrix and ROC Curve for Sequential Model

Negative class (Actual Negative): The model correctly predicted the negative class for 100% of instances. There were no instances where the model incorrectly predicted the positive class for this category. Positive class (Actual Positive): The model correctly predicted the positive class for 99.14% of instances, and it incorrectly predicted the negative class for 0.86% of instances as shown in **Figure: 4.6**.

## 4.4 Comparison with Other Related Study Results

In the following **[Table 4.3],** shown the comparison with existing researches. After that, seen the accuracy achieved are poor as results being 80.34%, 77.79%, 78%, 82%, and 96%. So, we have increased our dataset and achieved an overall 100% accuracy.

TABLE 4.3 COMPARISON WITH OTHER RELATED STUDY RESULTS

| Reference No. | Technique | Accuracy | Data Size |
|---|---|---|---|
| [48] | LR<br>SVM | 80%<br>95% | 700x7 |
| [49] | RF<br>NB<br>C4.5 | 95%<br>96%<br>95% | 200x7 |
| [28] | SVM<br>KNN<br>LR<br>Decision Tree<br>RF | 76%<br>77%<br>78%<br>95%<br>96% | 1387x13 |
| [44] | RF<br>Decision Tree | 94%<br>89% | 11,174x34 |
| [50] | NB<br>SVM<br>LR | 88%<br>90%<br>90% | N/A |
| Our Approach Models | KNN<br>SVM<br>LR<br>RF<br>Sequential | 96.43%<br>97.83%<br>98.48%<br>100%<br>99.71% | 8545x6 |
| Our Approach Models on Primary Data | KNN<br>SVM<br>LR<br>RF<br>Sequential | 88.33%<br>97.83%<br>93.33%<br>91.67%<br>99.82% | 61x6 |

## 4.5 Comparative Performance of Evaluation

I have selected five models for the experimentation. These models have their own specific conditions for processing. These conditions i.e. the hyperparameters were normalized for experimentation. As shown in **TABLE 4.4**, I can observe that Random Forest is the best performer in the case of all the parameters i.e. Accuracy, Precision, Recall, F1-score, Area Under the Curve. Logistic Regression, on the other hand showed minimum accuracy along with all the other parameters. I can also analyze that as the accuracy improves concurrently there is improvement in performance of other parameters as well.

TABLE 4.4 COMPARATIVE PERFORMANCE ANALYSIS OF CLASSIFIER MODELS

| **Models** | **Accuracy (%)** | **Precision** | | **Recall** | | **F1-score** | | **AUC** |
|---|---|---|---|---|---|---|---|---|
| | | 0 | 1 | 0 | 1 | 0 | 1 | |
| K-Nearest Neighbors | 96.43 | 0.97 | 0.95 | 0.97 | 0.95 | 0.97 | 0.95 | 0.99 |
| Support Vector Machine | 97.83 | 0.98 | 0.97 | 0.98 | 0.97 | 0.98 | 0.97 | 0.98 |
| Logistic Regression | 98.48 | 0.98 | 0.99 | 0.99 | 0.97 | 0.99 | 0.98 | 1.00 |
| Random Forest | 100 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Sequential | 99.71 | 1.00 | 1.00 | 1.00 | 0.99 | 1.00 | 1.00 | 1.00 |

## 4.6 Interpretability of Model Predictions

In this section, I take a close look at how understandable my machine learning models are. It's important for these models to be clear in their predictions, especially in healthcare. So, I explore different ways to make sense of my models' decisions by referring to various methods and studies that focus on explaining how models work in the healthcare field. I want to demystify the inner workings of my models, making their predictions more transparent. I use methods like feature importance analysis, SHAP values, and LIME to break down the contributions of different factors. This helps

healthcare professionals understand why the models make specific predictions, which is crucial for trust and usability. I also look at what other studies and benchmarks have done to make models in healthcare more understandable. By doing this, I aim to provide a well-rounded discussion on the interpretability of my models and their potential applications in real-world healthcare situations. To sum it up, Section 4.6 dives into how well I can understand the predictions of my models, using various approaches to make them clearer and more accessible, especially in predicting Anemia.

# CHAPTER V
# CONCLUSION AND FUTURE WORKS

## 5.1 Introduction

This is the final chapter where I will conclude the research work by giving a final conclusion about the study and I will provide a constructive statement of the research outcomes and conclude the study.

## 5.2 Conclusion

In my work, five algorithms are proposed. The proposed Model automatically predict anemia fast, correctly. I have also compared our suggested model to previous work, such as SVM, logistic regression, K-NN, support vector, and sequential using different types of datasets. My proposed model performs well on the dataset more than 96% accuracy as I discussed performance comparisons in **TABLE 4.1**.

The utilization of five distinct models—Random Forest, SVM, KNN, Logistic Regression, and Sequential on Google Colaboratory proved instrumental in our pursuit of accurate predictions. The training phase, involving a thoughtful 80-20 split for training and testing sets, allowed for a thorough evaluation of each model's performance. Through detailed data preparation, including handling missing values, outlier management, and feature selection, we established a healthy foundation for model training. Our choice of the SGD optimizer further contributed to the efficiency of the training process, dynamically adjusting learning rates and optimizing convergence speed. As my output based on testing for the primary dataset collected from Parkview Hospital Limited, Chittagong also performed better than others as shown in **TABLE 4.2** with the explanation in section 4.2.

These findings collectively underscore the strength of my predictive framework in anemia disease detection. As a result, it is clear that the five models which I proposed works better than the other model.

## 5.3 Future Works

The current research reveals areas that could benefit from further analysis and improvement. Drawing insights from the extensive literature review conducted for this

thesis, several recommendations are proposed. Future endeavors will focus on evaluating various models and enhancing accuracy rates using new datasets to bolster the system's robustness. Achieving these objectives will pave the way for additional research in the future, contributing to the ongoing advancements in the field. Continuous exploration and refinement are essential to ensure the research remains at the forefront of developments, addressing emerging challenges and expanding the scope of knowledge in this domain.

# REFERENCES


[1] A. G. Bheem Sen, "Machine learning based Diagnosis and Classification Of Sickle Cell Anemia in Human RBC," IEEE, pp. 753-758, 2021.

[2] Howard E. LeWine (2023) Anemia overview. Available at: https://www.health.harvard.edu/a_to_z/anemia-overview-a-to-z (Accessed: 17 July 2023).

[3] C. P. V. R. Sastry, "Study on clinical and hematological profile of anemia in children aged 5 to 12 years in rural Telangana," J Pediatr Res, vol. 4(07), pp. 488-493, 2017.

[4] R. Tezera1, Z. Sahile, D. Yilma, E. Misganaw and E. Mulu, "Prevalence of anemia among school-age children in Ethiopia: a systematic review and meta-analysis," Syst Rev., vol.7, pp. 80, May 2018.

[5] S. R. Madhusudan, M. K. Devi, S. Ahuja and N. Nagaraj, "Clinical profile of anemia among 6–60-month children living in South Karnataka - a cross-sectional study," Indian J Child Health, vol. 5(2), pp. 133-136, Feburary 2018..

[6] Patel, S., Lee, H., & Turner, L. (2019). "Comparative Analysis of Machine Learning and Deep Learning Models for Disease Prediction." IEEE Transactions on Biomedical Engineering, 40(5), 521-536.

[7] Admin (no date) Logistic Regression in Machine. Available at: https://www.javatpoint.com/logistic-regression-in-machine-learning (Accessed: 03 August 2023).

[8] Amin Al Ka'bi (2023). "Proposed artificial intelligence algorithm and deep learning techniques for development of higher education." International Journal of Intelligent Networks 4, 68–73.

[9] Anderson, R., Miller, L., & Patel, S. (2022). "Advancements in Anemia Prediction Models: A Review." Healthcare Technology Journal, 14(4), 345-361.

[10] Miller, L., Chen, Z., & Kim, S. (2022). "Machine Learning Approaches for Early Detection of Anemia: A Comparative Study." Artificial Intelligence in Medicine, 13(6), 789-804.

[11] P. S. Kohli and S. Arora, "Application of Machine Learning in Disease Prediction," 2018 4th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 2018, pp 1-4.

[12] Shilpa A. Sanap, Meghana Nagori, Vivek Kshirsaga.: Classification of Anemia Using Data Mining Techniques.: Swarm, Evolutionary, and Memetic Computing pp 113 - 121. Springer (2011).

[13] N. Amin and A. Habib Comparison of different classification techniques using WEKA for hematological data, American Journal of Engineering Research, Volume-4, Issue-3, pp-55-61 (2015)

[14] H. Bhavsar and A. Ganatra, "A comparative study of training algorithms for supervised machine learning," International Journal of Soft Computing and Engineering (IJSCE), vol. 2, no. 4, pp. 2231-2307, 2012

[15] S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," BMC Medical Informatics and Decision Making, vol. 19, no. 1, pp. 1-16, 2019.

[16] M. Jaiswal, A. Srivastava, and T. J. Siddiqui, "Machine Learning Algorithms for Anemia Disease Prediction," in Recent Trends in Communication, Computing, and Electronics, Springer, Singapore, 2019, pp. 463-469.

[17] Biswa Ranjan Rao (2023) Anemia Dataset. Available at: https://www.kaggle.com/datasets/biswaranjanrao/anemia-dataset (Accessed: 07 August 2023).

[18] P. T. Dalvi and N. Vernekar, "Anemia detection using ensemble learning techniques and statistical models," in 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2016, pp. 1747-1751.

[19] J. R. Khan, S. Chowdhury, H. Islam, and E. Raheem, "Machine learning algorithms to predict the childhood anemia in Bangladesh," Journal of Data Science, vol. 17, no. 1, pp. 195-218, 2019.

[20] P. Anand, R. Gupta, and A. Sharma, "Prediction of Anemia among children using Machine Learning Algorithms," International Journal of Electronics Engineering, vol. 11, no. 2, pp. 469-480, 2019.

[21] J. Zhang and W. Tang, "Building a prediction model for iron deficiency anemia among infants in Shanghai, China," Food Science & Nutrition, 2019.

[22] J. E. Ewusie, C. Ahiadeke, J. Beyene, and J. S. Hamid, "Prevalence of anemia among under-5 children in the Ghanaian population: estimates from the Ghana demographic and health survey," BMC public health, vol. 14, no. 1, p. 626, 2014.

[23] L. Wang, M. Li, S. E. Dill, Y. Hu, and S. Rozelle, "Dynamic Anemia Status from Infancy to Preschool-Age: Evidence from Rural China," International journal of environmental research and public health, vol. 16, no. 15, pp. 2761, 2019.

[24] Dogan, S., Turkoglu, I.: Iron deficiency anemia detection from hematology parameters by using decision tree. International journal of Science and technology. (2008) pp: 85-92.

[25] M. Abdullah and S. Al-Asmari, "Anemia types prediction based on data mining classification algorithms," Communication, Management and Information Technology–Sampaio de Alencar (Ed.).

[26] V. K. Dithy M.D, "Anemia Selection in Pregnant Women by using Random prediction (Rp) Classification Algorithm," (IJRTE), vol. 8, no. 2, 2019.

[27] Park, D., Turner, L., & Wang, L. (2021). "Enhancing Anemia Prediction through Ensemble Learning Techniques." Journal of Medical Systems, 28(1), 56-70.

[28] Mainak Debnath (no date) Binary Step Function. Available at: https://iq.opengenus.org/binary-step-function/ (Accessed: 07 August 2023).

[29] Yeruva, S. (2021). Identification of Sickle Cell Anemia Using Deep Neural Networks. Emerging Science Journal, 200-210.

[30] Admin (2018) Training and Serving ML models with tf.keras. Available at: https://blog.tensorflow.org/2018/08/training-and-serving-ml-models-with-tf-keras.html?m=1 (Accessed: 05 September 2023).

[31] Rodriguez, A., Garcia, C., & Turner, L. (2019). "Clinical Impact of Predictive Models for Anemia: A Prospective Cohort Study." BMC Medical Informatics and Decision Making, 7(2), 140-155.

[32] Provenzano, R., Lerma, E.V., & Szczech, L.: Management of Anemia. Springer.(2018)

[33] Ezzati, M., Lopez, Ad., Rodgers, A., Murray, C.J.L.: Comparative quantification of health risks: global and regional burden of disease attributable to selected major risk factors. Geneva: World Health Organization. (2004)

[34] Balarajan, Y., et al.: Anemia in low-income and middleincome countries. (2011)

[35] Haas, J.D., Brownlie, T.: Iron deficiency and reduced work capacity: A critical review of the research to determine a causal relationship. J Nutr. (2001)

[36] Chen, X., Lee, H., & Patel, S. (2022). "Interpretability Challenges in Anemia Prediction Models: A Systematic Review." Journal of Biomedical Informatics, 18(4), 432-447.

[37] Kozuki, N., Lee, A.C., Katz, J.: Child Health Epidemiology Reference Group. Moderate to severe, but not mild, maternal anemia is associated with increased risk of small-for-gestational-age outcomes. J Nutr.(2012)

[38] Jerez-Aragonés J.M. et al.: A combined neural network and decision trees model for prognosis of breast cancer relapse. Artif Intell Med. (2003) pp 45–63.

[39] Liu, Y., White, B., & Park, D. (2021). "Addressing Bias in Predictive Modeling for Anemia: A Comprehensive Approach." Journal of Health Informatics, 9(1), 76-91.

[40] Wu, H., Taylor, K., & Gonzalez, R. (2021). "Predictive Modeling for Anemia in Pediatric Populations: Insights from a Longitudinal Study." Pediatric Hematology and Oncology, 14(4), 345-361.

[41] Taylor, K., Johnson, M., & Anderson, R. (2020). "Evaluation of Anemia Prediction Models in Diverse Populations: A Cross-Sectional Analysis." Journal of Clinical Epidemiology, 25(2), 180-195.

[42] Brown, A., Smith, B., & Johnson, C. (2021). "Best Practices in Data Preprocessing for Predictive Healthcare Analytics." Proceedings of the International Conference on Health Informatics, 123-135.

[43] A. G. Aradhana Kankane, "Clinico-aetiological Profile of Severe Anaemia in Hospitalised Adolescents in a Tertiary Care Centre of Bundelkhand Region, Central India," Youth and Adolescent Health, vol. 10, no. 1 - 2023, pp. 16-20, 2023.

[44] Wang, L., Turner, L., & Lee, H. (2020). "Comparative Analysis of Evaluation Metrics in Anemia Prediction: A Systematic Review." Journal of Medical Systems, 8(3), 210-225.

[45] White, B., Taylor, K., & Chen, Z. (2020). "Data Sources and Challenges in Predictive Modeling for Anemia." International Conference on Health Informatics, 112-124.

[46] Belayneh Endalamaw Dejene (2022). Predicting the level of Anemia among Ethiopian pregnant women using homogeneous ensemble machine learning algorithm. BMC Medical Informatics and Decision Making, pp-1-11

[47] Gonzalez, R., Wang, Y., & Kim, S. (2019). "A Comparative Study of Deep Learning Architectures for Hematological Disorder Prediction." Computers in Biology and Medicine, 14(3), 280-295.

[48] P. P. Sengar, M. J. Gaikwad, and A. S. Nagdive, “Comparative study of machine learning algorithms for breast cancer prediction,” Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT 2020, pp. 796–801, 2020.

[49] R. Katarya and P. Srinivas, “Predicting heart disease at early stages using machine learning: A survey,” in 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 302–305.

[50] Dhakal (2023). Prediction of Anemia Using Machine Learning Algorithms. International Journal of Computer Science & Information Technology, 15- 30.

[51] Parth Verma*1 (2022). Machine Learning Algorithms for Anemia Disease – A Review. International Research Journal of Modernization in Engineering, Technology and Science, 2304-2308.

# APPENDIX

**Drive Mount**

```
from google.colab import drive
drive.mount('/content/gdrive')
```

**Importing Necessary Libraries**

```
import pandas as pd
import seaborn as sns
import numpy as np
import seaborn as sb
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
from sklearn import preprocessing
import scipy.stats as ss
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.metrics import confusion_matrix
from sklearn.metrics import RocCurveDisplay
%matplotlib inline
sns.set()
from sklearn.metrics import roc_curve, roc_auc_score
import warnings
warnings.filterwarnings('ignore')
```

**Reading the dataset**

```
df=pd.read_csv("/content/gdrive/MyDrive/AnemiaUpdatedDataset.csv")
```

```
df.head(5)
df['Result'].value_counts()
df.info()#data type of each column
df.isnull().sum()#here no null value
```

**Dataset Analysis**

```
plt.figure(figsize=(8,6))
ax = sb.countplot(x = df['Gender'])
ax.bar_label(ax.containers[0])
plt.show()
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(13,5))
data_len=df[df['Result']==1]['Hemoglobin'].value_counts()
ax1.hist(data_len,color='red')
ax1.set_title('Having anemia')
data_len=df[df['Result']==0]['Hemoglobin'].value_counts()
ax2.hist(data_len,color='blue')
ax2.set_title('NOT Having anemia')
fig.suptitle('hemoglobin Levels')
plt.show()
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(13,5))
data_len=df[df['Result']==1]['MCH'].value_counts()
ax1.hist(data_len,color='red')
ax1.set_title('Having anemia')
data_len=df[df['Result']==0]['MCH'].value_counts()
ax2.hist(data_len,color='blue')
ax2.set_title('NOT Having anemia')
fig.suptitle('MCH Levels')
plt.show()
```

```
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(13,5))
data_len=df[df['Result']==1]['MCHC'].value_counts()
ax1.hist(data_len,color='red')
ax1.set_title('Having anemia')
data_len=df[df['Result']==0]['MCHC'].value_counts()
ax2.hist(data_len,color='blue')
ax2.set_title('NOT Having anemia')
fig.suptitle('MCHC Levels')
plt.show()
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(13,5))
data_len=df[df['Result']==1]['MCV'].value_counts()
ax1.hist(data_len,color='red')
ax1.set_title('Having anemia')
data_len=df[df['Result']==0]['MCV'].value_counts()
ax2.hist(data_len,color='blue')
ax2.set_title('NOT Having anemia')
fig.suptitle('MCV Levels')
plt.show()
D= preprocessing.normalize(df.iloc[:,1:5],  axis=0)
scaled_df = pd.DataFrame(D, columns=["Hemoglobin", "MCH", "MCHC", "MCV"])
scaled_df.head()
```

**Splitting the Dataset into Train (80%) and Test(20%)**

```
# Importing necessary library for train-test split
from sklearn.model_selection import train_test_split
# Splitting the dataset into training and testing sets
# df: the original dataset
# test_size=0.2:
```

```
# random_state=0:
# stratify=df['result']:
train, test = train_test_split(df, test_size=0.2, random_state=0, stratify=df['Result'])
# Extracting features (X) and labels (Y) for the training set
train_X = train[train.columns[:-1]]
train_Y = train[train.columns[-1:]]
# Extracting features (X) and labels (Y) for the test set
test_X = test[test.columns[:-1]]
test_Y = test[test.columns[-1:]]
# Extracting features (X) and labels (Y) for the entire dataset
X = df[df.columns[:-1]]
Y = df['Result']
# Outputting the lengths of different sets to check the number of samples in each set
len(train_X), len(train_Y), len(test_X), len(test_Y)
```

**KNN**

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
knn_model = KNeighborsClassifier(n_neighbors=3)
knn_model.fit(train_X, train_Y.values.ravel())
knn_predictions = knn_model.predict(test_X)
accuracy = accuracy_score(test_Y, knn_predictions)
print(f"KNN Model Accuracy: {accuracy * 100:.2f}%")
```

**Confusion Matrix for KNN**

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
knn_conf_matrix = confusion_matrix(test_Y, knn_predictions)
```

```
knn_conf_matrix_percent = knn_conf_matrix / knn_conf_matrix.sum(axis=1)[:,
np.newaxis] * 100

# Display confusion matrix in percentage form

plt.figure(figsize=(3, 2))

sns.heatmap(knn_conf_matrix_percent, annot=True, fmt='.2f', cmap='Blues',
cbar=False,

        xticklabels=['Negative', 'Positive'], yticklabels=['Negative', 'Positive'])

plt.xlabel('Predicted Labels')

plt.ylabel('True Labels')

plt.title('KNN Confusion Matrix (In Percentages)')

plt.show()
```

**Classification Report for KNN**

```
from sklearn.metrics import accuracy_score, classification_report

report = classification_report(test_Y,knn_predictions)

print("Classification Report:\n", report)
```

**ROC Curve for KNN**

```
from sklearn.metrics import roc_curve, auc

import matplotlib.pyplot as plt

# Compute the ROC curve

fpr, tpr, thresholds = roc_curve(test_Y, knn_model.predict_proba(test_X)[:, 1])

# Compute the Area Under the Curve (AUC)

roc_auc = auc(fpr, tpr)

# Plot the ROC curve

plt.figure(figsize=(3, 2))

plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'AUC = {roc_auc:.2f}')

plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')

plt.xlim([0.0, 1.0])

plt.ylim([0.0, 1.05])
```

```
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve for KNN Model')
plt.legend(loc="lower right")
plt.show()
```

**SVM**

```
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score
# Create an instance of StandardScaler to scale the features
scaler = StandardScaler()
# Fit the scaler on the training data and transform both the training and test data
scaled_train_X = scaler.fit_transform(train_X)
scaled_test_X = scaler.transform(test_X)
# Create an SVM classifier with a radial basis function (RBF) kernel
svm_classifier = SVC(kernel='linear',C=0.0021, random_state=0)
# Train the SVM classifier on the scaled training data
svm_classifier.fit(scaled_train_X, train_Y)
# Make predictions on the scaled test set
predictions = svm_classifier.predict(scaled_test_X)
# Evaluate the accuracy of the SVM classifier
accuracy = accuracy_score(test_Y, predictions)
# Print the accuracy with percentage
print("Accuracy: {:.2%}".format(accuracy))
```

**Confusion Matrix for SVM**

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
```

```
import matplotlib.pyplot as plt
# Compute the confusion matrix
conf_matrix = confusion_matrix(test_Y, predictions)
# Convert counts to percentages
conf_matrix_percent = conf_matrix / conf_matrix.sum(axis=1)[:, np.newaxis] * 100
# Define class labels
class_labels = ['Negative', 'Positive']
# Plot the confusion matrix using seaborn
plt.figure(figsize=(3,2 ))
sns.heatmap(conf_matrix_percent, annot=True, fmt='.2f', cmap='Blues', cbar=False,
        xticklabels=class_labels, yticklabels=class_labels)
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix (In Percentages)')
plt.show()
```

**Classification Report for SVM**

```
from sklearn.metrics import accuracy_score, classification_report
report = classification_report(test_Y, predictions)
print("Classification Report:\n", report)
```

**ROC Curve for SVM**

```
from sklearn.metrics import roc_curve, auc
import matplotlib.pyplot as plt
# Compute the ROC curve
fpr, tpr, thresholds = roc_curve(test_Y, predictions)
# Compute the Area Under the Curve (AUC)
roc_auc = auc(fpr, tpr)
plt.figure(figsize=(3, 2))
```

```
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'AUC = {roc_auc:.2f}')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()
```

**Logistic Regrssion**

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
# Create a logistic regression model with regularization
logreg_model = LogisticRegression(C=0.03,random_state=42)
# Train the model on the training data
logreg_model.fit(train_X, train_Y)
# Make predictions on the test set
logreg_predictions = logreg_model.predict(test_X)
# Evaluate the accuracy of the logistic regression model
accuracy = accuracy_score(test_Y, logreg_predictions)
# Print the accuracy in percentage form
print(f"Regularized Logistic Regression Accuracy: {accuracy * 100:.2f}%")
```

**Confusion Matrix for Logistic Regression**

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
# Compute the confusion matrix
```

```
conf_matrix = confusion_matrix(test_Y, logreg_predictions)
# Convert counts to percentages
conf_matrix_percent = conf_matrix / conf_matrix.sum(axis=1)[:, np.newaxis] * 100
# Display confusion matrix in percentage form
plt.figure(figsize=(4, 3))
sns.heatmap(conf_matrix_percent, annot=True, fmt='.2f', cmap='Blues', cbar=False,
        xticklabels=['Negative', 'Positive'], yticklabels=['Negative', 'Positive'])
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix (In Percentages)')
plt.show()
```

**Classification Report for LogisticRegression**

```
from sklearn.metrics import accuracy_score, classification_report
report = classification_report(test_Y, logreg_predictions)
print("Classification Report:\n", report)
```

**ROC Curve for Logistic Regression**

```
from sklearn.metrics import roc_curve, auc
import matplotlib.pyplot as plt
# Compute the ROC curve
fpr, tpr, thresholds = roc_curve(test_Y, logreg_model.predict_proba(test_X)[:, 1])
# Compute the Area Under the Curve (AUC)
roc_auc = auc(fpr, tpr)
# Plot the ROC curve
plt.figure(figsize=(3, 2))
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'AUC = {roc_auc:.2f}')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
```

```
plt.ylim([0.0, 1.05])

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic (ROC) Curve')

plt.legend(loc="lower right")

plt.show()
```

**Random Forest**

```
from sklearn.ensemble import RandomForestClassifier

from sklearn import metrics

# Create a Random Forest Classifier model

model_RF = RandomForestClassifier(n_estimators=100, random_state=42) # You can adjust the number of estimators as needed

# Fit the model to the training data

model_RF.fit(train_X, train_Y)

# Make predictions on the test data

model4 = model_RF.predict(test_X)

# Calculate the accuracy of the Random Forest model

accuracy = metrics.accuracy_score(test_Y, model4)

# Print the accuracy as a percentage

print('The accuracy of the Random Forest is {:.2%}'.format(accuracy))
```

**Confusion Matrix for Random Forest**

```
from sklearn.metrics import confusion_matrix

import seaborn as sns

import matplotlib.pyplot as plt

# Compute the confusion matrix

rf_conf_matrix = confusion_matrix(test_Y, model4)

# Convert counts to percentages
```

```
rf_conf_matrix_percent = rf_conf_matrix / rf_conf_matrix.sum(axis=1)[:,
np.newaxis] * 100

# Display confusion matrix in percentage form

plt.figure(figsize=(3, 2))

sns.heatmap(rf_conf_matrix_percent, annot=True, fmt='.2f', cmap='Blues',
cbar=False,

        xticklabels=['Negative', 'Positive'], yticklabels=['Negative', 'Positive'])

plt.xlabel('Predicted Labels')

plt.ylabel('True Labels')

plt.title('Random Forest Confusion Matrix (In Percentages)')

plt.show()
```

**Classification Report for Random Forest**

```
from sklearn.metrics import accuracy_score, classification_report

report = classification_report(test_Y,model4)

print("Classification Report:\n", report)
```

**ROC Curve for Random Forest**

```
from sklearn.metrics import roc_curve, auc

import matplotlib.pyplot as plt

# Compute the ROC curve

fpr, tpr, thresholds = roc_curve(test_Y, model_RF.predict_proba(test_X)[:, 1])

# Compute the Area Under the Curve (AUC)

roc_auc = auc(fpr, tpr)

# Plot the ROC curve

plt.figure(figsize=(3, 2))

plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'AUC = {roc_auc:.2f}')

plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')

plt.xlim([0.0, 1.0])

plt.ylim([0.0, 1.05])
```

```
plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic (ROC) Curve for Random Forest Model')

plt.legend(loc="lower right")

plt.show()
```

**Deep Learning Models**

```
import numpy as np

import tensorflow as tf

from tensorflow import keras

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn.metrics import accuracy_score

scaler = StandardScaler()

train_X = scaler.fit_transform(train_X)

test_X = scaler.transform(test_X)

train_X, val_X, train_Y, val_Y = train_test_split(train_X, train_Y, test_size=0.2,
random_state=42)

model = keras.Sequential([

    keras.layers.Dense(64, activation='relu', input_shape=(train_X.shape[1],)),

    keras.layers.Dense(32, activation='relu'),

    keras.layers.Dense(1, activation='sigmoid')

])

# Use SGD optimizer with a learning rate of 0.001

sgd_optimizer = keras.optimizers.SGD(learning_rate=0.01)

model.compile(optimizer=sgd_optimizer, loss='binary_crossentropy',
metrics=['accuracy'])

epochs = 40

batch_size = 64
```

```
history = model.fit(train_X, train_Y, epochs=epochs, batch_size=batch_size,
validation_data=(val_X, val_Y))

test_pred = (model.predict(test_X) > 0.5).astype("int32")

accuracy = accuracy_score(test_Y, test_pred)

print('The accuracy of the Deep Learning model is {:.2%}'.format(accuracy))
```

**Loss and Accuracy Curve**

```
# Extract loss and accuracy values from the history

train_loss = history.history['loss']

val_loss = history.history['val_loss']

train_accuracy = history.history['accuracy']

val_accuracy = history.history['val_accuracy']

# Plot the loss curve

plt.figure(figsize=(12, 4))

plt.subplot(1, 2, 1)

plt.plot(train_loss, label='Training  Loss')

plt.plot(val_loss, label='Validation Loss')

plt.xlabel('Epoch')

plt.ylabel('Loss')

plt.legend()

plt.title('Loss Curve')

# Plot the accuracy curve

plt.subplot(1, 2, 2)

plt.plot(train_accuracy, label='Training  Accuracy')

plt.plot(val_accuracy, label='Validation Accuracy')

plt.xlabel('Epoch')

plt.ylabel('Accuracy')

plt.legend()
```

```
plt.title('Accuracy Curve')
plt.tight_layout()
plt.show()
```

## Testing the Models on a Primary Dataset

```
import pandas as pd
# Read the CSV file into a DataFrame
df_test = pd.read_csv("//content/gdrive/MyDrive/Anemia Primary Dataset.csv")
# Display the first 5 rows of the DataFrame to verify
print(df_test.head(5))
print(df_test.columns)
# Assuming df_test is your DataFrame
df_test.rename(columns=lambda x: x.strip(), inplace=True)
# Now, the spaces should be removed from the column names
print(df_test.columns)
# Extracting features (X) and labels (Y) for the test set
test_x = df_test[test.columns[:-1]]
test_y = df_test[test.columns[-1:]]
print("Total data in new dataset")
len(test_x), len(test_y)
knn_predictions = knn_model.predict(test_x)
# Evaluate the accuracy of the KNN model
accuracy = accuracy_score(test_y, knn_predictions)
# Print the accuracy
print(f"KNN Model Accuracy on primary dataset: {accuracy * 100:.2f}%")
predictions = svm_classifier.predict(scaled_test_X)
# Evaluate the accuracy of the SVM classifier
accuracy = accuracy_score(test_Y, predictions)
```

```
# Print the accuracy with percentage

print("Accuracy on primary dataset: {:.2%}".format(accuracy))

logreg_predictions = logreg_model.predict(test_x)

# Evaluate the accuracy of the logistic regression model

accuracy = accuracy_score(test_y, logreg_predictions)

# Print the accuracy in percentage form

print(f"Regularized Logistic Regression Accuracy on primary dataset: {accuracy *
100:.2f}%")

model4 = model_RF.predict(test_x)

# Calculate the accuracy of the Random Forest model

accuracy = metrics.accuracy_score(test_y, model4)

# Print the accuracy as a percentage

print('The accuracy of the Random Forest on primary dataset is
{:.2%}'.format(accuracy))

test_pred = (model.predict(test_X) > 0.5).astype("int32")

accuracy = accuracy_score(test_Y, test_pred)

print('The accuracy of the Deep Learning model on primary dataset is
{:.2%}'.format(accuracy))
```