KANTIPUR ENGINEERING COLLEGE

(Affiliated to Tribhuvan University)

Dhapakhel, Lalitpur

[Subject Code: CT654]

A MINOR PROJECT FINAL REPORT ON

THYROID CLASSIFICATION WITH L1 REGULARIZATION FOR EFFECTIVE FEATURE SELECTION

Submitted by:

Aadarsha Regmi	**[31051]**
Angel Tamang	**[31060]**
Anil Bhatta	**[31061]**
Manish Karki	**[31090]**

A MINOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

March, 2024

THYROID CLASSIFICATION WITH L1 REGULARIZATION FOR EFFECTIVE FEATURE SELECTION

Submitted by:

Aadarsha Regmi [31051]

Angel Tamang [31060]

Anil Bhatta [31061]

Manish Karki [31090]

A MINOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

Kantipur Engineering College

Dhapakhel, Lalitpur

March, 2024

KANTIPUR ENGINEERING COLLEGE
DEPARTMENT OF COMPUTER AND ELECTRONICS ENGINEERING

APPROVAL LETTER

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a project report entitled ”Thyroid Classification with L1 Regularization for Effective Feature Selection” submitted by

Aadarsha Regmi [31051]

Angel Tamang [31060]

Anil Bhatta [31061]

Manish Karki [31090]

in partial fulfillment for the degree of Bachelor in Computer Engineering.

.....
Er. Bishal Thapa
Project Coordinator
Department of Computer and Electronics Engineering

.....
External Examiner
External
External’s Designation

.....
Er. Rabindra Khati
Associate Professor
Head of Department
Department of Computer and Electronics Engineering

Date: March 1, 2024

ACKNOWLEDGMENT

We are grateful to everyone who contributed to the completion of this project. Firstly, we acknowledge the invaluable guidance, support, and feedback from our teachers in the Department of Electronics and Computer Engineering. We also extend our thanks to our classmates for their constructive feedback and engaging discussions about our project. Additionally, we appreciate the guidance provided by all lecturers in our department from the project's inception to its completion. Lastly, we express special gratitude to our family and friends for their love, encouragement, and support throughout our academic journey.

Thank you all for your invaluable contributions to this project.

Aadarsha Regmi [31051]

Angel Tamang [31060]

Anil Bhatta [31061]

Manish Karki [31090]

ABSTRACT

Thyroid disease, mainly caused by disfunctioning thyroid gland that plays a crucial role in regulating various bodily functions, has emerged as a serious health concern for many individuals as it is more common than one can imagine imbalances in thyroid hormone levels can affect metabolism, energy production and numerous organs. The early diagnosis of thyroid disease may benefit from the use of machine learning. The project utilizes max-voting ensembled classifier to identify thyroid conditions, namely hypothyroidism and hyperthyroidism. The dataset acquired is balanced to prevent bias by the models used, and to identify relevant features among available attributes of the data, L1 regularization was utilized which is a natural candidate in a feature selection process. L1 regularization imposes a penalty term to the loss function of a model during optimization creating a sparse features vector. Thus, irrelevant features coefficients is drived to zero encouraging effective feature selection. The classification model is reliable with high recall value observed during its evaluation on testing datas.

Keywords *−Machine learning, Thyroid disease, Classification model, Decision tree, Random forest, Support Vector Machines, Multi-Layer Perceptron, Ensemble, L1 regularization, Streamlit.*

TABLE OF CONTENTS

Approval Letter i

Acknowledgment ii

Abstract iii

List Of Figures vi

List Of Tables vii

Abbreviations viii

1 Introduction 1

1.1 Background 1

1.2 Problem Statement 2

1.3 Objectives 3

1.4 Application Scope 3

1.5 Features 3

1.6 System Requirements 4

1.6.1 Development Requirements 4

1.6.2 Deployment Requirements 4

1.7 Feasibility Study 4

1.7.1 Economic Feasibility 4

1.7.2 Operational Feasibility 5

1.7.3 Schedule Feasibility 5

2 Litreature Review 6

2.1 Related Research 6

3 Methodology 8

3.1 Working Mechanism 8

3.1.1 Data Collection 8

3.1.2 Data Preprocessing 9

3.1.3 Machine Learning Technique 9

3.1.4 Split Dataset 15

3.1.5 Train Model 15

3.1.6 Model validation 16

3.1.7 Model Output 16

3.2 Development Model 16
3.2.1 Requirement Specification 16
3.2.2 Development and Implementation 17
3.2.3 Verification and Validation 17
3.2.4 Increments 17
3.3 Evaluation Criteria 18
3.4 System Diagrams 18
3.4.1 Usecase Diagram 19
3.4.2 DFD 19
3.4.3 Class Diagram 20
3.4.4 Activity Diagram 20
4 Result And Discussion **21**
4.1 Preprocessing 21
4.2 Model Training 25
5 Conclusion And Future Enhancements **30**
5.1 Conclusion 30
5.2 Future Enhancements 30
References **32**
Annex **33**

LIST OF FIGURES

1.1 Symptoms 2
1.2 Gantt Chart 5
3.1 Block diagram of Thyroid Classification 8
3.2 Block diagram of Data Preprocessing 9
3.3 Decision Tree 10
3.4 Working diagram of Random Forest 12
3.5 SVM 13
3.6 MLP structure 14
3.7 Incremental Development Model 17
3.8 Usecase Diagram 19
3.9 DFD diagram 19
3.10 Class diagram 20
3.11 Activity Diagram 20
4.1 Strip plot: Numerical Attributes vs Target 22
4.2 Feature Selection 23
4.3 Overfitting Decision Tree 25
4.4 Decision Tree on Balanced Dataset 26
4.5 Evaluation of Unbalanced Decision Tree 27
4.6 Evaluation of Balanced Decision Tree 27
4.7 Evaluation of Random Forest 28
4.8 Evaluation of MLP 28
4.9 Evaluation of Ensemble 29
5.1 Result 1 33
5.2 Result 2 33
5.3 Result 3 34
5.4 Result 4 34

LIST OF TABLES

1.1 Table 1 Development Requirements 4
1.2 Table 2 Deployment Requirements 4
4.1 Thyroid disease attributes & dataset info 24
4.2 Performance of the Models 26

ABBREVIATIONS

DTC	Decision Tree Classification
EDA	Exploratory Data Analysis
ML	Machine Learning
ROC	Receiver Operating Characteristics
PR	Precision-Recall
RF	Random Forest
MLP	Multi-Layer Perceptron
SVM	Support Vector Machines

CHAPTER 1
INTRODUCTION

1.1 Background

Thyroid disease is a significant global health concern, affecting millions of people worldwide. The thyroid gland, a vital organ in our body, plays a crucial role in metabolism, growth, and development. It produces two main hormones, thyroxine (T4) and triiodothyronine (T3), which are essential for the body's metabolic processes. The production of these hormones is regulated by thyroid-stimulating hormone (TSH), which is released by the pituitary gland. An imbalance in these hormones can lead to thyroid diseases. Thyroid diseases can be broadly classified into conditions that affect the structure of the gland, such as goiter and thyroid nodules, and those that affect the function of the gland, such as hypothyroidism, hyperthyroidism, and thyroiditis. Hypothyroidism is a condition where the thyroid gland does not produce enough thyroid hormones, leading to symptoms like fatigue, weight gain, and depression. On the other hand, hyperthyroidism is a condition where the thyroid gland produces too much thyroid hormones, leading to symptoms like rapid heart rate, weight loss, and anxiety. Thyroiditis is an inflammation of the thyroid gland, which can cause either hyperthyroidism or hypothyroidism. More serious conditions include thyroid cancer and autoimmune thyroid diseases, such as Graves' disease and Hashimoto's thyroiditis. In the context of Nepal, thyroid disorders are prevalent, with a study showing that the prevalence of thyroid dysfunction was 17.42% among the population of the western region of Nepal [1]. However, this project will not be using a dataset specific to Nepal.

Machine learning, a subset of artificial intelligence, has shown great promise in the field of healthcare, particularly in disease diagnosis. It has the potential to improve the accuracy and speed of diagnosis. In recent years, there has been a growing interest in applying machine learning algorithms for thyroid disease classification.

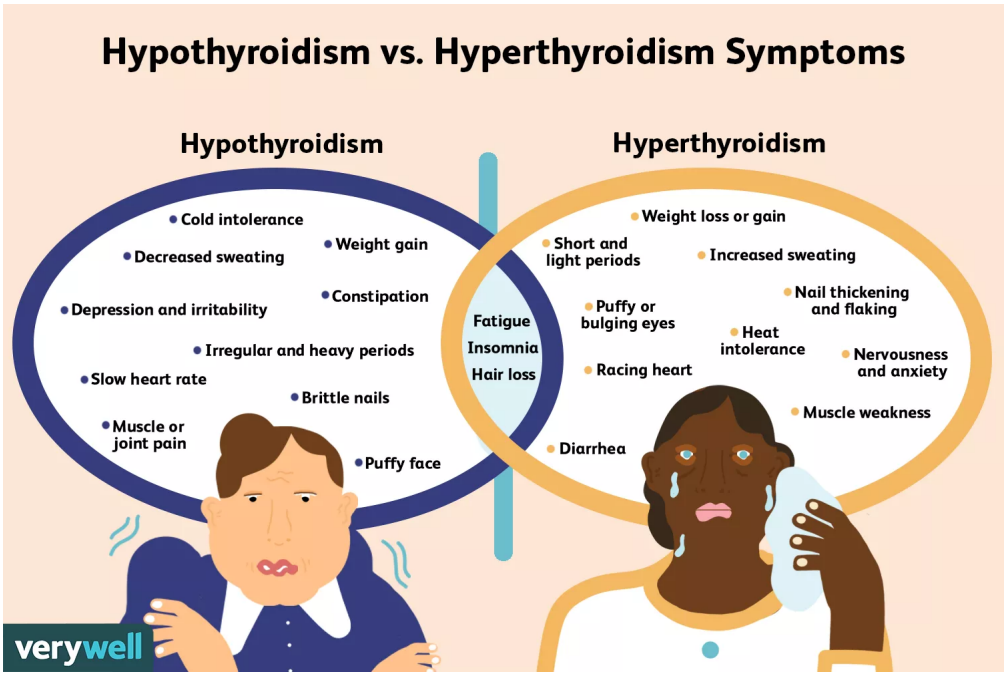

Figure 1.1: Symptoms
Source:https://www.verywellhealth.com/hypothyroidism-hyperthyroidism-5180646

1.2 Problem Statement

Thyroid diseases are a significant health concern due to their prevalence and potential impact on quality of life. Accurate and timely diagnosis is crucial for effective treatment and management. The interpretation of thyroid function tests can be complex due to various influencing factors such as Thyroid Stimulating Hormone(TSH), thyroxine (T4) and triiodothyronine (T3). Machine learning algorithms, such as Decision trees, Random Forest, MLP offer a promising solution to these challenges. They have the potential to improve the accuracy and speed of diagnosis by learning from patterns in the data. However, their application in thyroid disease classification is still an emerging field and needs further research. This project aims to develop a Classification model for thyroid disease classification using a dataset that is not specific to any particular region.

1.3 Objectives

To classify thyroid diseases, the specific objectives are as follows:

i To build classification models for thyroid disease classification.
ii To provide a web-tool to the users for reliable thyroid testing.

1.4 Application Scope

The application of machine learning in the classification of thyroid diseases has the potential to revolutionize the way these diseases are diagnosed and managed. The model in this project could serve as a valuable tool for healthcare professionals, aiding in the accurate and timely diagnosis of thyroid diseases. The scope of this project is broad, as the dataset used for model development is not specific to any particular region. This enhances the applicability and relevance of the project findings, potentially benefiting a wider population. The project also aims to contribute to the growing body of research on the application of machine learning in healthcare, particularly in the context of thyroid disease classification.

1.5 Features

The key features of this project include:

- Use of Classification algorithms for thyroid disease classification.
- Comprehensive data analysis and preprocessing.
- Development and rigorous evaluation of the models.
- Performance evaluation of the developed model using appropriate metrics.
- Comparison of the models with each other.
- Broad applicability due to the use of a non-region-specific dataset.

1.6 System Requirements

This project needs certain hardware and software requirements in order to be developed and run. These requirements are discussed below:

1.6.1 Development Requirements

Table 1.1: Table 1 Development Requirements

Hardware Requirements	Software Requirements
Personal Computer/laptop with the specifications: -RAM: 8GB(minimum) -Processor: CPU with four or more threads -512GB HDD or SSD(recommended) -GPU: recommended if possible	OS:windows 7 and above. -Python -Jupyter notebook -ML framework: Scikit-learn, Numpy, Pandas, Matplotlib, seaborn, imblearn and so on.

1.6.2 Deployment Requirements

Table 1.2: Table 2 Deployment Requirements

Hardware Requirements	Software Requirements
Personal Computer/laptop with the specifications: -RAM: 8GB(minimum) -Processor: CPU with four or more threads -Storage: 512GB HDD or SSD(recommended) -GPU: recommended if possible	OS: windows 7 and above. -Visual Studio Code -Streamlit -Pickle -Python

1.7 Feasibility Study

1.7.1 Economic Feasibility

This system is economically feasible which consists of a laptop or a personal computer without any expense of other items. Nowadays, almost every house has access to the internet, so our system is economically feasible.

1.7.2 Operational Feasibility

For the operation of the system, the person doesn't need to be an expert in using a computer. Someone with minimum knowledge about computers and technology can also benefit from the system. There is no requirement for huge and expensive hardware.

1.7.3 Schedule Feasibility

Since the development team has the capacity and basic understanding of the project, the project was completed in 7 month time period.

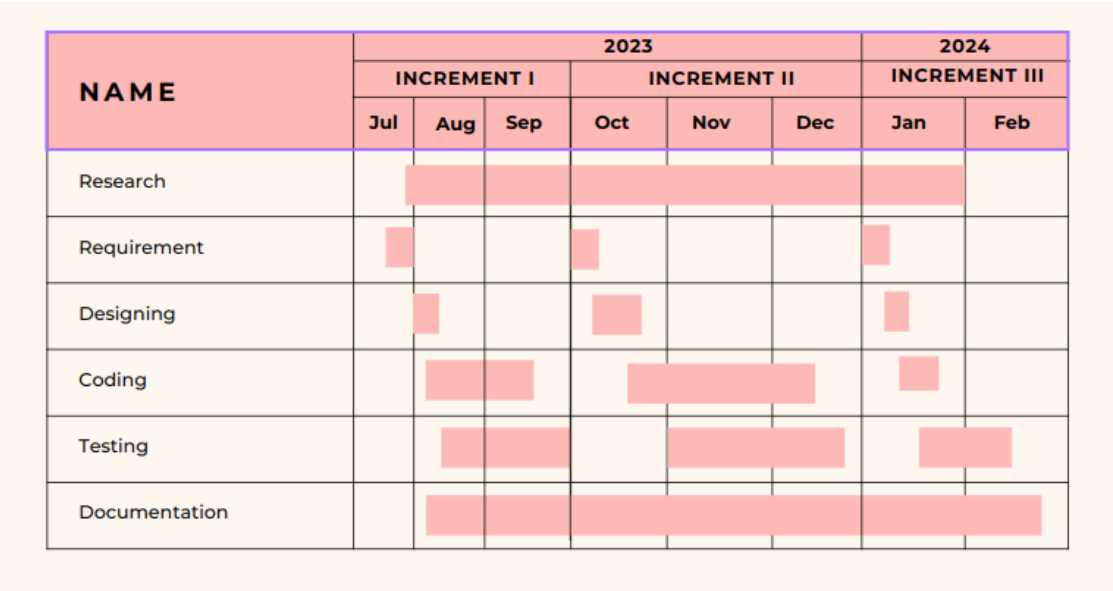

Figure 1.2: Gantt Chart

CHAPTER 2
LITREATURE REVIEW

2.1 Related Research

According to the paper "Thyroid function", Iodine is most important as a component of the hormones, thyroxine and 3,3,5-triiodothyronine (T3). The recommended daily iodine requirement is 150-200μg. Since iodine is a crucial constituent of thyroid hormones, it is not surprising that thyroid dysfunction is very common in geographical areas of iodine deficiency. However, even when this trace element is present in adequate supply, thyroid disease is present in 3-5% of the population. Furthermore, the regulated supply of thyroid hormone to specific tissues is crucial during fetal development [2].

Paper "Thyroid Disease Classification Using Machine Learning Algorithms" aims to classify thyroid conditions because medical reports show serious imbalances in thyroid diseases. The data was applied to a range of machine learning algorithms (Decision Tree, SVM, Random Forest, Naive Bayes, Logistic Regression, Linear Discriminant Analysis, k-Nearest neighbors, Multi-Layer Perceptron). On using all the attributes the results were as: Decision Tree 90.13 accuracy, SVM 92.53 accuracy, Random Forest 91.2 accuracy, Naive Bayes 90.67 accuracy, Logistic Regression 91.73 accuracy, Linear Discriminant Analysis 83.2 accuracy, KNeighborsClassifier 91.47 accuracy and MLP 96.4 accuracy. In the second step, 3 traits were removed, the deleted attributes were query_thyroxine, query_hypothyorid & query_hyperthyroid. The algorithms' performance after this were:Decision Tree 98.4 accuracy, SVM 92.27 accuracy, Random Forest 98.93 accuracy, Naive Bayes 81.33 accuracy, Logistic Regression 91.47 accuracy, Linear Discriminant Analysis 83.2 accuracy, KNeighborsClassifier 90.93 accuracy and MLP 97.6 accuracy. [3].

Paper " Feature selection, L1 vs. L2 regularization, and rotational invariance" Focused on logistic regression, showing that using L1 regularization of the parameters, the sample complexity (i.e., the number of training examples required to learn "well,") grows only logarithmically in the number of irrelevant features. This logarithmic rate matches

the best known bounds for feature selection, and indicates that L1 regularized logistic regression can be effective even if there are exponentially many irrelevant features as there are training examples. , L1 regularization, uses a penalty term which encourages the sum of the absolute values of the parameters to be small. L1 regularization in many models causes many parameters to equal zero, so that the parameter vector is sparse. This makes it a natural candidate in feature selection settings, where many features should be ignored[4].

Paper "Prediction using step-wise L1, L2 regularization and feature selection for small data sets with large number of features", applied L1 regularization to select the most relevant features. The initial set of more than 6,000 features was reduced to about 50, are filtering out irrelevant and redundant features[5].

Crucially, in evaluating classification models, the choice of metrics, particularly in the context of imbalanced datasets like those in thyroid disease classification, becomes paramount. The article "The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets" emphasizes this. It points out that PRC plots are more informative than ROC plots in such situations, as they focus on the minority class, providing a clearer picture of the classifier's performance in identifying less prevalent but clinically significant cases. This insight is vital for our project's methodology and aligns with our objective to enhance diagnostic accuracy in thyroid disease [6].

Paper "Recall-based Machine Learning Approach for early detection of Cervical Cancer", stresses that the recall value obtained should be imposed over accuracy because it involves wrong predictions as well which are of no significance. This left them with precision and which would also be outweighed because the former itself involves actual cervical cancer patients clearing the idea that no actual positive case with negative prediction should be missed over and actual negative case with positive prediction. Thus, higher the recall value for health disease classification, higher the model specificity[7].

CHAPTER 3
METHODOLOGY

3.1 Working Mechanism

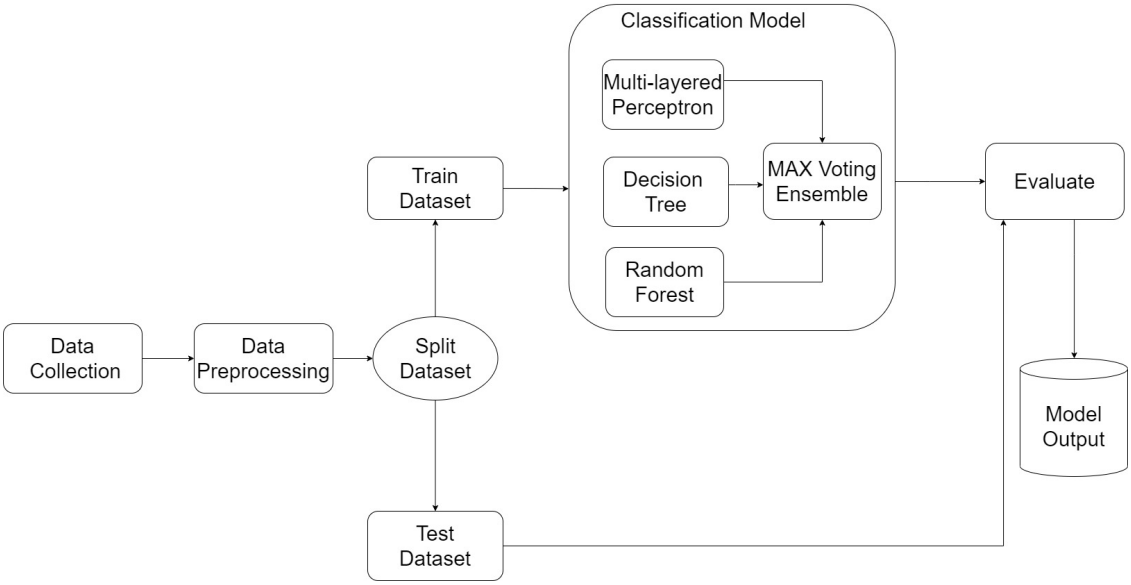

Figure 3.1: Block diagram of Thyroid Classification

The proposed structure of the project includes different steps. Firstly, thyroid dataset is collected from the Kaggle Website. The dataset has some null values and requires some preprocessing accordingly. After that the dataset is divided into training and testing dataset, training data is used to train the classification models and testing data is used to test and evaluate the perfomance of the system.

3.1.1 Data Collection

Machine learning algorithms are used in the rapid and early diagnosis of thyroid diseases and other diseases, as they are now in a significant position in the health field and help us in diagnosing and classifying diseases and for this reason we have collected our dataset that was found on Kaggle.The data that we have used in our study is a set of data taken from external hospitals and laboratories specialized in analyzing and diagnosing the thyroid diseases. In this dataset, we have found 9172 observations along with 31 attributes.

3.1.2 Data Preprocessing

The process of pre-processing the data is very important, as good data is crucial for good performance of the models. The pre-processing process is used to reveal the data, as it examines the data with great care. The pre-processing process includes data transformation, cleaning the data, feature selection and data balancing.

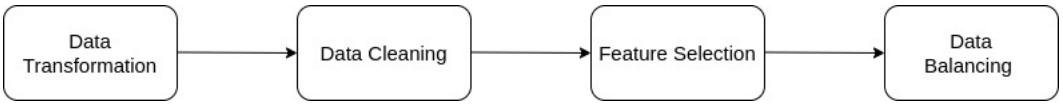

Figure 3.2: Block diagram of Data Preprocessing

In order to, classify among hypothyroid, hyperthyroid and no condition, sub-diagnosis class labels of hypothyroid and hyperthyroid are transformed as 'hypothyroid' and 'hyperthyroid' respectively. The dataset had considerable missing values which are addressed to clean the data. Furthermore, dimensional reduction is achieved by using L1 regularization. In this feature selection, linear model is penalised with the L1 norm and SelectFromModel, a meta transformer, alongside the estimater model that assigned importance to each feature is used to select the non-zero coefficients. Machine Learning models can only work with numerical values. For this reason feature encoding is done to transform the categorical values into numerical ones. Finally, to avoid favouring the majority class i.e. no condition by the machine learning models, SMOTE oversampling technique is performed. This balancing technique is ,however, performed to the training dataset only after train-test split on original dataset in order to avoid data leakage.

3.1.3 Machine Learning Technique

The key aim of using machine learning algorithms is to differentiate between three forms of thyroid disease. The first is hyperthyroidism, the second is hypothyroidism, and the third is stable patients who do not have any thyroid issues. In order to facilitate this, different Classification models will be implemented. This is achieved through scikit-learn, an open source machine learning library that also provides various tools for model fitting, data preprocessing, model evaluation, and many other utilities.

Decision Tree

A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks. It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. It is constructed by recursively splitting the training data into subsets based on the values of the attributes until a stopping criterion is met, such as the maximum depth of the tree or the minimum number of samples required to split a node[8].

- Impurity: A measurement of the target variable's homogeneity in a subset of data. It refers to the degree of randomness or uncertainty in a set of examples. The Gini index and entropy are two commonly used impurity measurements in decision trees for classification tasks.

- Information Gain: Information gain is a measure of the reduction in impurity achieved by splitting a dataset on a particular feature in a decision tree. The splitting criterion is determined by the feature that offers the greatest information gain, It is used to determine the most informative feature to split on at each node of the tree, with the goal of creating pure subsets.

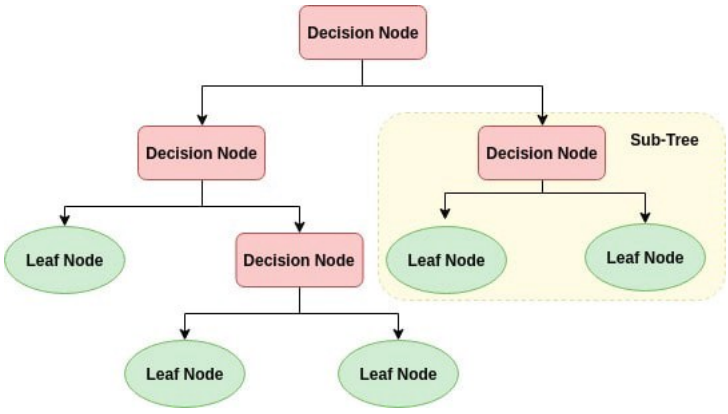

Figure 3.3: Decision Tree
Source:
https://www.numpyninja.com/post/decision-trees-example-in-machine-learning

- Entropy:
  - Entropy is the measure of the degree of randomness or uncertainty in the dataset.

In the case of classifications, it measures the randomness based on the distribution of class labels in the dataset.

$$\textit{Entropy} = \Sigma - (P_i \cdot \log_2 P_i) \tag{3.1}$$

Where, P_i = Probability of Class

$$\textit{Information Gain} = E(\text{Parent node}) - \sum(W_i \cdot E(\text{Child node})) \tag{3.2}$$

Where, E = Entropy

W_i = Size of Child / Size of Parent

- Gini Impurity or index:

  ◦ Gini Impurity is a score that evaluates how accurate a split is among the classified groups. The Gini Impurity evaluates a score in the range between 0 and 1, where 0 is when all observations belong to one class, and 1 is a random distribution of the elements within classes. In this case, we want to have a Gini index score as low as possible. Gini Index is the evaluation metric we shall use to evaluate our Decision Tree Model.

$$\textit{Gini index} = 1 - \sum P_i^2 \tag{3.3}$$

$$\textit{Information Gain} = G(\text{Parent node}) - \sum(W_i \cdot G(\text{Child node})) \tag{3.4}$$

Where, G = Gini index

W_i = Size of Child / Size of Parent

The decision tree model is implemented by importing DecisionTreeClassifier from scikit-learn's tree module. The decision tree model is then trained using the training data. Other utility modules of scikit-learn like metrics is used to evaluate this models performance on test data. Such metrics used to assess the diagnostic performance are Precision-Recall curve, confusion matrix and learning curve. Furthermore, minimal cost complexity pruning is performed by choosing cross validated complexity parameter.

Random Forest

Random forest is a supervised machine learning algorithm that can be used for solving classification and regression problems both. However, mostly it is preferred for classification. It is named as a random forest because it combines multiple decision trees to create a "forest" and feed random features to them from the provided dataset. Instead of depending on an individual decision tree, the random forest takes prediction from all the trees and selects the best outcome through the voting process[9]. Now, the question arises why do we prefer random forests over decision trees. So, individual trees are more prone to overfitting but random forests can reduce this problem by averaging the predicted results from each tree.

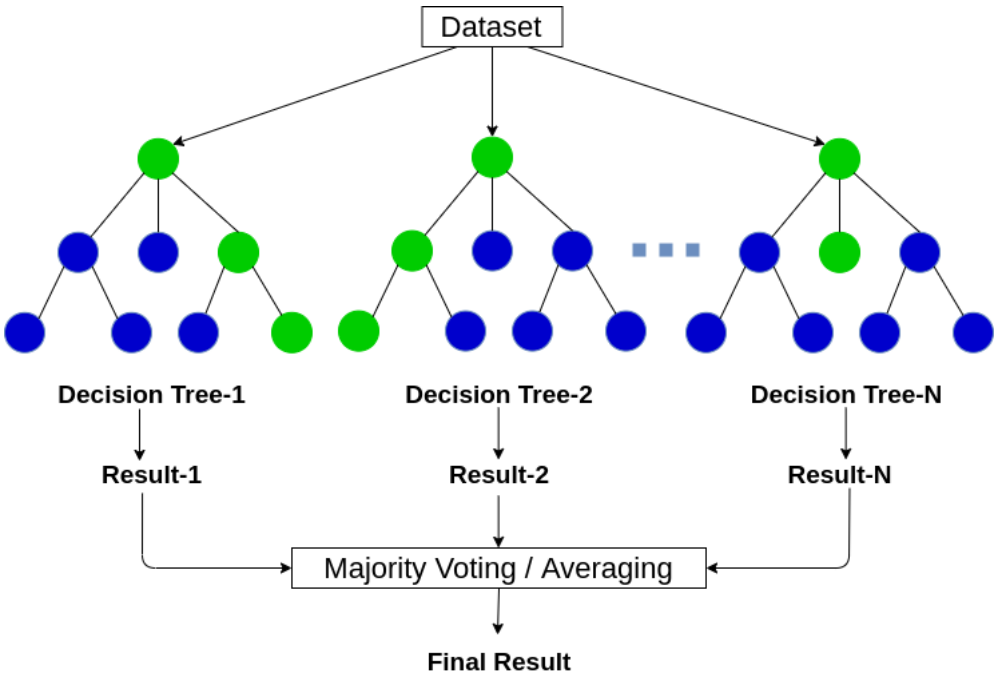

Figure 3.4: Working diagram of Random Forest
Source:https://datamahadev.com/random-forests-in-machine-learning-a-detailed-explanation/

Random Forest is implemented by importing RandomForestClassifier from ensembled module of scikit-learn. The model is initialized with 50 base estimators and trained on the training dataset. Like decision tree, minimal cost complexity for random forest is performed. Likewise, the model is evaluated to diagnose its performance.

Support Vector Machine(SVM)

Support Vector Machine(SVM) constructs a Hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks.The Support Vector Machine (SVM) stands as a supervised machine learning algorithm with applications in both classification and regression tasks. While SVM is versatile enough to handle regression problems, its optimal utility lies in the domain of classification. The central objective of the SVM algorithm involves the identification of a hyperplane within an N-dimensional space that distinctly segregates the data points. The dimensionality of this hyperplane is contingent upon the quantity of input features. In scenarios where the input features number two, the hyperplane is represented as a line. For instances with three input features, the hyperplane assumes the form of a 2-dimensional plane. However, the visualization of hyperplanes becomes progressively intricate as the number of features surpasses three[10].

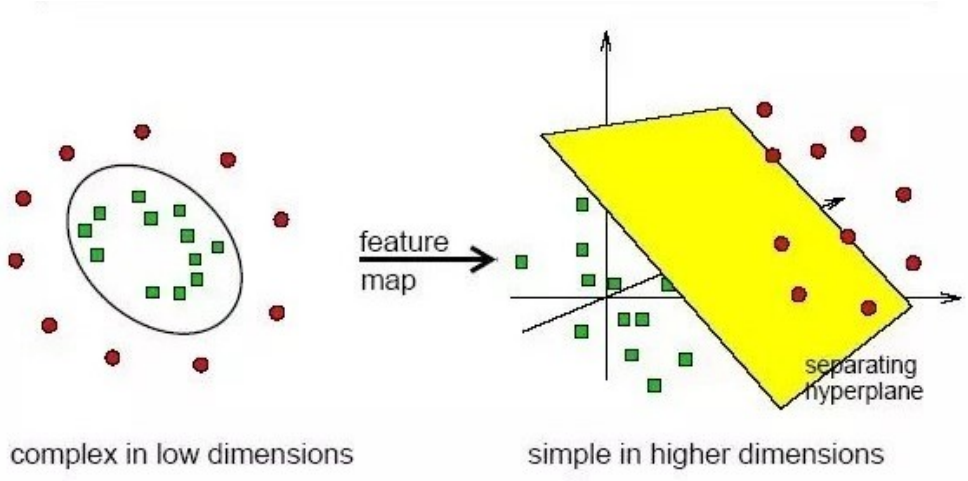

Figure 3.5: SVM
Source:https://www.dtreg.com/solution/support-vector-machines

Multi-Layer Perceptron(MLP)

Multi-Layer Perceptrons, a fundamental type of neural network, are vital tools in supervised learning, predominantly used for complex pattern recognition, classification, and regression tasks. An MLP consists of multiple layers: an input layer, one or more hidden layers, and an output layer. Each layer is made up of nodes, or neurons, which are interconnected and apply activation functions to their inputs. Initially, the weights

of the connections between neurons are randomly assigned. Each connection between neurons has an associated weight, which determines the strength of the connection. Additionally, each neuron has a bias, which helps in adjusting the output.he information flows through the network in a forward direction during the training process. The input data is fed into the input layer, and the output is obtained from the output layer. The computations are performed layer by layer, using the weights and biases.

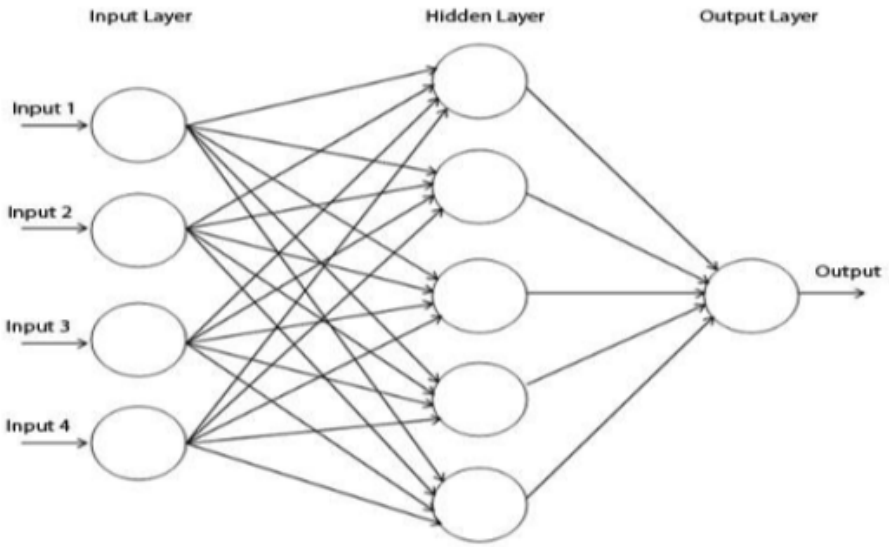

Figure 3.6: MLP structure
Source: https://www.researchgate.net/publication/329277388 _Application_of_Multilayer_Perceptron_MLP_for_Data_Mining_in_Healthcare_Operations

If a multilayer perceptron has a linear activation function in all neurons, that is, a linear function that maps the weighted inputs to the output of each neuron, then linear algebra shows that any number of layers can be reduced to a two-layer input-output model. In MLPs some neurons use a nonlinear activation function.The ReLU activation function is widely used in neural networks due to its simplicity and effectiveness. It is defined as:

$f(x) = max(0, x)$

The ReLU function returns the input x if it is positive, and zero otherwise. It introduces non-linearity to the model, allowing it to learn and represent complex patterns. Each connection between neurons has an associated weight, which determines the strength of the connection.

The difference between the predicted output and the actual target is quantified using a loss function; log-loss in case of classification. The goal during training is to minimize

this loss. During training, the network learns by adjusting its weights and biases to minimize the difference between its predicted output and the actual target values. This process is known as backpropagation. Gradient descent is often used as the optimization algorithm to update the weights and biases in the direction that minimizes the error.
$NewWeight = OldWeight - LearningRate * Gradient$
Training is typically done over multiple passes through the entire dataset, known as epochs[11].

MLP is implemented by importing MLPClassifier from neural_network module in scikit-learn. The model is initialized by allowing 500 maximum iterations followed by strength of the L2 regularization term of 0.076, and trained on the training dataset. The model is evaluated in the similar fashion as previous model, meanwhile the learning curve for MLP depicts the loss function at each epoch. This shows how the optimization in MLP is iterating until convergence.

3.1.4 Split Dataset

An important phase in the model development process is splitting the facts for system mastery models. It entails breaking up the available dataset into training and testing datasets. The testing out set is used to evaluate the model's overall performance while the training set is used to teach the model. The typical cut up is 20–30% for testing out and 70–80% for training, but this might change based on the size of the dataset and the specific use case. The testing set provides an objective assessment of the generalizability of our model. We assess our model's performance on the testing set once it has been fully trained and tuned using the training and validation sets. This phase accurately predicts how well our model will perform on new thyroid patient samples that it hasn't seen before.

3.1.5 Train Model

Training a machine learning model involves the process of feeding data to the model, adjusting its internal parameters, and optimizing its performance. In the context of thyroid patient classification, the model will learn from the input features and their

corresponding labels during this process. The specific training algorithm and duration depend on the chosen model and complexity of the data.

3.1.6 Model validation

Model validation is the task of evaluating whether a chosen model is appropriate or not. The significance of a correct medical diagnosis necessitates specific considerations when validating a machine learning model for thyroid illness categorization.

3.1.7 Model Output

We use streamlit which allows us to display descriptive text, model outputs, and enter new inputs through the User-Interface for classification of thyroid. Inorder to save the model trained, module called pickle is used, which can be defined as a module in Python used for serializing and de-serializing Python objects. This converts Python objects like lists, dictionaries, etc. into byte streams (zeros & ones).

3.2 Development Model

The Incremental model, which is a mixture of waterfall and iterative development approach is opted, as the project demands few requirements and is fairly simple. As the name suggests, the major objective is focused as the first increment of the project. The activities in this model are completed iteratively and each outcome acts as an input for the next phase, thus increments are developed in such a way.

3.2.1 Requirement Specification

The final product can be divided into two parts: the frontend and the backend. The frontend shall be a simple page where users can input a few specific hormones (assuming the user has measured these hormones prior to using the product), and obtain their thyroid condition. Whereas, the backend consists of training our models evaluating them before exporting as a pickle file.

3.2.2 Development and Implementation

The project has the architecture close to any other ML projects, i.e. EDA –> pre-processing –> model training –> save model. The product is written using python programming language and jupyter notebook. The interface for the use of the product is rather simple as the only requirement is classification of thyroid, which is achieved through the use of the streamlit framework.

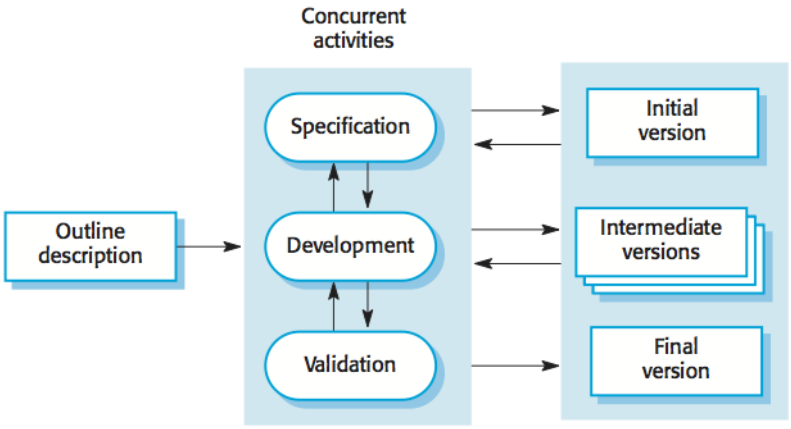

Figure 3.7: Incremental Development Model
Source:
https://copyprogramming.com/howto/the-best-sdlc-model-to-deploy-in-software-dev

3.2.3 Verification and Validation

In order to approve that the product meets its defined objectives, verification and validation is exercised. This involves Meetings and Code reviews. However, considering the imbalance of data in our thyroid dataset there are chances our model overfitting. Thus, we implemented balancing methods to handle the imbalanced data. The code itself is verified with inspections and walkthroughs among the group and even by external parties. Finally, the product is verified by testing its classification prediction on the testing split of dataset.

3.2.4 Increments

As discussed earlier the project went through several increments. The first increment solely focuses on the major requirement, developing models for the required classifica-

tion. The further increments are focused on evaluation of the models, exporting the best model, developing a front-end for the product.

3.3 Evaluation Criteria

Upon the successful construction of a system model, it undergoes training using the training dataset sourced from the standardized dataset.The effectiveness of the system's performance is assessed by Accuracy, Precision, Recall, and F1 score .

Accuracy:

The ratio of true positives and true negatives to all positive and negative observations is the definition of model accuracy, a performance statistic for machine learning models.

$Accuracy = \frac{TP+TN}{TP+FN+TN+FP}$

Precision:

The percentage of labels that were correctly predicted positively is represented by the model precision score.

$Precision = \frac{TP}{TP+FP}$

Recall:

The model's ability to properly forecast the positive out of actual positives is measured by the model recall score.

$Recall = \frac{TP}{TP+FN}$

F1 Score:

The F1 score combines precision and recall using their harmonic mean, and maximizing the F1 score implies simultaneously maximizing both precision and recall.

$F1Score = \frac{2*Precision*Recall}{Precision+Recall}$

3.4 System Diagrams

Thyroid Detection System prompts the user to enter different data from their medical test report such as age, sex, T3, T4, TSH values for the system to classify whether they are suffering from hypothyroidism/hyperthyroidism or not. Use of different maching learning models such as Decision Tree, Random Forest, MLP aids in thyroid classfica-

tion so the user has better understanding of their thyroid health. If the user doesnot have all the required data from the medical test report, the system prompts the user to enter their details for their thyroid classification.

3.4.1 Usecase Diagram

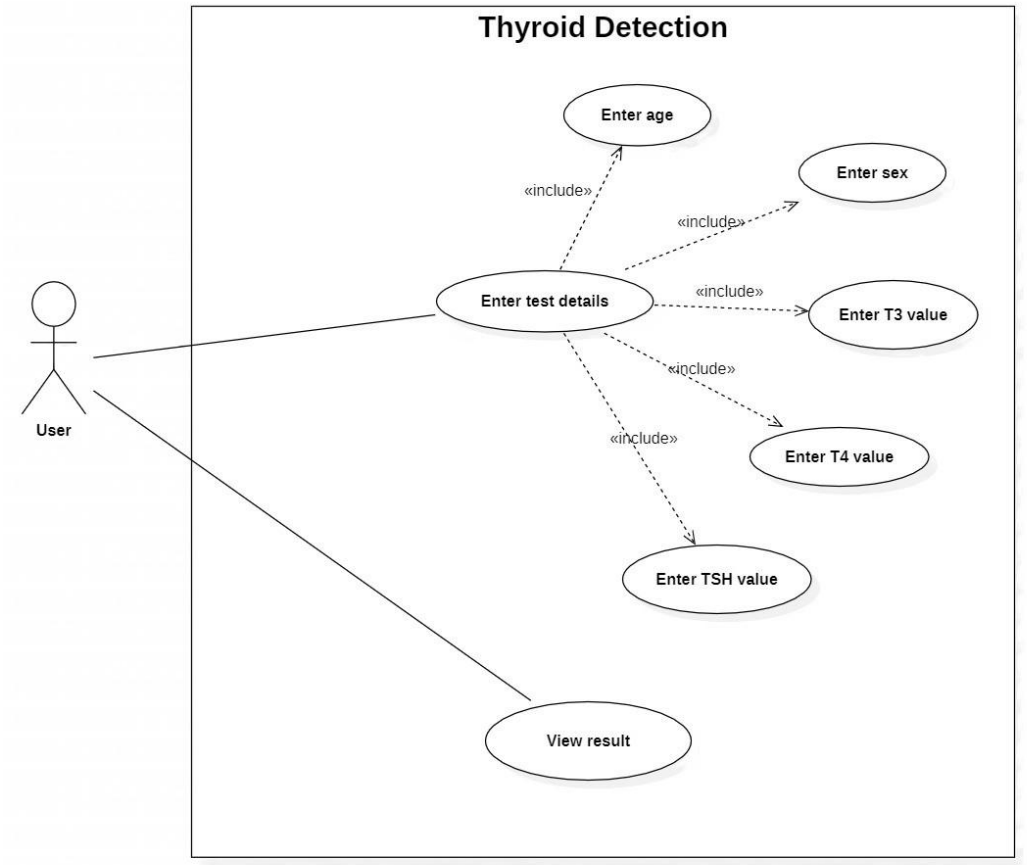

Figure 3.8: Usecase Diagram

3.4.2 DFD

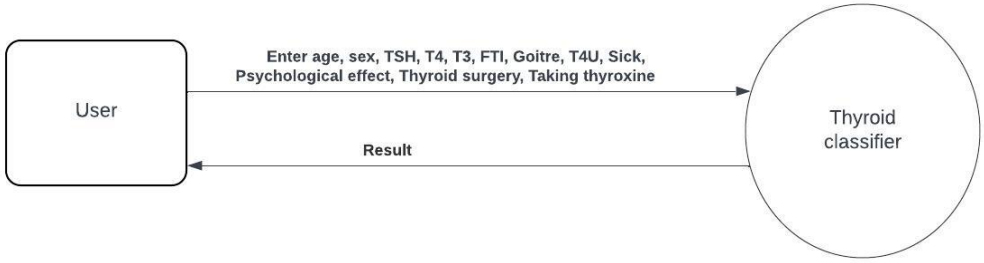

Figure 3.9: DFD diagram

3.4.3 Class Diagram

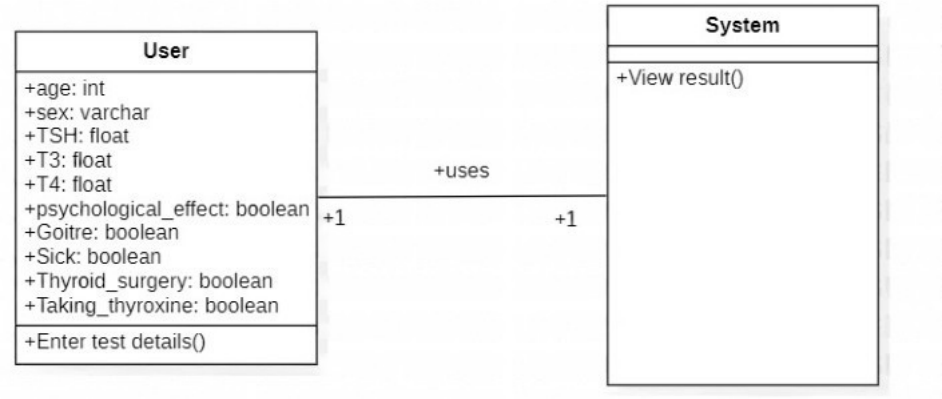

Figure 3.10: Class diagram

3.4.4 Activity Diagram

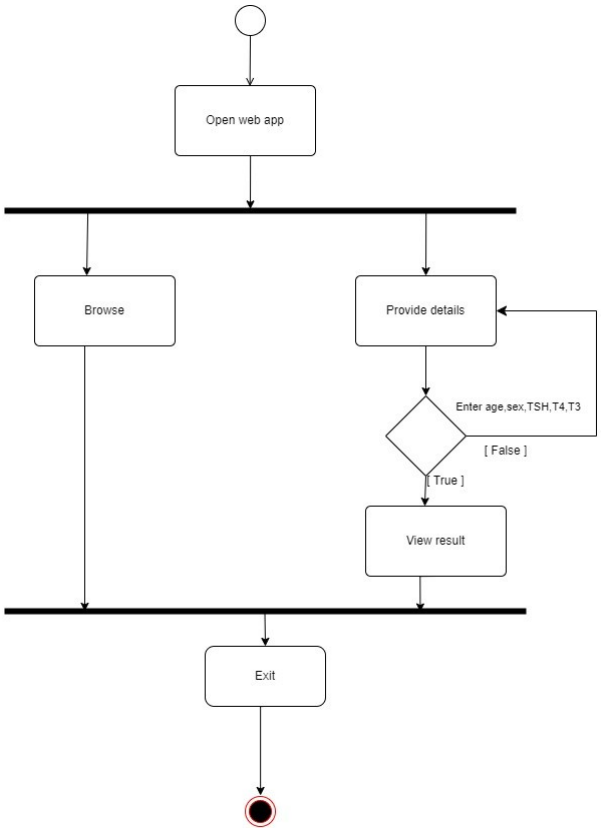

Figure 3.11: Activity Diagram

CHAPTER 4
RESULT AND DISCUSSION

4.1 Preprocessing

As described in the proposed methodology, the feature selection and feature preprocessing yield the balanced thyroid disease classification dataset. The majority of the classification count is categorized as "no condition". The "no condition" means that the data sample is not categorized as any other classes like hyperthyroid, hypothyoid, binding proteins, general health, replacement therapy, antithyroid treatment, or miscellaneous meaning the subject is healthy with no thyroid conditions. For the condition hyperthyroid, the diagnosis classes were hyperthyroid(A), T3 toxic(B), toxic goitre(C), secondary toxic(D) and their count was 147, 21, 6 and 8 respectively. The low counts of this diagnosis class were remapped into hyperthyroid class which would make easier for thyroid classification. Similarly, the same was performed for hypothyroid condition where the subclasses were remapped into hypothyroid.

The diagnosis classes for binding protein were increased and decreased binding protein whose count were 346 and 30 respectively. The count for the general health condition was 436. The diagnosis classes for replacement therapy were underreplaced, consistent with replacement therapy, and overreplaced whose counts were 111, 115 and 110 respectively. The diagnosis classes for antithyroid treatment condition were antithyroid drugs, I131 treatment and surgery with counts 14, 5 and 14 respectively. The miscellaneous condition and its classes with count were; discordant assay result - 196, elevated TBG - 85, elevated thyroid hormones - 0. For no thyroid condition the count was 6771. The dataset consists of 9173 patient records out of which 6771 are normal patient records, and notable conditions include primary hyperthyroid 233 and compensated hypothyroid with 359 patients. Since the number of samples for other classes were not significant, we remapped the dataset and selected only hyperthyroid, hypothyroid and no thyroid conditions while other classes with low instances were not considered. Thus, we have a dataset of 7542 instances with 593 hypothyroid cases and 182 hyperthyroid cases.

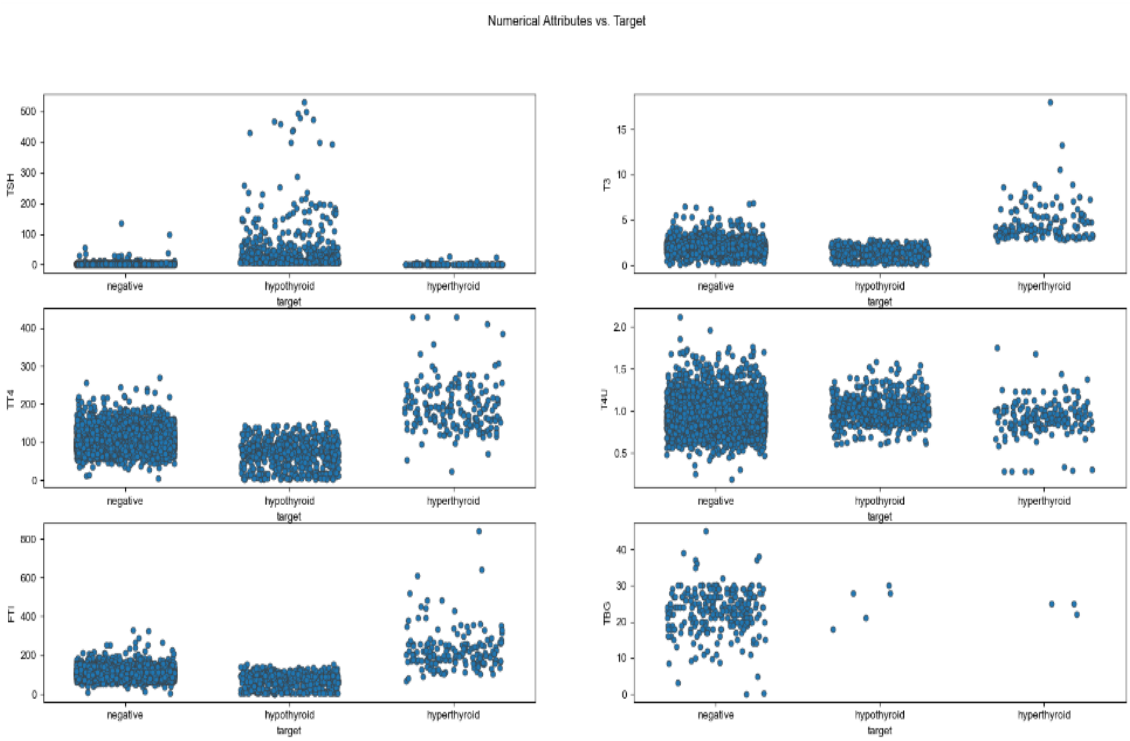

Figure 4.1: Strip plot: Numerical Attributes vs Target

On observing the above strip plot, the attributes TSH, TT4, T3, FTI can be good candidate to effectively classify thyroidism.Furthermore, the attributes TBG has less instances which we addressed in data cleaning.

Data Cleaning

The dataset has tons of missing values so we analyzed all the missing values from the dataset. 7287 instances of TBG, about 96.5% of total were found to be missing, so we dropped the TBG attribute as a whole. Also for T3, out of 2209 instances, 29.2% of data was found to be missing but we had observed that T3 is an important attribute for classification so we didnot completely drop it and performed median imputation. Same imputation was done for other missing features. After this our dataset is clean with no missing/null-values.

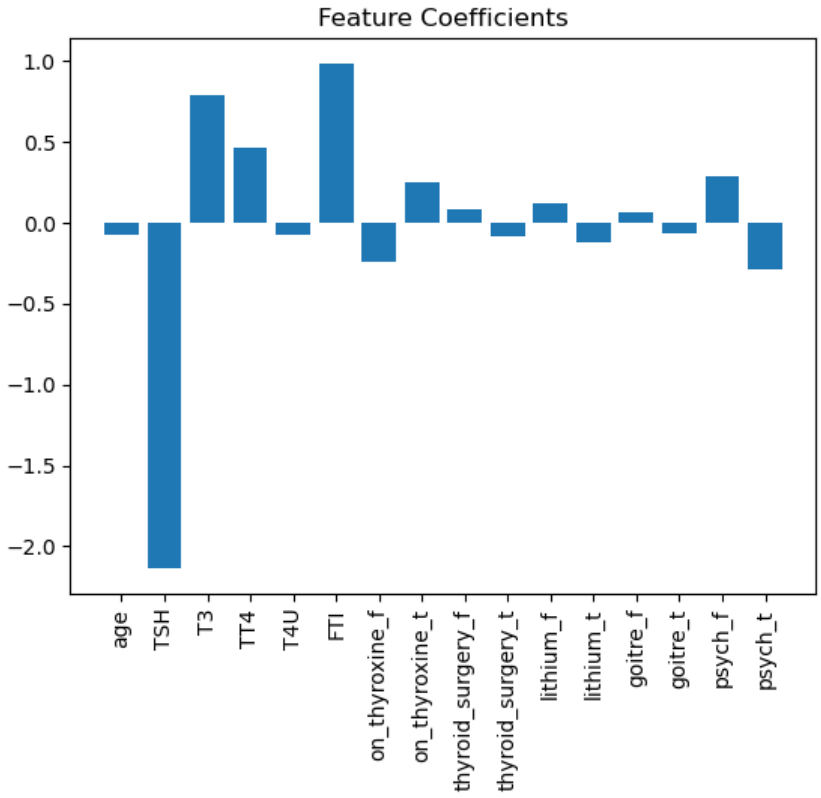

Figure 4.2: Feature Selection

Feature Selection

L1 regularization was used for feature selection. L1 regularization is commonly used in machine learning to enhance model generalization and perform feature selection by penalizing the absolute values of the coefficients in logistic regression. The loss function with L1 regularization is modified by adding the L1 norm of the coefficients(weights) to the standard loss term:

$$J(\theta) = \text{Loss}(\theta) + \lambda \sum_{i=1}^{n} |\theta_i| \tag{4.1}$$

Here, $J(\theta)$ is the regularized loss funtion. $Loss(\theta)$ is the original loss(e.g., categorical cross-entropy loss for multi-class classification),λ is the regularization parameter, n is the number of features, and θ_i represents the coefficients.

The L1 regularization term, $\lambda \sum_{i=1}^{n} |\theta_i|$ introduces a penalty for having non-zero coefficients. As a result, during the optimization process, some of the coefficients are driven to exactly zero, effectively leading to feature selection[4]. After feature selec-

tion the relevant features that were selected are: age, sex, TSH, T3, TT4, T4U, FTI, On_Thyroxine, Thyroid_Surgery, sick, goitre, psych and Target.

Table 4.1: Thyroid disease attributes & dataset info

Attribute	Description	Domain of Value	Non-Null	DType
Age	Age in years	1 to 97	7542	Float64
Sex	Sex	Female(F) Male(M)	7542	object
TSH	Thyroid-stimulating hormone	0.01 to 530 μIU/ml	7542	Float64
T3	Triiodothyronine	0.05 to 18.00 ng/mL	7542	Float64
TT4	Total thyroxine	2.00 to 600.00 ng/mL	7542	Float64
T4U	Thyroxine Uptake	0.17 to 2.33	7542	Float64
FTI	Free Thyroxine Index	1.40 to 881.00	7542	Float64
On_Thyroxine	Taking thyroxine tablet	True(t) False(f)	7542	object
Thyroid_Surgery	Undergone thyroid related surgery	True(t) False(f)	7542	object
sick	Currently ill	True(t) False(f)	7542	object
goitre	Enlarged thyroid gland	True(t) False(f)	7542	object
psych	Thyroid related psychological symptoms	True(t) False(f)	7542	object
Target	Thyroid Disease	Hyperthyroid Hypothyroid Negative	7542	object

The categorical features are encoded as most machine learning models only accept numerical variables, thus preprocessing the categorical variables was a necessary step.

4.2 Model Training

Firstly, we trained Decision Tree with default parameters. Even though its predictions were optimistic for the testing dataset, it was clearly overfitting. The model was complex due to it trying to fit all the training dataset and couldn't generalize well for new datas.This was evident as the loss on test was high compared to train dataset.
The formula for categorical log loss is:

$$\text{CategoricalLogLoss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log(p_{ij}) \tag{4.2}$$

Where:
N is the number of samples or instances in the dataset.
M is the number of classes.
y_{ij} is a binary indicator (0 or 1) if class label *j* is the correct classification for instance *i*.
p_{ij} is the predicted probability that instance *i* belongs to class *j* [12].

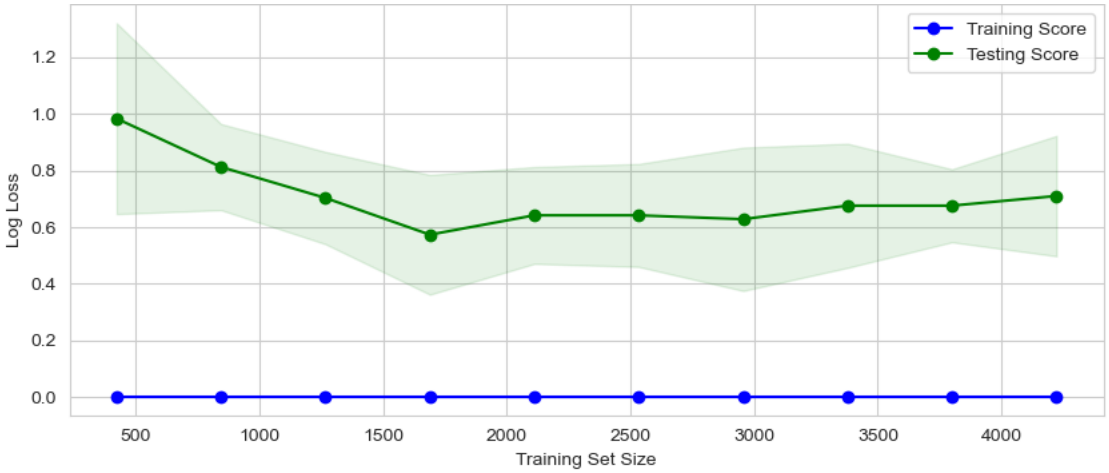

Figure 4.3: Overfitting Decision Tree

When it comes to medical datasets, it is normal that the datas on normal patients dramatically outweighs actual cases of illness. Same was the case for our thyroid dataset so oversampling technique namely SMOTE, was performed. Now, another decision tree model was trained on balanced dataset. This model was not complex thus could generalize to new unseen datas.

Other models for classifications were tested i.e. SVM, RF, and MLP. However, the

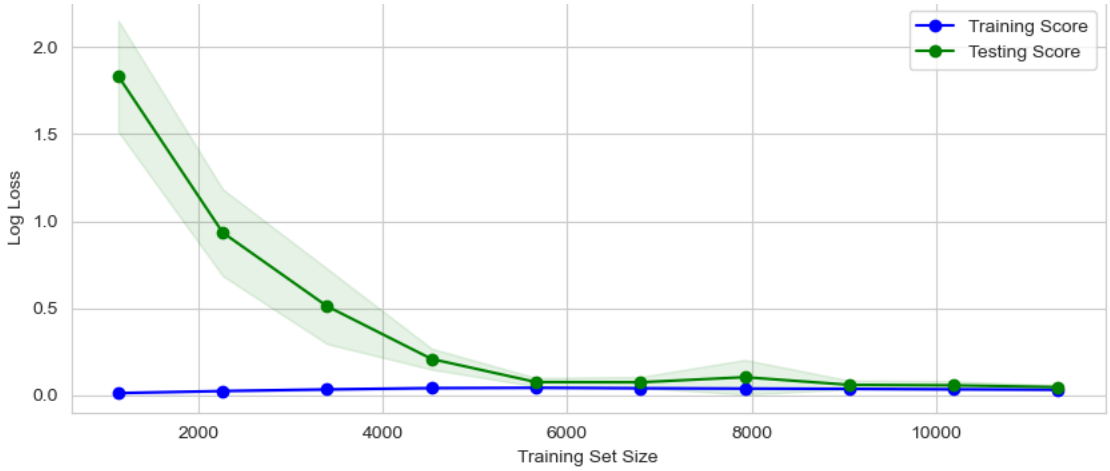

Figure 4.4: Decision Tree on Balanced Dataset

performance of SVM was subpar compared to other models evaluated. Remanining models, RF and MLP, worked well on the balanced dataset. Since, we need to classify thyroid conditions, we had to make sure the models are correctly classifying positive instances, Hypothyroid and hyperthyroid. Thus, Recall is the primary focus of criteria for our models(Low False-Negatives). Generally, recall improves at the expense of precision or precision improves at the expense of recall[13]. While, balancing the dataset, recall was improved while trading-off precision, we had to make sure the trade-off in precision is not that large(High False-Positives).

This low precision of SVM, compared to other models, was the sole reason to not

Table 4.2: Performance of the Models

	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)
Decision tree on imbalanced dataset	98.674	90.152	90.512	90.331
Decision tree on balanced dataset	98.188	85.168	96.620	89.793
SVM	92.930	69.039	91.292	74.946
RF	97.791	83.467	97.830	89.174
MLP	96.774	80.868	96.494	87.372
Ensembled Model	98.011	84.966	97.233	90.106

consider it for the final Ensembled model. Since three models are performing well, we employed Soft Max Voting ensemble model which can incorporate the predictions made by all three models to generate best possible result. This ensembled model has high recall score, while balancing out the precision-recall trade-off; it has the highest F1-score of 90.106%.

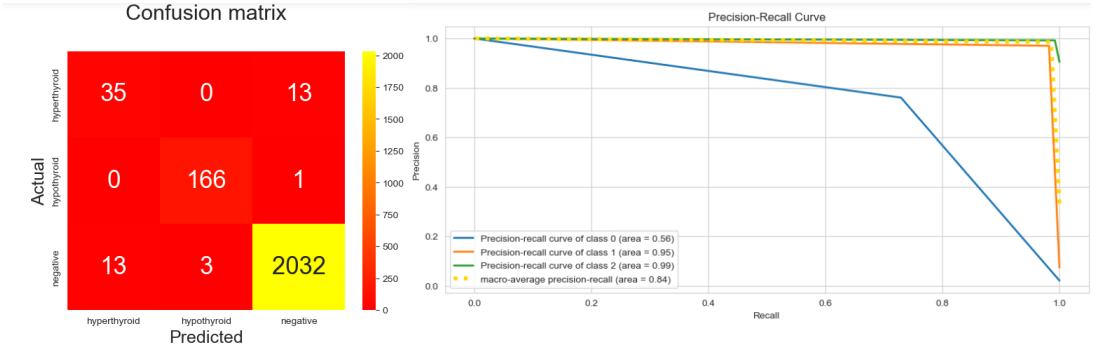

Figure 4.5: Evaluation of Unbalanced Decision Tree

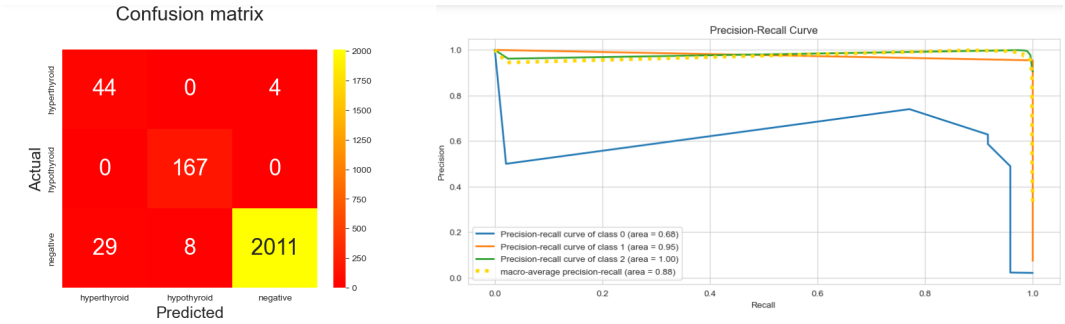

Figure 4.6: Evaluation of Balanced Decision Tree

After finalizing the ensemble model as our model to classify thyroids for early diagnosis, we exported this model as a pickle file. This model as a pickle file is made to predict the classifications of new datas entered by the final system's user. We prepared a simple and interactive web-app through the use of streamlit for users to interact with the system.

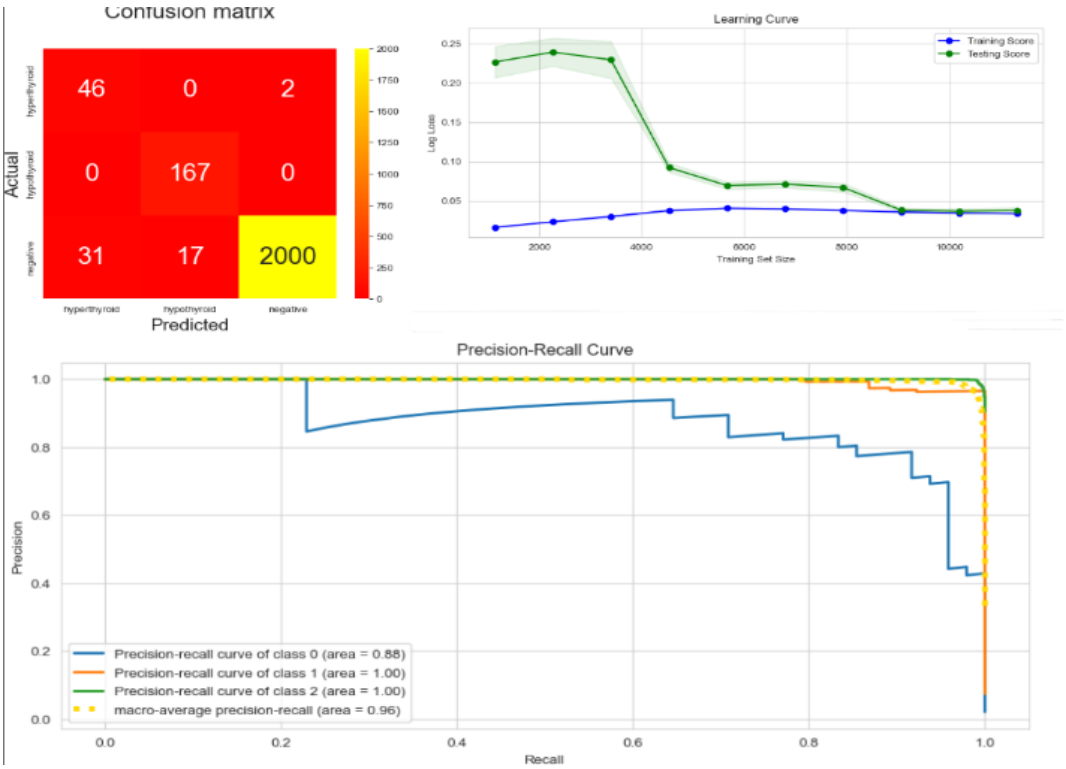

Figure 4.7: Evaluation of Random Forest

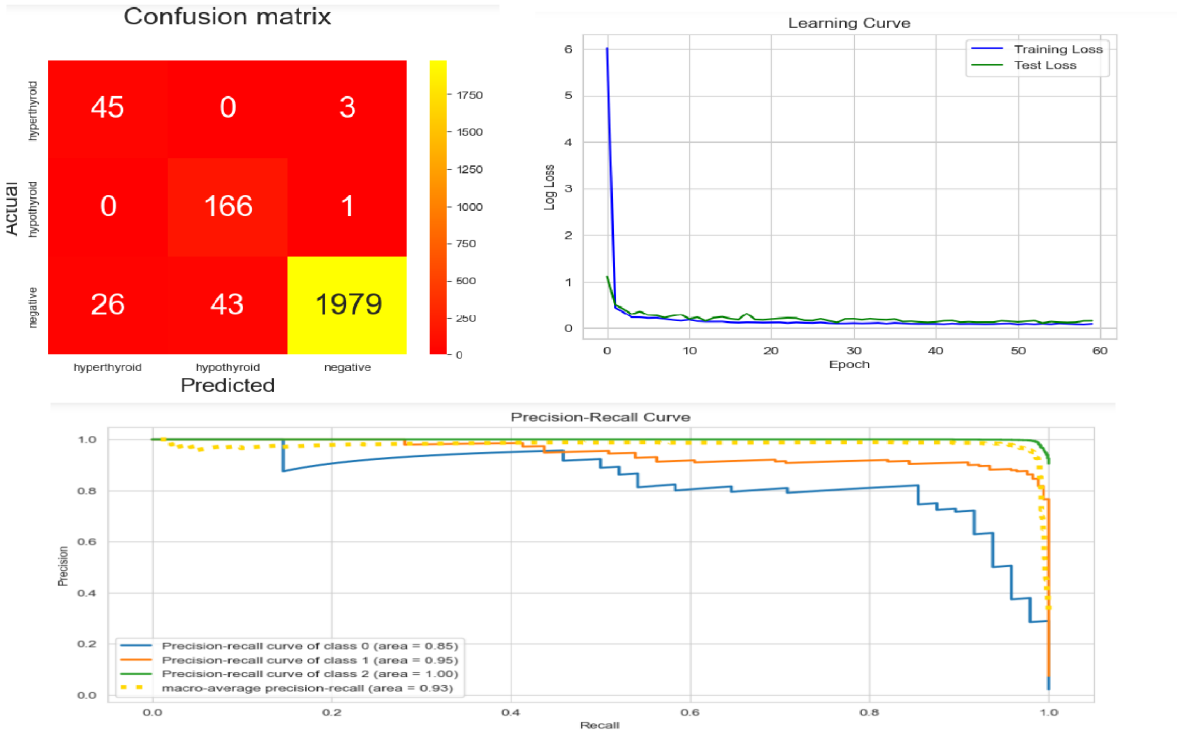

Figure 4.8: Evaluation of MLP

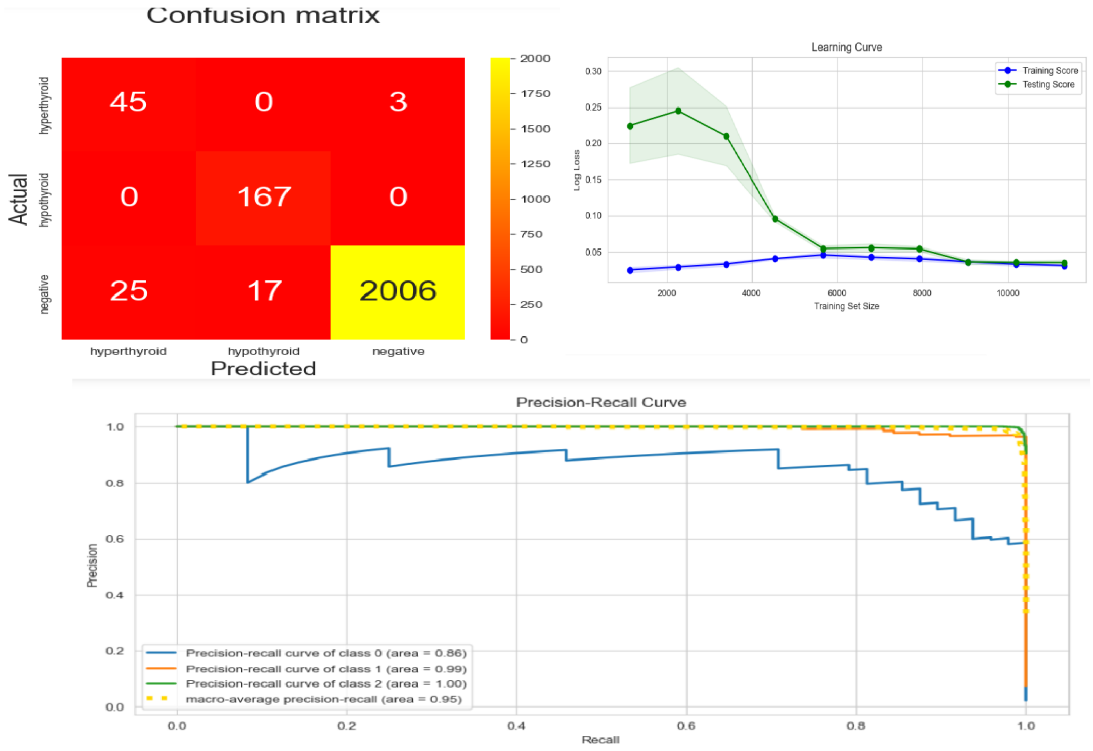

Figure 4.9: Evaluation of Ensemble

CHAPTER 5
CONCLUSION AND FUTURE ENHANCEMENTS

5.1 Conclusion

Since thyroid conditions are problem plaguing the world, thyroid conditions classification is a matter of great importance. The project focused on the classification of thyroid disorders, with specific emphasis on hyperthyroidism, hypothyroidism and no thyroid conditions. Through the use of machine learning models, the project aimed to accurately differentiate between these conditions based on relevant features extracted from thyroid function tests. The algorithms' recall is one of the key factors when evaluating our models, as it is concerned with the positive classes' classification, hypothyroidism and hyperthyroidism. Ultimately, our model ensembled from classification models showed promising result with high recall and balanced trade-off with precision. The development of such a classification system holds great potential for aiding clinicians in diagnosing thyroid disorders more efficiently and effectively, ultimately leading to better patient outcomes.

5.2 Future Enhancements

- Implement a mechanism for the system to continuously learn from new data and update its classification models over time, ensuring that it remains accurate and up-to-date with the latest medical knowledge.
- Implement features for long-term monitoring of patients with thyroid disorders, such as tracking changes in thyroid function over time and adjusting treatment plans accordingly.
- Develop tools or features to engage patients in managing their thyroid health, such as educational resources or reminders for follow-up appointments and medication adherence.

REFERENCES

[1] R. K. Yadav, "A prevalence of thyroid disorder in western part of nepal," vol. 7, Feb 2013. [Online]. Available: https://www.jcdr.net/article_fulltext.asp?issn=0973-709x&year=2013&volume=7&issue=2&page=193&issn=0973-709x&id=2724

[2] J. R. Arthur and G. J. Beckett, "Thyroid function," vol. 55, no. 3, p. 658–668, Sep 1999. [Online]. Available: https://doi.org/10.1258/0007142991902538

[3] K. salman and E. Sonuç, "Thyroid disease classification using machine learning algorithms," *Journal of Physics: Conference Series*, vol. 1963, no. 1, p. 012140, jul 2021. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1963/1/012140

[4] A. Ng, *Feature selection, L 1 vs. L 2 regularization, and rotational invariance*. [Online]. Available: https://ai.stanford.edu/~ang/papers/icml04-l1l2.pdf

[5] O. Demir-Kavuk, M. Kamada, T. Akutsu, and E.-W. Knapp, "Prediction using step-wise l1, l2 regularization and feature selection for small data sets with large number of features," *BMC Bioinformatics*, vol. 12, no. 1, Oct 2011.

[6] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," vol. 10, no. 3, p. 1–21, Mar 2015. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0118432

[7] A. Gupta, A. Anand, and Y. Hasija, "Recall-based machine learning approach for early detection of cervical cancer," in *2021 6th International Conference for Convergence in Technology (I2CT)*, 2021, pp. 1–5.

[8] GeeksForGeeks, "Decision tree - geeksforgeeks," Oct 2017. [Online]. Available: https://www.geeksforgeeks.org/decision-tree/

[9] S. Awasthi, "Random forests in machine learning: A detailed explanation," Dec 2020. [Online]. Available: https://datamahadev.com/random-forests-in-machine-learning-a-detailed-explanation/

[10] P. Chapagain, “Heart disease prediction using outlier removal based max voting ensemble method,” *International Journal on Engineering Technology*, vol. 1, no. 1, p. 83–101, Dec 2023.

[11] M. Amin and A. Ali, “Application of multilayer perceptron (mlp) for data mining in healthcare operations,” 02 2017.

[12] Apr 2021. [Online]. Available: https://neptune.ai/blog/cross-entropy-loss-and-its-applications-in-deep-learning

[13] M. Gordon and M. Kochen, “Recall-precision trade-off: A derivation,” *Journal of the American Society for Information Science*, vol. 40, no. 3, pp. 145–151, 1989.

[14] C. Goyal, “Why you shouldn’t just delete outliers,” May 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/why-you-shouldnt-just-delete-outliers/

ANNEX

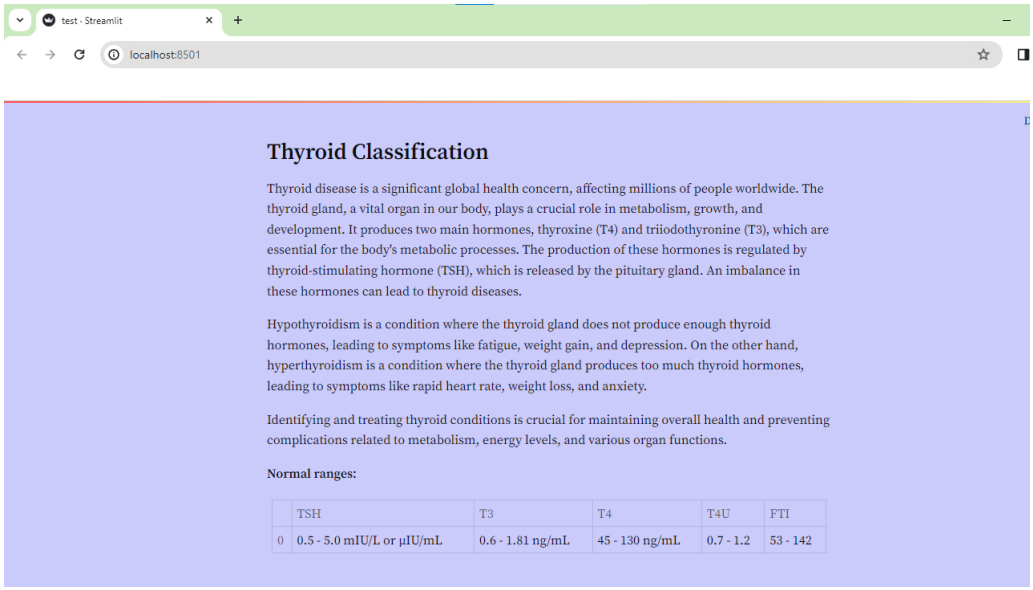

Figure 5.1: Result 1

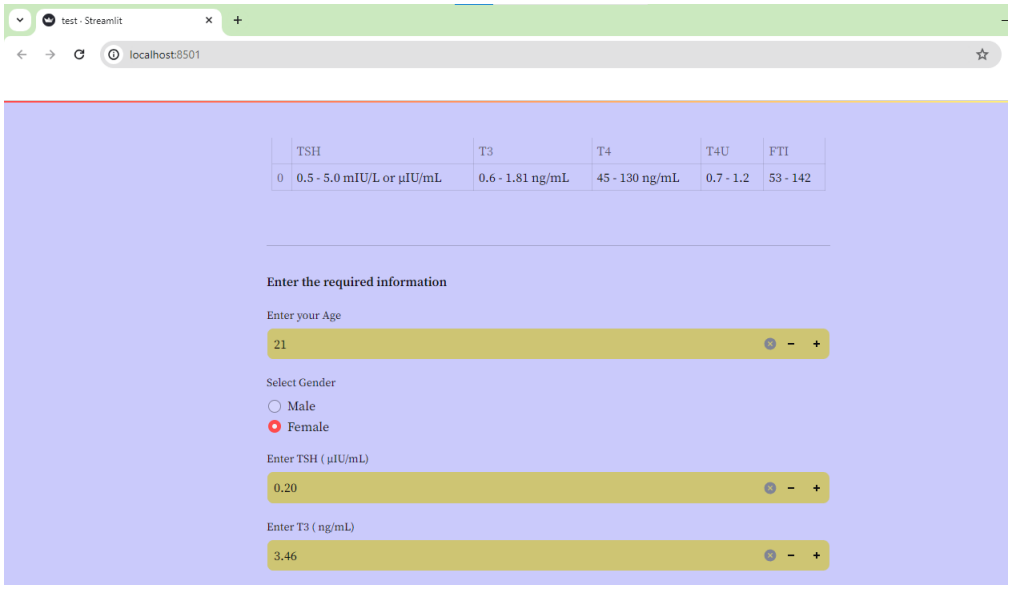

Figure 5.2: Result 2

Figure 5.3: Result 3

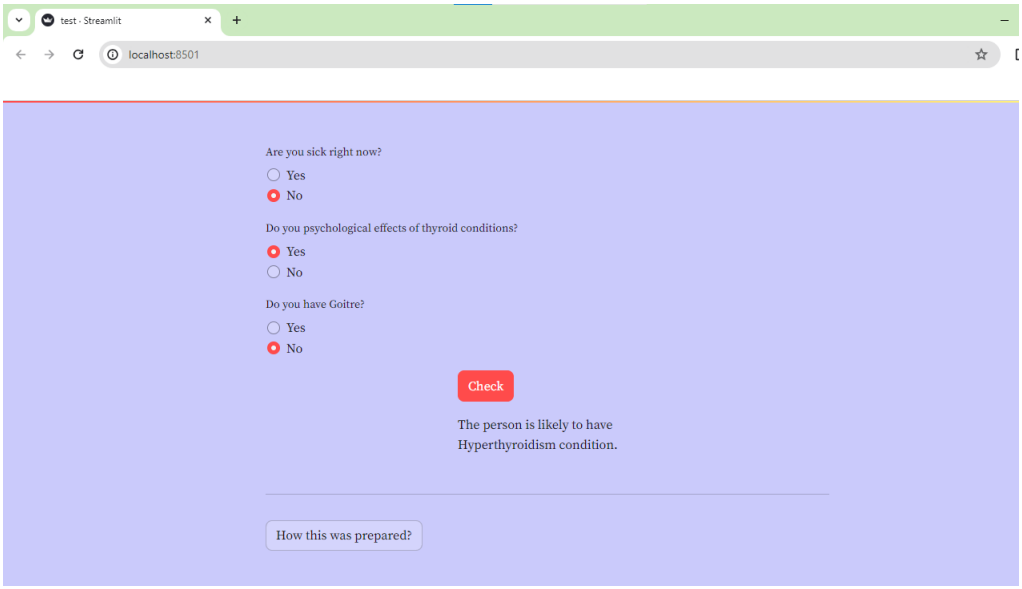

Figure 5.4: Result 4