School of Computing, Engineering and Digital Technologies
Department of Data Science
Teesside University
Middlesbrough TS1 3BA

Advanced Classification of Thyroid Diseases Using Machine Learning Algorithms and Web-Based App

Analysis, Design and Implementation Report

Submitted in partial requirements for the degree of MSc Data Science (with Advanced Practice)

Date: January, 2024

Osasere Oture

Supervisor: Zahid Iqbal

ACKNOWLEDGEMENTS

My heartfelt appreciation and gratitude for this project goes to my project supervisor, Zahid Iqbal, for the consistent guidance, advice, and suggestions that greatly shaped the outcome of this research,

I would also like to thank all my lecturers and Teesside University for offering me an excellent opportunity to learn and broaden my knowledge in this field, thereby impacting my academic and professional development.

Lastly, I want to thank my husband Alexander Oture, my kids Nero and Runo Oture, my parents, siblings, and friends for their support and encouragement throughout my study programme.

Advanced Classification of Thyroid Disease Using Machine Learning Algorithms and Web-Based App

An analysis, design and implementation report for the development of an application to classify the different types of thyroid disease

ABSTRACT

Thyroid disease is a health concern related to the thyroid gland, which is vital for controlling the metabolism of the human body. Predominantly affecting women in their fourth or fifth decades of life, thyroid disease can result in physical and mental issues. Distinguishing the types of thyroid disease can be challenging since they share striking similarities with conditions of other diseases. Clinicians usually employ traditional diagnostic methods, especially the measurement of thyroid hormone levels, to identify and categorize forms of thyroid disease. This approach frequently presents the challenge of human-caused delays and errors. Therefore, this research improves the diagnostic process for thyroid disease by developing an automated system capable of classifying three thyroid conditions using five machine learning models and a deep learning model. Resampling techniques, such as SMOTE oversampling and Random undersampling, are utilized to correct the issue of class imbalance in the dataset. The issue of overfitting is also addressed through the use of hyperparameter tuning. Evaluating the overall performance of different classifier models takes into account the accuracy score and the F1-score, providing the harmonic mean of both precision and recall. The experimental analysis showed that the Gradient Boosting Classifier (GBC), using oversampling techniques, achieved the highest level of performance in classifying thyroid diseases, obtaining an accuracy and F1-Score of 99.76%. Furthermore, this study demonstrated that TSH was the most indicative biomarker for thyroid disease classification. Finally, we developed a web-based application utilizing the most effective model, GBC, which facilitates easy classification of thyroid diseases.

Key terms: Thyroid disease, Thyroid hormones, Oversampling, Undersampling, Flask Framework

ABBREVIATIONS AND ACRONYMS

ANN	Artificial Neural network
CNN	Convolutional Neural Network
CSS	Cascading Style Sheet
DL	Deep Learning
DTC	Decision Tree Classifiers
EDA	Exploratory Data Analysis
FP / FN	False Positive / False Negative
TP / TN	False Positive / True Negative
FPR	False Positive Rate
FTI	Free Thyroxine Index
GBC	Gradient Boosting Classifier
HTML	Hypertext Markup Language
KNN	K-Nearest Neighbor
LR	Logistic Regression
ML	Machine Learning
MLPC	Multilayer Perceptron Classifier
ROC	Receiving Operating Characteristic
RT	Random Forest
SMOTE	Synthetic Minority Oversampling Technique
SVM	Support Vector Machine
T3	Triiodothyronine
T4	Thyroxine
T4U	Thyroxine Utilization
TDs	Thyroid Diseases
TPR	True Positive Rate
TSH	Thyroid Stimulating Hormone
TT3	Total Triiodothyronine
TT4	Total Thyroxine
UCI	University of California Irvine

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

1. INTRODUCTION

1.1 Background

Thyroid disease is a widespread medical condition affecting a substantial population worldwide, especially in regions with insufficient levels of iodine in the soil or water. People of different ages and genders are prone to this illness, with women showing a higher inclination to develop thyroid-related disorders throughout their lives. Individuals between the ages of 17 and 45 are predominantly impacted by thyroid disease abnormalities (Duggal et al., 2020). However, thyroid disease typically manifests in women during their fourth or fifth decades of life (Freitas et al., 2016). The prevalence of thyroid disease is considered to be tenfold higher in women compared to men (Vanderpump, 2011).

Thyroid disease mainly arises due to an autoimmune malfunction in the thyroid gland, which plays a vital role in secreting thyroid hormones that are responsible for regulating the overall human physiological balance and health. The two types of active hormones secreted are levothyroxine (T4) and triiodothyronine (T3), both of which are required for the regulation of body temperature, heart rate, and blood pressure, as well as for the absorption of proteins, carbohydrates, and fat into the bloodstream. The thyroid stimulating hormone (TSH) controls the production and release of both the T3 and T4 hormones. Disruption in the secretion of these hormones can result in physical and mental issues, even death. Hypothyroidism and hyperthyroidism are the two widespread issues associated with the thyroid gland. Hyperthyroidism arises when the thyroid gland becomes hyperactive by producing surplus thyroid hormone. Hypothyroidism is the opposite of hyperthyroidism. It occurs when there is an underproduction of thyroid hormones (Farling, 2000).

Thyroid disease can present itself in a multitude of ways, and distinguishing the conditions caused by thyroid disease from other disease conditions can be challenging due to their striking similarities. In the field of endocrinology, thyroid disease is currently rated as the second most widespread condition, with a major challenge in accurately categorizing the different types of this disease in clinical treatment (Pan et al., 2016).

The detection of thyroid disease is mainly conducted through the laboratory examination of blood, which may contain noise that can lead to inaccurate predictions (Nguyen et al., 2015). Other traditional diagnostic techniques for thyroid disease involve clinical evaluation and medical imaging methods. While these approaches may be advantageous, they may also bring subjectivity and perhaps inhibit quick results gathering. The survey conducted by Montagna et al. (2023) shows that approximately four and a half years are used to diagnose a thyroid condition. In cases requiring rapid action, the possibility of an incorrect diagnosis or a delay in treatment can lead to a major source of stress and impact on wellbeing. Based on this, the prompt and precise classification of thyroid disease is imperative in order to provide suitable interventions and optimize patient outcomes.

The advancement in machine learning (ML) and its increasing use in the field of medicine have facilitated the creation of more sophisticated diagnostic tools. Machine learning (ML), with its ability to effectively handle large datasets and identify complex patterns, has significantly improved the accuracy of detecting and classifying health-related conditions. In the instance of thyroid disease, different machine learning algorithms have been employed to classify the various types of thyroid diseases using a range of datasets. However, the accuracy of models based on data quality and a shift to digitization are also crucial in offering real-time diagnostic assistance and improving timely decision-making among medical practitioners, which ultimately contributes to better patient care and outcomes.

1.2 Research Questions

In light of the aforementioned discussions in the background section, the following research questions are examined in this thesis:

- **Research Question 1:** Which biomarkers are the most significant contributors to classifying different types of thyroid diseases?
- **Research Question 2:** How do resampling techniques impact models' performance in accurately classifying thyroid diseases?
- **Research Question 3:** How does the integration of a machine learning model into a web-based application enhance the prompt classification of thyroid disease?

These questions aim to explore the utilization of machine learning and deep learning methods in the context of classifying thyroid disease, with a focus on developing a web-based application for faster diagnosis relying on the most significant biomarkers and features. Additionally, the questions delve into the comparative analysis of different machine learning approaches to identify the most effective method for achieving diagnostic accuracy.

1.3 Research Aim and Objectives

The aim of this research is to create a classification model that utilizes various machine learning models and a deep learning model to categorize three types of thyroid disease conditions. Additionally, the objective is to create a user friendly web-based application for the classification of thyroid diseases using the model with the highest performance. This overarching goal includes the following specific objectives:

- Conduct a comprehensive review of prior studies to gain a better understanding of existing knowledge. This knowledge will be leveraged to develop an enhanced classification system for thyroid diseases.
- Create a classification model by applying both machine learning algorithms and a deep learning algorithm.
- Address the issue of class imbalance in the target variable by employing resampling techniques, with the aim of optimizing the model's performance for accurate classification of thyroid diseases.
- Develop a web-based application that incorporates the most efficient machine learning classifier model in order to expediate the classification of thyroid diseases using the provided data.
- Conduct a performance comparison of the various classifier models to determine the most appropriate ones for classifying thyroid diseases.

1.4 Research Structure

This research work consists of five chapters: Introduction, Literature Review, Methodology, Implementation, Experimental Results and Discussions, and Conclusion and Future Work.

- The introduction presents an overview of the prevalence and impact of thyroid diseases on healthcare and discusses the difficulty of precisely classifying the disease. The chapter provides an outline of the research questions, aims and objectives, significance, and overall structure of the study.

- Chapter two covers research on traditional diagnostic methods, machine learning, and deep learning methods for thyroid disease classification. The chapter also examines prior studies on data mining techniques for addressing class imbalances and explores the importance of web-based applications in the healthcare field.
- Chapter three provides comprehensive details on the systematic and sequential procedures conducted in this study. These procedures encompass data collection, data preprocessing, data balancing, model application, model evaluation, and web app development.
- Chapter four provides an overview of the implementation and compares the results that arise from implementing models with different resampling techniques and evaluation metrics. This chapter also addresses the research questions of the study.
- The final chapter serves as the concluding section, providing a summary of the entire study, its discoveries, limitations, and potential for future research work.

2. LITERATURE REVIEW

2.1 Overview of Thyroid Diseases

The thyroid gland is structurally designed like a butterfly and is situated in the anterior section of the neck. Its dimensions vary across individuals and are shaped by various factors, including gender and age (Islam et al., 2022). The thyroid gland has the vital function of secreting active T3 and T4 hormones, which are necessary for controlling the metabolism of the body. Thyroid dysfunction can result in various forms of thyroid disorders, with hypothyroidism and hyperthyroidism being the most prevalent.

- **Hypothyroidism** is characterized by the deficient production of thyroid hormones by the thyroid gland; in other words, there are high levels of TSH and lowered levels of T3 and T4. Common symptoms include increased body weight, exhaustion, heightened sensitivity to low temperatures, difficulty passing stool, and dry skin (Chiasera, 2013).
- **Hyperthyroidism** refers to a condition characterized by excessive production of thyroid hormones. In other words, there is a deficiency of TSH and an excess of T3 and T4. The symptoms encompass weight loss, heightened appetite, heart palpitations, intolerance to heat, anxiety, and additional manifestations (Chiasera, 2013).

The major causes of thyroid diseases are malfunction of the autoimmune system, inadequate levels of iodine in the body, exposure to radiation therapy, gland excision, intake of medications like lithium, aging and pregnancy (Akash et al., 2023).

2.2 Traditional Diagnostic Methods

Thyroid diseases are normally diagnosed through a range of traditional methods, primarily based on clinical examinations, laboratory blood tests, imaging techniques, and a fine needle aspiration (FNA) biopsy. These methods are explained in detail below.

2.2.1 Clinical Examinations

Clinical examinations involve a comprehensive assessment conducted by healthcare professionals to identify physical indicators of thyroid dysfunction. This includes the inspection of the thyroid gland through manual examination, the assessment of vital signs, and the evaluation of symptoms such as fatigue, fluctuations in weight, and alterations in the texture of the skin and hair.

2.2.2 Thyroid Function Tests (TFTs)

TFTs are diagnostic procedures aimed at assessing the levels of thyroid hormones (T3 and T4) and TSH in the blood. Variations in the normal range of T3, T4, and TSH, which are 80-200 ng/dL, 4.5-11.2 mcg/dL, and 0.4-4.0 mIU/L respectively, may be an indication of hyperthyroidism or hypothyroidism. TFTs include several tests such as TT3, TT4, TSH, FTI, T4U and antibody tests. However, there is now more emphasis on prioritizing TSH as the main biomarker to determine the presence of thyroid diseases in laboratory tests. It is also important to note that when interpreting TFT results, other factors that may falsely impact the normal level of these tests should be considered. Factors such as pregnancy, clinical symptoms, and the medical history of the patient should be taken into account (Koulouri et al., 2013).

2.2.3 Imaging Methods

Over the years, the anatomy of the thyroid gland has been routinely visualized using ultrasound imaging. It assists in the detection of nodules, cysts, and other abnormalities. Deng et al. (2014), utilizing a dataset of 146 patients, evaluated the diagnostic efficacy of conventional ultrasound, along with CEUS and ARFI, in distinguishing focal solid thyroid masses. The specificity and accuracy for the Ultrasound were 81.5% and 78.3%, respectively.

2.2.4 Fine Needle Aspiration (FNA) Biopsy

Bonjoc et al. (2020) study of the diagnosis of thyroid cancer showed that when a nodule is found, a FNA biopsy is frequently performed to collect tissue samples for further investigation. This process assists in determining if a nodule is benign or cancerous, which provides guidance for subsequent treatment selections. Bahaj et al. (2022) utilized FNAC to diagnose thyroid disease nodes in 314 patients from a tertiary referral center in Makkah. The accuracy of the findings was 74.8%, with a specificity score of 82.1%.

While these procedures are useful, they have fundamental limitations in terms of accuracy and timely classification of the various thyroid disorders, underscoring the need for more advanced and precise diagnostic approaches. Consequently, the integration of machine learning algorithms and data driven models has been embraced to enhance the precision and effectiveness of diagnosing thyroid diseases.

2.3 Advanced Diagnostic Methods

In the diagnosis of thyroid disease, recent advancements have extended beyond traditional methods. The utilization of advanced diagnostic methods, involve machine learning, deep learning, and web-based system approaches that leverage computational algorithms to analyse diverse results or data obtained from traditional approaches and complex datasets. Consequently, they provide a more precise, efficient, and prompt diagnosis by identifying patterns studied within the data. This section presents various ways these advanced methods have been employed in previous studies with the aim of improving the timely classification of different thyroid diseases.

2.3.1 Machine Learning Methods

Over the years, numerous remarkable studies have been conducted in the field of medicine, utilizing various machine learning methods, including the classification of thyroid diseases. Chen et al. (2020) identified ultrasonic traits associated with cancerous thyroid nodes using LASSO (Least Absolute Shrinkage and Selection Operator) in conjunction with logistic regression (LR). A scoring system was also employed alongside random forest (RF) for the disease classification. The combination of logistic lasso regression (LLR) and RF exhibited superior performance, achieving an accuracy of 82%.

Abbad et al. (2021) employed KNN with different distance functions to detect thyroid disease. The analysis was based on two datasets from KEEL and Dera Ghazi Khan government hospital in Pakistan. The implementation of KNN on these datasets was conducted using three feature selection techniques. The Euclidean and Cosine distance functions demonstrated superior accuracy on the dataset from Pakistan when utilizing the x^2-based feature selection.

Olatunji et al. (2021) utilized data from King Fahad Specialist Hospital in Saudi Arabi, which contained 14 attributes. RF, SVM, NB and ANN techniques were employed for the early categorization of thyroid disease, utilizing a reduced number of features. RF emerged the most effective with an accuracy of 90.91% using seven features.

Tabassum et al. (2022) conducted a comparative analysis of ML and DL methods for predicting thyroid disease. They implemented a dual-pronged approach, initially employing ML algorithms such as LR, DT, SVM Linear Kernel, SVM RBF Kernel and RT and evaluated their performance accuracy. The second stage involved the implementation of DL algorithms specifically RNN

(Recurrent neural network) utilizing the same performance metrics. ML algorithms, DT and RF exhibited the highest level of accuracy at 98.16% while DL algorithm RNN achieved a performance accuracy of 97%. A dataset sourced from Kaggle comprising 3,772 observations and 30 attributes was utilized for this research.

Sankar et al. (2022) assessed the efficacy of the XGBoost algorithm with those of LR, KNN, and DT for predicting thyroid disease. These algorithms were applied to the UCI thyroid dataset. The accuracy scores were: LR – 81.25%, KNN– 87.50%, DT – 96.85% and XGBoost with the highest score of 98.59%.

Using the WEKA (Waikato Environment for Knowledge Analysis), Kumar et al. (2023) applied several data mining approaches to classify instances of hypothyroid illness. Dimensionality reduction techniques were employed to identify subsets of the attributes, while the J48 and decision stump tree algorithms were applied to the data. The findings of the study confirmed the J48 algorithm to have a remarkable accuracy of 99.58% and a lower error rate, thereby surpassing the decision stump (also referred to as an ensemble technique).

2.3.2 Deep Learning Methods

Deep learning (DL) is a part of machine learning that utilizes neural networks with several layers to obtain important information, usually from larger datasets or image data. Researchers have successfully employed DL in categorizing thyroid diseases. Borzouei et al. (2020) presented a system that employed neural network and logistic regression for diagnosing hypothyroidism and hyperthyroidism. The models were utilized on a dataset consisting of 310 patients from a hospital located in Hamadan. The evaluation of the models performance was based on average accuracy. The neural network model achieved an accuracy of 96.3%, surpassing the logistic regression model, which achieved 91.4% accuracy.

Dixit et al. (2023) presented a multi-layer neural network model that utilizes a combination of customized loss functions, machine learning, and feature engineering processes to improve the efficiency of the model on the thyroid dataset from the UCI source. An accuracy of 92.36% was achieved for the eight-class classifier in this study.

Balasree et al. (2023) proposed a system that utilized the SoftMax multi-perceptron neural network and feature analytic models for thyroid disease classification accuracy. ISCF (Intensive Cluster

Feature Selection) was employed for selecting the features based on their minimal accuracy and then trained using MPNN (Multi-perceptron neural network). The ISCF-MPNN resultant variables provided in the study demonstrate that the highest classification accuracy is 97% in relation to the influence rate of thyroid disease.

The research by Brindha et al. (2023) examined the effectiveness of two classifier models, CNN and SVM, in detecting hypothyroidism and hyperthyroidism. The models were trained using the dataset from the UCI library. The CNN classifier performed better, with an accuracy score of 89% and precision of 87%.

2.3.3 Web Based Systems

Web-based systems are crucial in healthcare, boosting accessibility and providing an intuitive interface that facilitates prompt data input and retrieval of results. Vasan et al. (2018) developed a web application for predicting thyroid disease. Python Flask was utilized for the back-end coding and HTML for the front-end web page. The UCI dataset, which consists of 215 blood test samples and five characteristics, was used for modelling. Logistic regression (LR) achieved the maximum accuracy of 96.92%. The logistic regression (LR) algorithm was incorporated into the web application to forecast the specific disease category.

N. Ananthi et al. (2022) employed reactJS to develop a web application that predicts six classes of thyroid diseases. The ResNet algorithm was applied to detect thyroid diseases, while the techniques of optimization and loss minimization were utilized to boost performance accuracy. The web app's functionality involved inputting an X-ray image of the neck area into the web application, which then analyses the image using a pre-trained model based on a thyroid image dataset. A prediction of the thyroid type is then made based on the learned patterns.

Nugroho et al. (2023) utilized the MoRbAC approach in a web-based application to automatically classify malignant objects in ultrasound images. The application employed a streamlined Chan-Vese active contour model and image morphological procedures. After being tested on twenty images of thyroid lumps and breast lesions, the system achieved a precision of 98.75%.

2.4 Data Balancing Techniques

Class imbalance in clinical datasets poses a common challenge, with potential significant impacts on model performance if left unaddressed. To address this issue, various data balancing or resampling techniques, including oversampling and undersampling, are typically employed to balance the classes of data. In the context of thyroid disease classification, researchers have explored different resampling methods in their studies.

Srivastava et al. (2021) utilized the Bordline_synthetic oversampling approach (BL_SMOTE) in conjunction with ensemble classifiers, namely random forest and decision tree, to predict thyroid disease. The oversampling technique was employed to tackle the issue of class imbalance in the UCI data. The model proved to be effective, with a precision score of 99.12%.

Alyas et al. (2022) employed various ML methods, including RT, DT, and KNN, to classify different forms of thyroid illnesses. The undersampling technique was used to address the issue of an imbalanced distribution within the dataset. The evaluation of the research effort was based on precision and recall. The random forest algorithm demonstrated the highest level of accuracy, achieving an average performance of 94.8%.

Mollica et al. (2022) established a novel method for classifying thyroid diseases using Probabilistic Graphical Models (PGMs) on a data containing 730 genes from 60 samples of thyroid. The model's performance was compared with that of SVM and DT. With oversampling techniques applied, the PGM model yielded better output in the study compared to classical machine learning algorithms

2.5 Key Features For Thyroid Disease Classification

Machine learning methods have also been utilized by researchers to identify the most indicative markers for classifying types of thyroid diseases.

Duggal et al. (2020) utilized three feature selection approaches—univariate, recursive feature elimination, and tree-based selection—in predicting thyroid diseases. Across all methods, TSH, TT4, T3, T4U and FTI emerged as the most significant features for the prediction, except in the case of tree-based approach, where age was identified as a significant feature.

Balasubramanian et al. (2022) employed CART and Random Tree (RT) as supervised learning methods and PCA as unsupervised learning method to demonstrate that FTI, TSH, TT4, and T3 are the topmost attributes for classifying thyroid disease from a KEEL dataset.

(Savi and Nuriyeva, 2022), utilizing feature importance based on RT, identified TSH, FTI, TT4, On_thyoxine, T3, and T4U as the significant features for the prediction of thyroid diseases. The utilization of these features yielded better performance, with ANN outperforming all other models, achieving an accuracy score of 98%.

The goal of this review was to identify similar methodologies applied by previous researchers in developing a classification model that takes into account a diverse range of features. The aforementioned studies illustrate that researchers very often integrate diverse methodologies and tools to develop efficient models. These models are then assessed across multiple criteria to identify the one with the best performance for implementation.

3. METHODOLOGY

This chapter provides a thorough understanding of the classification methods employed to accomplish the objectives and address the research questions outlined in Chapter 1 of this project. The methodology utilized in this project encompasses the multiple phases of the data science lifecycle.

3.1 Proposed Classification Model

The proposed model designed for this project is represented in Figure 1, which comprises nine distinct phases outlining the various tasks carried out in this study. Hence, this chapter is structured into nine sections, each offering a thorough explanation of every aspect. It commences with problem understanding and progresses through data collection, culminating in the deployment of the model in a web-based application. The design of the model was specifically tailored to achieve a precise classification of thyroid diseases (TDs).

The implementation of this model was performed on a Windows 11 operating system. The hardware specifications include an AMD Ryzen 7 7730U CPU with Radeon Graphics, operating at a clock speed of 2.00 GHz and supported by 16.0 GB of RAM. The operating system is a 64-bit version compatible with a processor based on the x64 architecture.

For code development, the Visual Studio Code Integrated Development Environment (IDE), specifically the Community 2022 edition, was utilized. This IDE offers a versatile and user-friendly environment for coding and project organization, thereby improving the efficiency of the model development process. The code was written in Python v3.12.0.

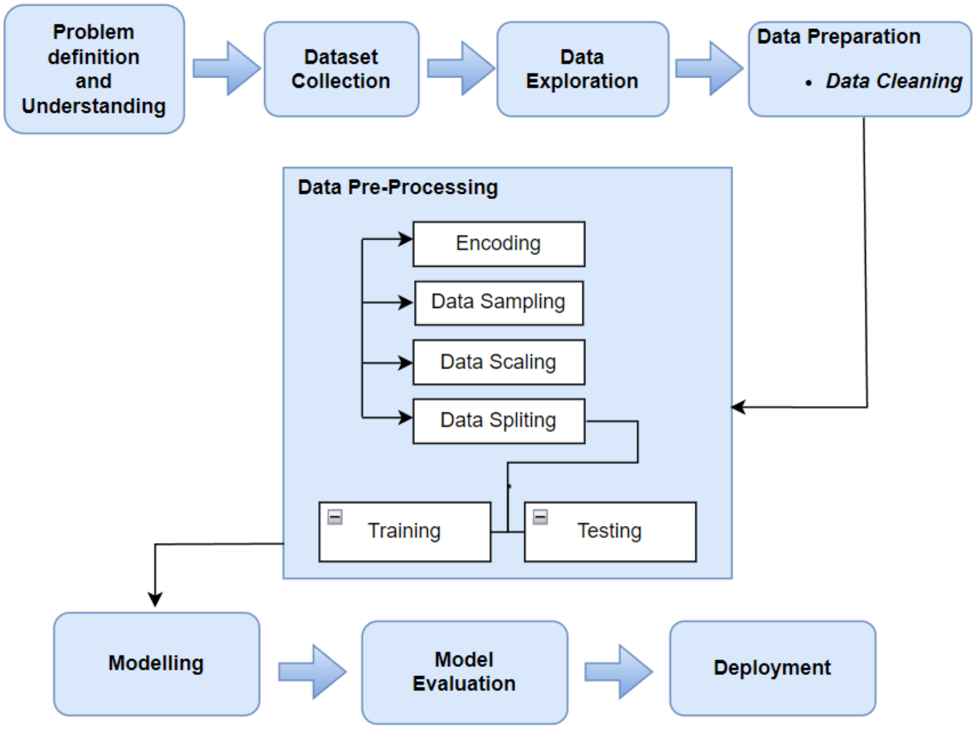


Figure 1: Proposed Model Diagram.

3.2 Problem Definition and Understanding

The overall success of any task, irrespective of the organizational sector, must commence with this initial phase. This step is of utmost importance as it establishes the foundation for the subsequent stages. During this stage, the project's goals and objectives are outlined, forming the basis for a concise description of the problem to be addressed through the implementation of data mining tools. The efficacy of the subsequent stages depends on the clarity and knowledge of the problem at hand during this preliminary phase. For this project, the task is to precisely classify the particular type of thyroid disease in a patient by using a variety of distinct features or independent variables. This process entails the application and comparison of various ML algorithms, a DL algorithm, and other data mining techniques to identify the most effective algorithm for classifying thyroid diseases.

3.3 Dataset Collection

The thyroid disease dataset used for this research was obtained from a Kaggle link, comprising 9,172 patient observations and 31 attributes. The dataset comprises a consolidated compilation of thyroid illness records sourced from the UCI repository. The dataset is deemed appropriate for this research because of the significant number of patient records, and limited prior research on it. The dataset's target feature comprises a range of medical diagnoses related to thyroid disease. However, for the purpose of this research, only patient cases with diagnoses classified as "negative," "hypothyroid," or "hyperthyroid" are taken into account. “Negative” signifies that a patient does not have any signs of thyroid disease. In the execution of the models, 22 attributes are utilized, comprising sixteen (16) categorical variables and six (6) continuous variables denoted as "float" in Table I.

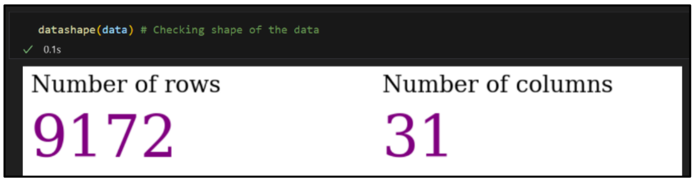
Column Name	Data Type	Data Description
Age	Integer	This refers to the age of the patients observed.
Sex	String	Gender information showing whether the patient is male or female.
On_thyroxine	Boolean	This indicates if the patient is using thyroxine medication.
Query_on_thyroxine	Boolean	This indicates if the patient has undergone an examination to determine the use of thyroxine medication.
On_antithyroid meds	Boolean	This is to shows if the patient is using antithyroid medication.
Sick	Boolean	This shows the well-being of the patient.
Pregnant	Boolean	Indicates if or not the female patient is an expectant mother.
Thyroid surgery	Boolean	This is to show if the patient has taken out a thyroid gland.
I131-treatment	Boolean	Specifies if the patient is receiving the I131 treatment or not.
Query_hypothyroid	Boolean	This shows if a patient has been tested for hypothyroid disease.
Query_hyperthyroid	Boolean	This denotes whether a patient has been tested for hyperthyroid disease.
Lithium	Boolean	Shows whether the patient is taking Lithium. Lithium is a medication used for the treatment of bipolar disorder as a mood stabilizer.
Goitre	Boolean	Denotes whether a patient has the presence of goitre.
Tumor	Boolean	Denotes whether the patient has the presence of a tumour.
Hypopituitary	Float	This shows whether the patient has a hypopituitary gland.
Psych	Boolean	This determines the patient’s psychological state.

TSH	Float	This shows the level of TSH hormone in the blood from laboratory work. TSH stands for thyroid stimulating hormone. High and low TSH are signs of thyroid.
T3	Float	This shows the concentration of T3 in the blood based on lab tests.
TT4	Float	This shows the concentration of TT4 in the blood based on lab tests.
T4U	Float	This shows the concentration of T4U in the blood based on lab tests.
FTI	Float	This shows the concentration of FTI in the blood based on lab tests.
Target	String	This shows the medical diagnosis of thyroid diseases represented by letters.

Table I: Description of some Attributes of The Thyroid Disease Dataset.

3.4 Data Exploration

Data Exploration, often referred to as exploratory data analysis (EDA), is a crucial stage in data analysis after obtaining the dataset. During this phase, an extensive examination of the source data is conducted to understand the data and obtain insights regarding its structural composition, statistical summary, and characteristics. This process involves assessing the data shape, data types, patterns, relationships between variables and identifying potential issues such as missing values and outliers. The knowledge gained from this stage provided valuable information that influenced decision-making in the subsequent phases of the proposed model.



```
datashape(data) # Checking shape of the data
✓ 0.1s
```

```
Number of rows          Number of columns
9172                    31
```

Figure 2: Checking The Shape of the Data.

3.5 Data Preparation

The objective of this phase is to prepare the data for modelling. This involves cleansing and transforming the data into a format that is appropriate for applying classification algorithms. According to Brownlee (2020), data preparation is a time-consuming but necessary process because raw data cannot be directly utilized to train or evaluate machine learning (ML) algorithms. The next section examines the procedures involved in cleaning the data utilized for this study.

3.5.1 Data Cleaning

The initial step in preparing the data is the data cleaning procedure, a systematic process for resolving issues, inconsistencies, and errors within a dataset. Data cleaning enhances data integrity, ensuring accurate outcomes from model applications. Hence, in implementing this model, dedicated modules were created to handle all aspects of data cleaning systematically.

A pivotal part of this cleaning procedure involves checking and removing any instances of missing or null values. In the utilized data, the "TBG" column of the dataset was found to have the highest number of missing values, totalling 8,823, as shown in Figure 3. This column was eliminated, along with other redundant columns, to enhance the efficiency of the algorithms. Another aspect of the data cleaning involved examining the data for duplicate entries, and it was observed that there were no instances of duplicate values.

Finally, in the data cleaning step, the target variables were appropriately mapped to their respective classes, ensuring a refined and prepared dataset for subsequent model implementation.

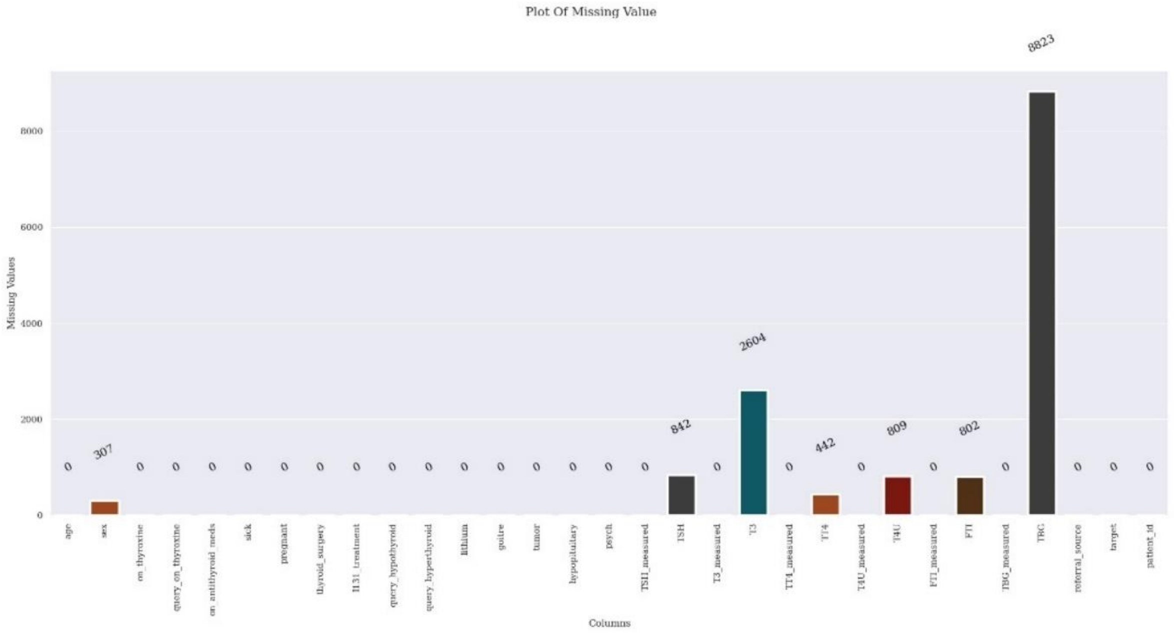


Figure 3: Checking The Missing Values.

3.6 Data Preprocessing

The data preprocessing approach fundamentally impacts the general performance of machine learning (ML) models (Alexandropoulos et al., 2019). Data preprocessing is an essential stage that improves data quality and mitigates complexities, given that most real-world datasets are likely to exhibit attributes such as noise, inconsistencies, and redundancies. Therefore, processing raw datasets is imperative to transform the data in a way that is suitable for applying data mining models to achieve optimal performance. The dataset utilized for this study underwent various preprocessing steps.

Firstly, categorical features were encoded into numerical values, and then the data was standardised using Standardscaler (Z-score normalisation). Resolving the issue of data imbalance was also a crucial aspect of the preprocessing phase. Resampling techniques were utilized to address the significant imbalance and unequal distribution of classes in the target variable of the dataset. The purpose of this strategic intervention was to synchronise the distribution and ensure an unbiased representation of the target variable. Finally, the data was partitioned into model training and model testing. This division was done conventionally, with 80% of the data assigned to model training and the remaining 20% assigned to the testing dataset.

The subsections below provide more details on some of the data pre-processing and transformation steps conducted in this study.

3.6.1 Data Encoding

This involves transforming categorical variables into numerical ones to fit appropriately into the model. Table 1 shows that sixteen of the 22 features extracted from the dataset were categorical variables that had to be transformed into numerical representations. The label encoding technique was utilized by assigning a distinct label to a categorical variable with an integer value. Appendix 1 depicts the source code for encoding the categorical features.

3.6.2 Data Resampling

In machine learning, data resampling methods are often applied to address the issue of class imbalance in a dataset. An unbalanced dataset is characterised by a significant difference in instances between one target class label and the other(s). This uneven data distribution within a dataset can result in inaccuracies when training or evaluating machine learning models. Data resampling involves selecting data points from an existing dataset to form a new dataset to build and train models. The two commonly used resampling methods by most researchers in addressing the issue of data imbalance are "oversampling" and "under-sampling."

In this project's work, the dataset is highly imbalanced. The target variable consists of three classes: "negative," which accounts for 89.45% of observations in the dataset; "hypothyroid," with 8.05% of the observations; and "hyperthyroid," with just 2.49% of the observations. Resampling techniques are used to resolve the class imbalance in the target variable under consideration.

A. Oversampling

Oversampling involves augmenting the number of occurrences or instances of the lesser class to generate an unbiased dataset. The creation of the new data points entirely depends on the features of the pre-existing lesser or minority class. An advantage of the oversampling method is that all information from the majority and minority classes is conserved.

Oversampling can be achieved by replicating existing instances of the minority class (simple oversampling) or creating synthetic instances to represent the minority class (SMOTE oversampling). The SMOTE technique was employed to rectify the data imbalance in this research study.

- **SMOTE (Synthetic Minority Oversampling Technique)**

  The SMOTE algorithm improves classifier model performance by creating artificial instances for the minority class rather than just replicating existing instances. Bias in data is therefore eliminated by avoiding the creation of identical instances. The SMOTE technique uses the KNN algorithm to create new instances or data points. It achieves this by selecting instances closely located in the feature space of the minority class; then, it calculates the distances between these instances by drawing a connecting line and then selecting a point on that line, as illustrated in Figure 4.

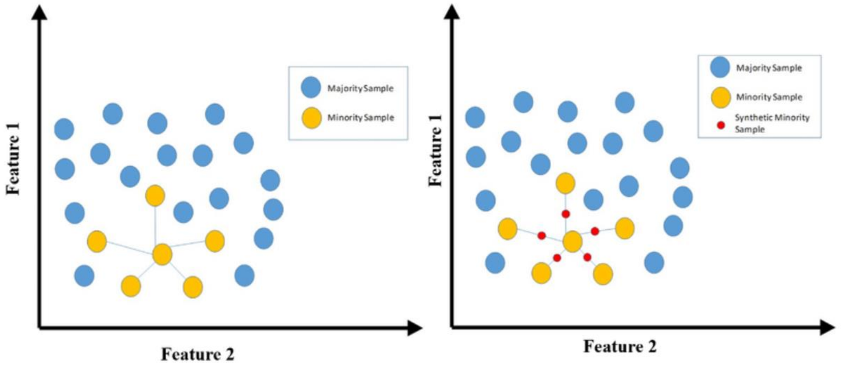


Figure 4: Illustration of SMOTE Oversampling Approach
Source: ‘https://www.sciencedirect.com/science/article/pii/S174680942100001X’

B. Undersampling

Undersampling is the inverse of oversampling, as illustrated in Figure 5. Undersampling entails lessening the number of instances belonging to the majority class. This approach produces a dataset with a high number of dimensions, leading to accelerated training time. One drawback of this procedure is that crucial information from the majority class, which could be relevant for performance accuracy, may be discarded. While undersampling encompasses two strategies, namely, random undersampling and informed undersampling, only the former has been employed in this study.

- **Random Undersampling**

  Random undersampling is a straightforward and non-experimental strategy that aims to tackle the problem of imbalanced data by randomly removing instances of the majority class. Unless the dataset is vast, there is a risk of losing essential information, which might negatively impact the model's performance.

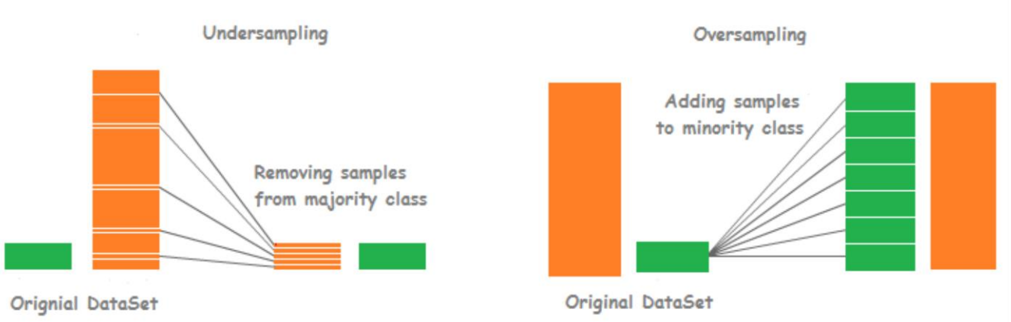


Figure 5: Comparison of Oversampling and Undersampling Techniques
Source: ‘https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9078901’

3.6.3 Data Scaling

Data scaling, also referred to as "feature scaling," is a preprocessing method used to standardise the values of features or variables in a dataset with different ranges. The purpose of this is to guarantee that all the characteristics are making an equal contribution to the performance of the models during the learning phase. Data scaling is essential when using algorithms that rely on distances or gradients, such as KNN, SVM, LR, and ANN.

The dataset has been standardised using the "StandardScaler" method, which applies z-score normalisation to ensure a consistent scale. The StandardScaler function transforms each feature in a dataset, setting the average value to zero and the standard deviation value to one. The mathematical representation of this is as follows:

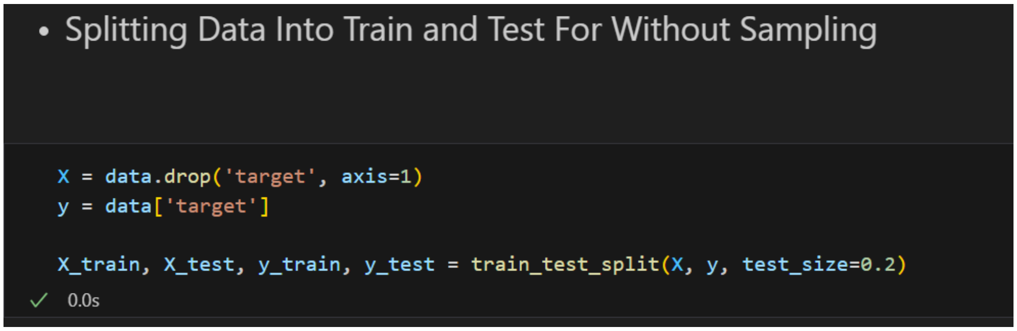
$$z = \frac{x - mean(\bar{x})}{standard\ deviation\ (\sigma)}$$

Where,

$\boldsymbol{\bar{x}}$ *is the mean.*
$\boldsymbol{\sigma}$ *is the standard deviation.*

3.6.4 Data Splitting

Before applying the models to the pre-processed data, it was first partitioned into two subsets for training and testing using the traditional ratio 80:20 for unsampled data and the ratio 75: 25 for oversampled and unsampled data as shown in Appendix 2.



- Splitting Data Into Train and Test For Without Sampling

```
X = data.drop('target', axis=1)
y = data['target']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

✓ 0.0s

Figure 6: Data Splitting Code for Unsampled Data.

3.7 Modelling

In this study, machine learning (ML) models have been mainly used for classifying thyroid diseases. A deep learning (DL) model was also employed, and the performance accuracy is compared to that of machine learning (ML) models.

3.7.1 Machine Learning Models

- **Decision Tree (DT) Classifier:** This decision-making tool makes predictions about the target class by applying a decision rule based on dataset features. The decision tree (DT) classifier uses a hierarchical structure like a flowchart, with internal nodes representing distinct features, branches containing decision rules, and leaf nodes indicating the resulting conclusions (Charbuty and Abdulazeez, 2021). In a decision tree, data is partitioned into subsets, commencing at the root node and concluding at the leaf nodes, which signify the final predictions or outcomes. The definitive predictions are based on the veracity of the features stated at each internal node.

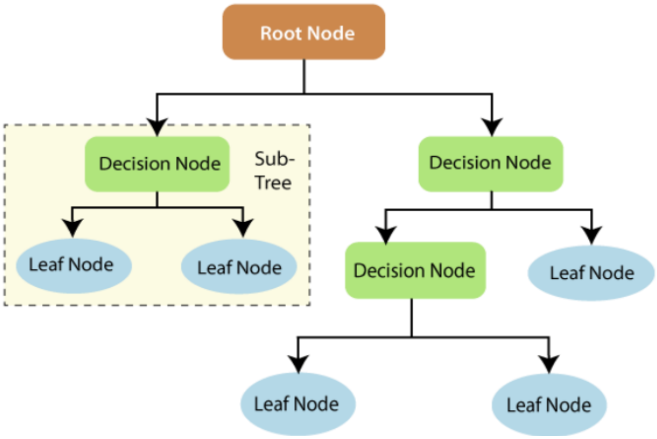


Figure 7: Decision Tree Classifier (DTC)

Source: ‘https://www.jastt.org/index.php/jasttpath/article/view/65/24’

- **Support Vector Machine (SVM) Classifier:** A supervised ML technique that categorizes data into discrete classes using a hyperplane inside the feature space. In two-dimensional space, the hyperplane, or decision boundary, divides a plane into two pieces. Categories are created for instances or data points on either side of the divide. SVM relies on support vectors, the data points closest to the hyperplane, and the margin, which is the distance between them. SVM finds the hyperplane with the highest margin and closest to class data points during training. SVM classifiers are used in one-vs.-one or one-vs.-all multiclass problems. Figure 8 depicts the graphical representation of SVM.

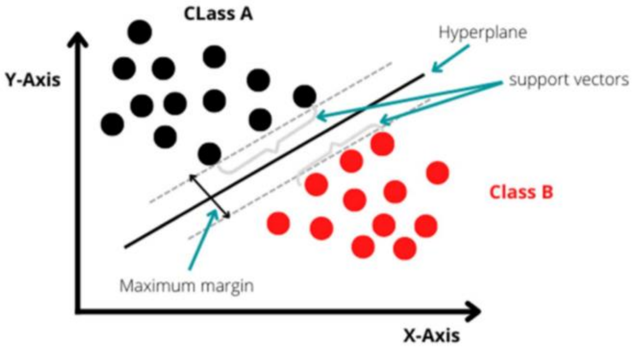


Figure 8: Support Vector Machine (SVM) Classifier
Source: ‘https://jesit.springeropen.com/articles/10.1186/s43067-023-00101-5’

- **Logistic Regression (LR):** Logistic regression is an algorithm used in classification tasks to predict the class of an outcome variable based on a given set of dependent variables. In this study, the dependent variables are the classes of thyroid disease, which are "negative," "hyperthyroid," and "hypothyroid." Logistic regression (LR) employs the sigmoid, or logistic function, to predict the likelihood that a given instance belongs to a particular class. This function converts any inputted feature value into a different value between 0 and 1. Logistic regression is expressed statistically as follows:

$$P(class = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 . feature_1 + \ldots \beta_n . feature_n)}}$$

Where,

P(class=1) *is the probability that the instance belongs to class 1.*
e *is the base of the natural logarithm.*
$\boldsymbol{\beta_0}$ *,* $\boldsymbol{\beta_1, \ldots, \beta_n}$ *are the coefficients (weights) associated with the features.*
$\boldsymbol{feature_1}, \ldots, \boldsymbol{feature_n}$ *are the values of the input features.*

During training, logistic regression tries to adjust the coefficients (β) in order to make predictions. These adjustments are based on the difference between the predictions and the real classes. This is typically done using techniques like maximum likelihood estimation or gradient descent.

- **K-Nearest Neighbour (KNN) Classifier:** KNN predicts a new data point's class using the K closest neighbours from the training data (Gou et al., 2012). Figure 9 shows how KNN uses Euclidean, Manhattan, or Minkowski distance to calculate the distance between the new test data point and all the training data points. Predictions are produced based on these neighbours' classes or values. When inputted data contains anomalies, a significant or odd K value should be considered to remove classification ties. The most common distance calculation method is Euclidean distance:

$$distance(x, X_i) = \sqrt{\sum_{j=1}^{d} (x_j - X_{ij})^2]}$$

Where

x is a point in the 'd-dimensional' space with coordinates $x_{1,} x_2, x_3 \dots x_d$.

X_i *is a another point in same 'd-dimensional' space with coordinates*$X_{i1,}\ X_{i2,}\ X_{i3} \dots\ X_{id}$.

D represents the 'total number of dimensions' inside the given space.

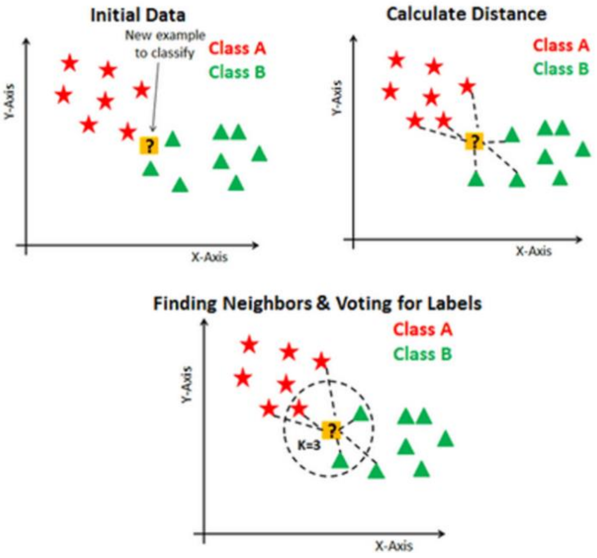


Figure 9: K-Nearest Neighbor Classifier.

Source: ['https://machinelearninggeek.com/knn-classification-using-scikit-learn/'](https://machinelearninggeek.com/knn-classification-using-scikit-learn/)

- **Gradient Boosting (GB) Classifier**: The Gradient Boosting Classifier is a member of the boosting family. The algorithm is an ensemble learning technique that constructs a unified predictive model by aggregating the predictions of multiple simple models, sometimes known as "weak learners." The gradient-boosting classifier is very efficient at reducing errors of bias and variance. Gradient boosting works by successively training simple models while employing gradient descent to reduce loss. This process is repeated until the final combined model is a better predictive model.

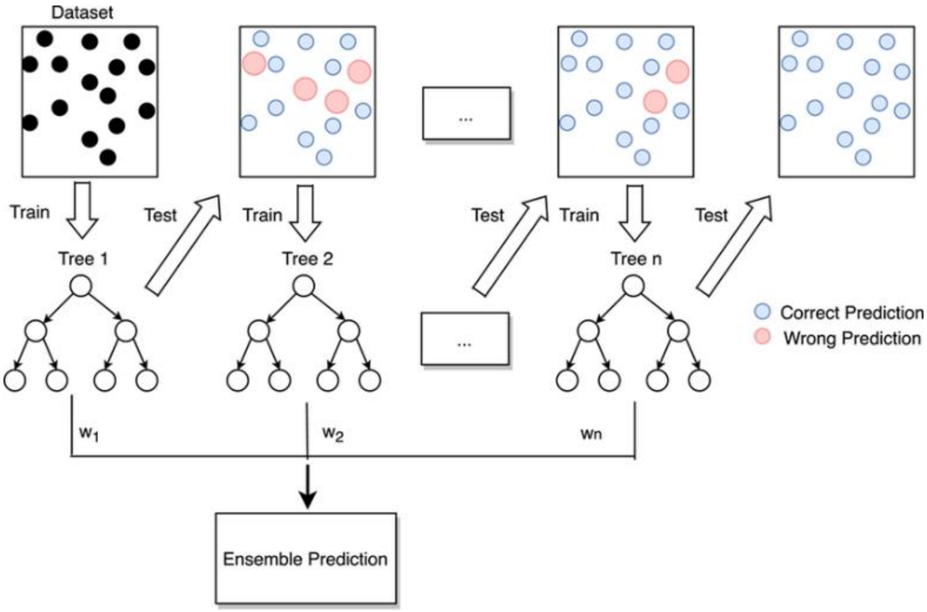


Figure 10: Flow Diagram of Gradient Boosting Classifier.
Source: ‘https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/2020MS002365’

3.7.2 Deep Learning Models

- **Multilayer Perceptron (MLP) Classifier:** The MLPC is a feed-forward neural network consisting of different weighted perceptron layers. As illustrated in Figure 11, the critical elements of MLPC comprise the input layer, one or more concealed or hidden layers, and the output layer. In the training process, the data fed into the input layer is processed using different weights or parameters in the hidden layer, and the final outcome is presented in the output layer. Backpropagation and loss calculation are used to reduce losses in training. The formula for MLPC can be represented as:

$$y_t = b_0 + w_{t1}x_1 + w_{t2}x_2 + .. + w_{n1}x_n$$

Where,

y_t represent the value predicted at t.

w_{tj} denote the weight associated with the j^{th} input at time t.

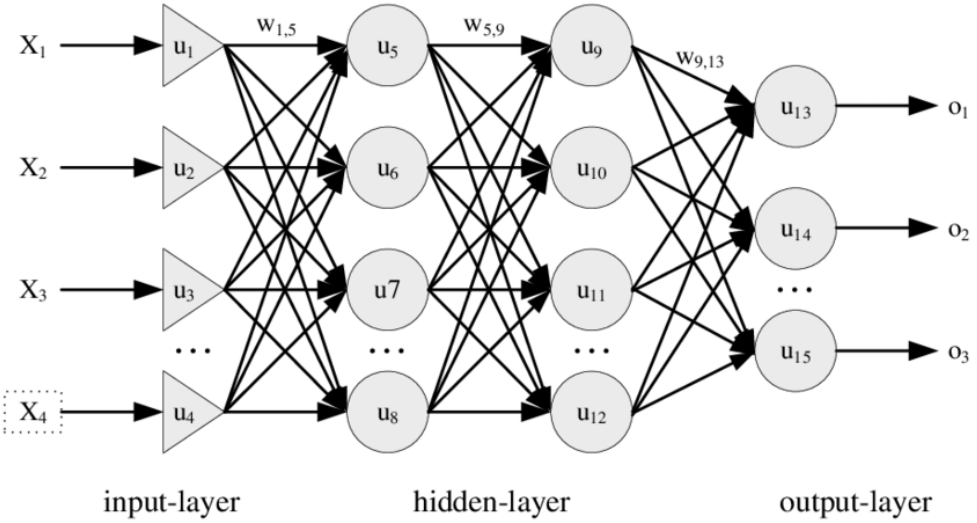


Figure 11: Architecture of Multilayer Perceptron Classifier.

Source: ‘https://www.researchgate.net/profile/Adhistya-Permanasari-2/publication/265784353_Utilization_of_Neural_Network_for_Disease_Forecasting/links/56ea3c3808ae95bddc2a6871/Utilization-of-Neural-Network-for-Disease-Forecasting.pdf’

Based on Figure 11, the nodes, are denoted as "u". The inputs are represented as "x" and the outputs as "o". The connections between nodes are unidirectional and have trainable weights, denoted as "w".

3.8 Model Evaluation Metrics

This segment provides an overview of the metrics utilized to evaluate the performance of the machine learning (ML) and deep learning (DL) models implemented in this research. Four possible outcomes are determined by comparing the output of the test sample to the actual label. These are false positive (FP), true negative (TN), true positive (TP), and false negative (FN).

- True positive (TP) indicates that the outcome of the target class has been accurately identified.
- True negative (TN) indicates the ability of the model to accurately foretell the absence of thyroid disease or the target class.

- False positive (FP) indicates an identification error in the target class outcome. A situation where the model incorrectly predicts the instance as hyperthyroid or hypothyroid when it is negative for thyroid disease.
- False negative (FN) is an indication that the anticipated target is not present. The model incorrectly predicts the instance as negative when it is hyperthyroid or hypothyroid.

The evaluation metrics are discussed in the subsections below.

3.8.1 Accuracy

Accuracy *(ACC)* measures the proportion of accurate predictions (TP + TN) against the overall number of predictions (P + N). This metric exhibits superior performance when there is a uniform distribution of instances across all classes in the dataset. The calculation of accuracy is as follows:

$$ACC = \frac{No. of\ accurate\ predictions}{Total\ no. of\ Predictions}$$

$$ACC = \frac{(TP + TN)}{(TP + TN + FN + FP)}$$

Also represented as:

$$ACC = (TP + TN)/(P + N)$$

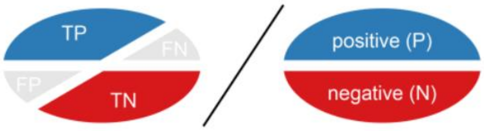


Figure 12: Ellipse Calculation for Accuracy (ACC)
Source: ‘https://classeval.wordpress.com/introduction/basic-evaluation-measures/’

3.8.2 Precision

Precision (PREC) is a metric that computes the reliability of a model in correctly predicting positive outcomes. The computation entails estimating the proportion of accurate positive predictions (TP) to the overall number of predictions classified as positive (TP + FP).

$$PREC = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Also expressed as:

$$PREC = (TP)/(TP + FP)$$

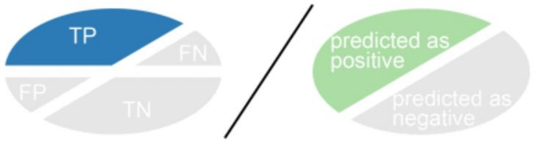


Figure 13: Ellipse Calculation for Precision (PREC)
Source: ‘https://classeval.wordpress.com/introduction/basic-evaluation-measures/’

3.8.3 Recall

Recall (REC) is synonymous with sensitivity. The calculation is obtained by dividing the count of true positive predictions (TP) by the total count of real positive samples (P). The recall score quantifies the model's capacity to classify positive samples precisely. A high recall signifies a substantial amount of accurately identified positive samples.

$$REC = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$REC = \frac{(TP)}{(TP + FN)}$$

Also expressed as:

$$REC = (TP)/(P)$$

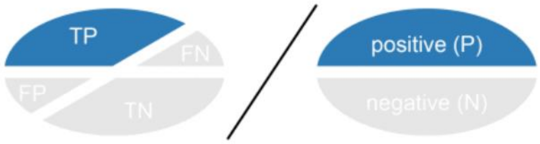


Figure 14: Ellipse Calculation for Recall (REC)
Source: ‘https://classeval.wordpress.com/introduction/basic-evaluation-measures/’

3.8.4 F1-Score

The F1-Score is computed as the harmonic mean of precision and recall. Both metrics have an equal impact on the score, ensuring that the F1 metric precisely represents the reliability of a model. This metric is better suited for a class distribution that is not evenly balanced.

$$F1 - Score = 2 * \frac{Precision + Recall}{Precision + Recall}$$

3.8.5 ROC Curve

ROC curve is an abbreviation for Receiver Operating Characteristic Curve. The ROC is a graph that compares the True Positive Rate (TPR) or sensitivity with the False Positive Rate (FPR) or specificity for various categorization threshold values.

$$\boldsymbol{TPR} = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$

$$\boldsymbol{FPR} = \frac{False\ Positive\ (FP)}{True\ Negative\ (TN) + False\ Positive\ (FP)}$$

3.8.6 Confusion Matrix

It is a graphical depiction that summarises the different occurrences when a model correctly or incorrectly predicts values in a dataset relative to the actual values. The confusion matrix offers a comprehensive perspective on the performance of a classification model. This study used a 3 × 3 matrix as it focuses on a multiclass task.

3.9 Deployment

This is the final phase of the model workflow, where the algorithm with the best performance accuracy is used in developing a web application that classifies the type of thyroid disease in patients. The Flask framework is used to develop the web application. Flask is recognised for its lightweight nature and user-friendly characteristics, without relying on specific database libraries or tools to function. In this study, Flask is employed in conjunction with Bootstrap, HTML, and CSS for the website's design. Flask is employed for backend development, whereas Bootstrap is utilized for frontend user interface design. The trained model is stored in a pickle file, indicating that the objects undergo transformation into a byte stream before being stored in a file or database. Figure 15 shows the flask framework's architectural configuration, which illustrates the interaction between the website and the trained model in the pickle file.

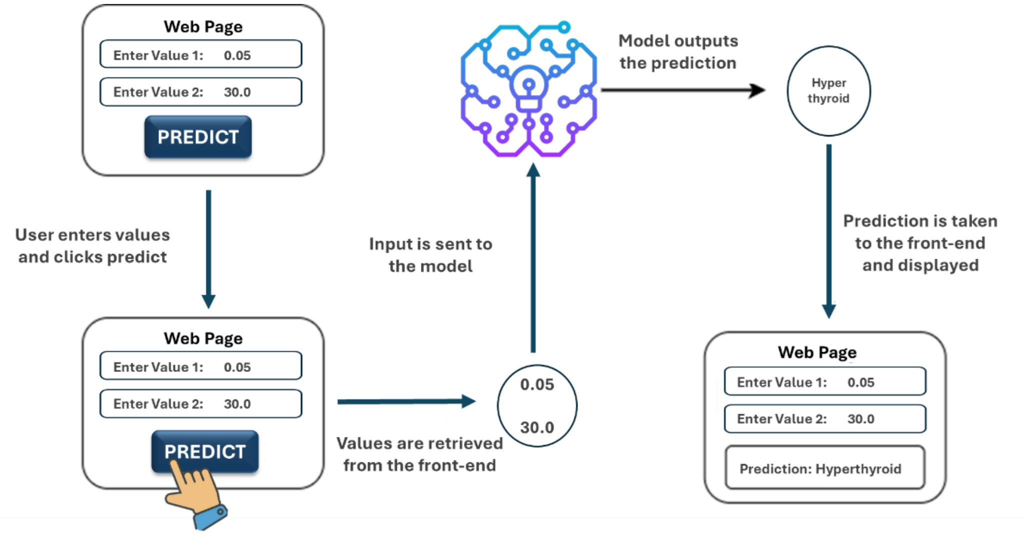


Figure 15: Web Application Architecture

4. IMPLEMENTATION, RESULTS AND DISCUSSIONS

4.1 Implementation

The implementation of the proposed model commenced with configuring all the libraries, modules, and functions in the Visual Studio Code IDE. Subsequently, the data was imported into the programme using the Pandas library. Following the data loading, the exploratory data analysis (EDA) commenced using several Python visualisation packages, including Seaborn, Matplotlib, and the dython function. These tools were employed to analyse and visually represent the correlation between different features within the datasets. During the exploratory data analysis (EDA) phase, univariate, bivariate, and multivariate analyses were conducted to better understand the data's characteristics. These analyses were then visualised to enhance our insights further. The subsequent stages of implementation entailed data preparation for modelling, which encompassed removing outliers, scaling the data, and addressing the problem of data imbalance through resampling techniques such as oversampling and undersampling. It was also crucial to address the data imbalance issue, as it can adversely affect the predictive performance of the models.

The dataset was partitioned into distinct training and testing datasets for modelling without data sampling. Furthermore, additional splitting was performed for modelling the resampled data. Next, the data undergoes training and testing procedures utilizing the five models explained in Chapter Three. Hyperparameter tuning is employed to optimise the model's performance and alleviate the issues related to overfitting. The model's performance is subsequently evaluated using accuracy, precision, recall, F1-score, and ROC metrics. The importance of the most efficient features for the best-performing algorithm is determined and utilized as input data for the web application in predicting and classifying thyroid disease.

The subsequent sections of this chapter present the outcomes attained at different execution phases.

4.2 Results and Discussions

4.2.1 Exploratory Data Analysis

- **Univariate Analysis**

Univariate analysis is an essential component of exploratory data analysis (EDA). It involves statistically examining and visually representing the distribution and variability of a single variable at a time. Graphical representations such as histograms, bar charts, and boxplots are commonly used to visualise univariate insights. An analysis was conducted on the distribution of each numerical and categorical feature in the dataset. The following observations were made based on the graphs:

- Figure 16 displays the histogram distribution of the numerical variables or features in the dataset. The distribution of Age and T4U was found to be approximately normal, with the majority of the data concentrated around the mean at the centre. Nevertheless, TSH, T3, and FTI exhibited a positively skewed distribution, with most data concentrated on the lower values. This suggests that the average value is higher than the median value.
- The categorical features of the dataset are represented in Appendix 2. The graph clearly shows an unequal distribution, as one category of each feature significantly surpasses the other category. The gender distribution appears relatively equitable, with females (F) accounting for 65.57% of the data and males (M) accounting for 34.43%.

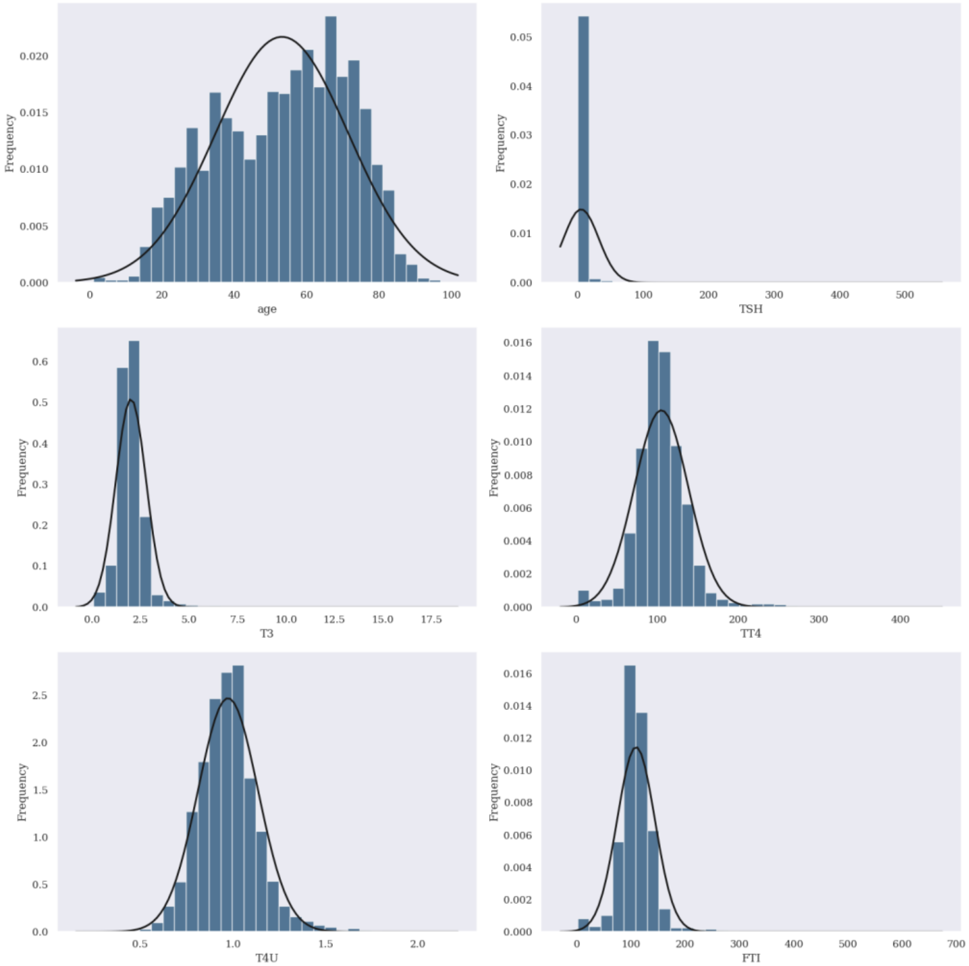


Figure 16: Histplot of Numerical Features

- **Bivariate Analysis**

Bivariate analysis involves the concurrent analysis of two features of a dataset in order to ascertain the relationship or correlation between the two variables. In this study, bivariate analysis was carried out by examining the target variable, which comprises the categories of thyroid disease, against other features in the dataset. The analysis yielded the following summary of insights:

- Figure 17 (a) provides evidence that female patients with thyroid disease outnumber male patients, which supports the statistics mentioned in the introductory chapter of this study.
- Figure 17(b) demonstrates that female patients with either hyperthyroid or hypothyroid disease are not pregnant. This is beneficial as these diseases pose a danger to both the mother and child and necessitate regular monitoring by a health practitioner.
- In Figure 17 (c), a significant proportion of people with hypothyroid and hyperthyroid are not sick. In Figure 17 (d), several patients yet to be tested have hypothyroid disease. This comes as no surprise, as this disease presents hidden symptoms that are not readily apparent or can be misrepresented as symptoms of other diseases.

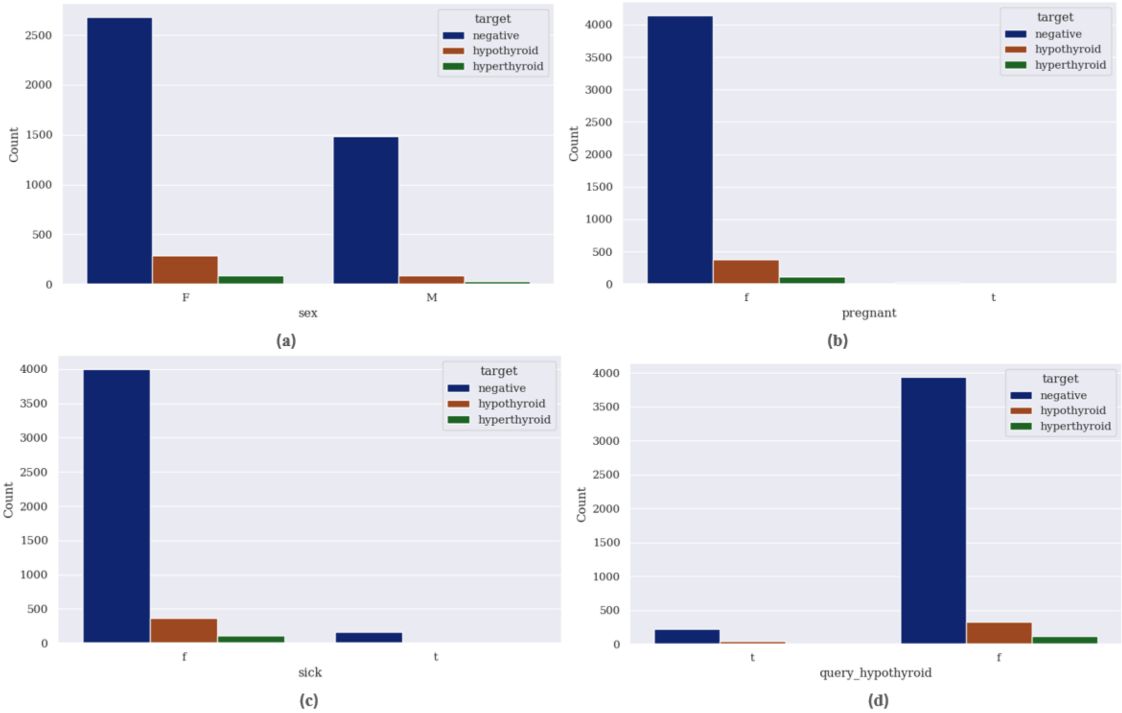


Figure 17: (a) Gender distribution by target. (b) Pregnant distribution by target. (c) Sick distribution by target. (d) Query_hypothyroid distribution by target.

- **Correlation Matrix**

A correlation matrix evaluates relationships between variables in a dataset. A correlation value of 1 indicates a strong relationship between variables, whereas a correlation value of 0 or -1 indicates a neutral or weak relationship. Appendix 3 displays the heatmap correlation matrix illustrating the correlation between the target variable and other variables in the dataset for this study. The hormone tests TSH, T3, T4, and FTI are all strongly correlated with the target variable, with FTI having the most vital relationship with a 0.68 correlation coefficient. As indicated in Chapter 1, hormone tests are more useful in the classification of thyroid disease.

4.2.2 Model Performance

This study employed five classification models to classify thyroid diseases accurately. The models performance are assessed using assessment metrics such as accuracy (ACC), precision (PREC), recall, F1-score, and ROC curve. Due to an imbalance in the dataset, resampling techniques were adopted, and their performance on the models was compared to those without resampling. Hyperparameter tuning was performed on each of the models for enhanced performance. Table II displays the optimal hyperparameters for all the experimental modelling techniques.

Optimal Hyperparameters			
ML Classifiers	**Without Sampling**	**Oversampling**	**Undersampling**

Decision Tree	max_depth = 7,
	min_samples_leaf =1,
	min_samples_split = 2
		max_depth = 7,
		min_samples_leaf = 4,
		min_samples_split = 10
			max_depth = 3,
			min_samples_leaf = 4,
			min_samples_split = 2

SVM	Default	Default	Default
Logistic Regression	Default	Default	Default
KNN	Default	Default	Default

Gradient Boosting	learning_rate = 0.2,
	max_depth = 5,
	min_samples_leaf = 4,
	min_samples_split = 10
		max_depth = 9,
		min_samples_leaf =1,
		min_samples_split = 5
			learning_rate = 0.1,
			max_depth = 5,
			min_samples_leaf = 2,
			min_samples_split = 5

Calibrated	Default	Default	Default
DL Classifier			

MLP	Default	Hidden_layer_size = 100,100, 100
		Max_iter = 1000
			Hidden_layer_size = 100,100, 100
			Max_iter = 1000

Table II: Optimal Hyperparameters for the Different Modelling Technique

- **Model Performance Without Sampling**

The first modelling experiment was the application of classification models to the imbalanced dataset depicted in Figure 18. The classifiers were turned using the optimal hyperparameters for the without sampling technique detailed in Table II above. Table III shows the performance of each classifier model across various evaluation metrics using the identified optimal hyperparameters. Notably, the Decision Tree classifier (DTC) achieved the highest accuracy of 99.46%, surpassing all other classifiers. However, when dealing with imbalanced data, it is advisable not to rely solely on the accuracy score; the F1-score can be used as it considers the mean of precision and recall. DTC remains the highest-performing classifier based on the F1-score at 99.45%.

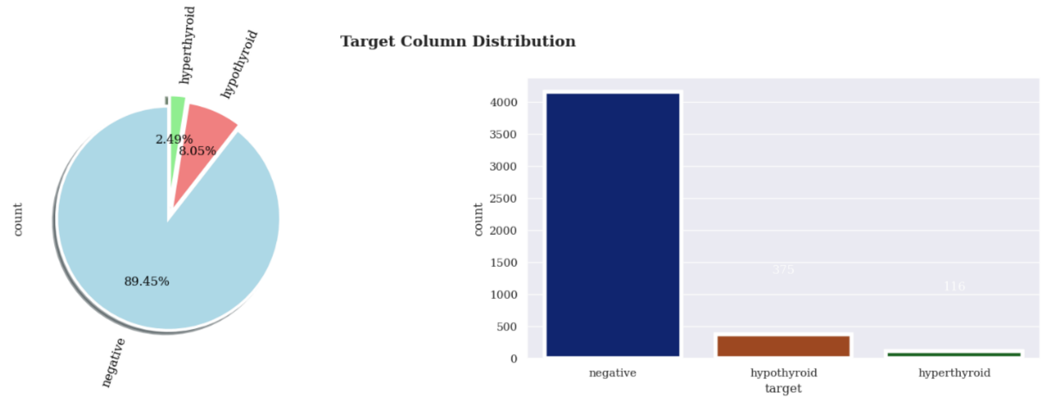


Figure 18: Distribution of Target Class Without Sampling.

ML Classifiers Models	Accuracy	Precision	Recall	F1-Score
Decision Tree	**99.46**	**99.45**	**99.46**	**99.45**
SVM	94.63	94.56	94.63	93.91
Logistic Regression	96.78	96.66	96.78	96.60
KNN	95.70	95.65	95.70	95.29
Gradient Boosting	98.81	98.78	98.81	98.79
DL Classifier				
MLP	97.85	97.87	97.85	97.85

Table III: Classifiers Evaluation Performance without Sampling.

- **Model Performance With Oversampling**

The second modelling experiment involved applying models to the oversampled data. The SMOTE oversampling technique was employed to augment the instances in the minority class of the target variable. The new distribution of the target class after oversampling is depicted in Figure 19. The risk of overfitting brought on by duplication is an inherent limitation of this method. The issue is resolved through the implementation of hyperparameter tuning. Table II above shows the optimal hyperparameters for the classifiers with the oversampling technique. Table IV shows the performance of each classifier model when using the optimal hyperparameters for evaluation. The Gradient Boosting Classifier's (GBC) performance surpasses all other classifiers with an accuracy and F1-score of 99.76%.

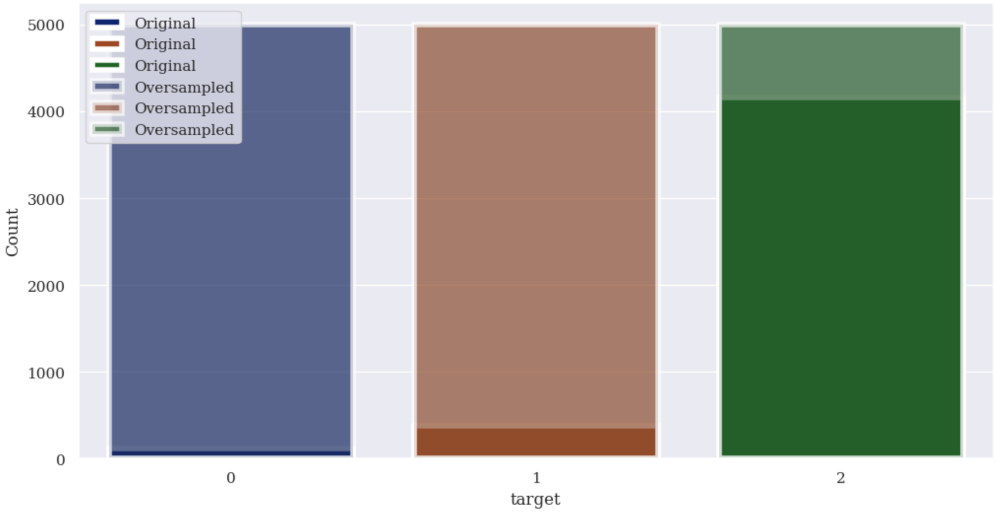


Figure 19: Distribution of Target Class With Oversampling.

ML Classifiers	Accuracy	Precision	Recall	F1-Score
Decision Tree	99.63	99.63	99.63	99.63
SVM	97.83	97.85	97.83	97.83
Logistic Regression	98.76	98.76	98.76	98.76
KNN	94.60	94.66	94.60	94.58
Gradient Boosting	**99.76**	**99.76**	**99.76**	**99.76**
DL Classifier				
MLP	99.40	99.39	99.40	99.39

Table IV: Classifiers Evaluation Performance with Oversampling.

- **Model Performance With Undersampling**

In this last modelling experiment, classification models were applied to the undersampled data. The random undersampling technique was utilized to decrease the number of instances in the majority class—the new distribution of the target class after undersampling is shown in Figure 20. The optimal hyperparameters for the classifiers with the undersampling technique are detailed in Table II above. The performance of each classifier model, measured by all evaluation metrics using the identified optimal hyperparameter, is presented in Table V below. Once again, the Gradient Boosting Classifier (GBC) outperforms all other classifiers, achieving the highest accuracy and F1-score of 99.14% and 99.13%, respectively.

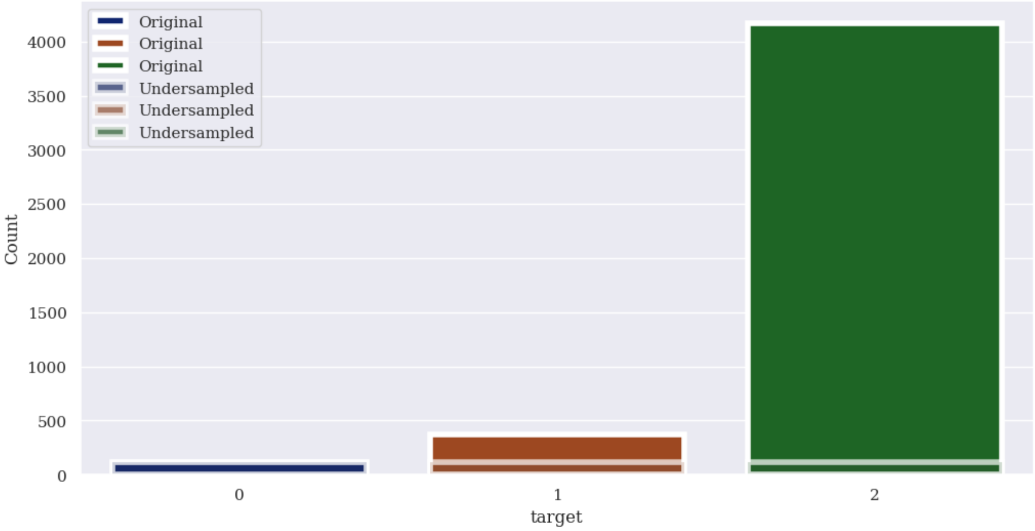


Figure 20: Distribution of Target Class With Undersampling.

ML Classifiers	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Decision Tree	95.71	95.71	95.71	95.68
SVM	74.28	78.04	74.28	74.30
Logistic Regression	84.28	87.35	84.28	84.70
KNN	72.85	76.47	72.85	73.23
Gradient Boosting	**97.14**	**97.19**	**97.14**	**97.13**
DL Classifier				
MLP	84.28	84.85	84.28	84.49

Table V: Classifiers Evaluation Performance with Undersampling.

- **Performance Comparison of Models Using Different Sampling Techniques**

In evaluating the models for classifying thyroid diseases using different sampling techniques, the evaluation metrics considered are Accuracy and F1-Score. Table VI below summarises each classifier model's performance across various sampling techniques based on these two evaluation metrics. As indicated in Table VI and illustrated in Figures 21 and 22, the findings highlight better performance when using the oversampling technique, with Gradient Boosting attaining the highest score of 99.67%. However, an exception in performance is observed for the KNN model, where both the accuracy score and F1-Score are slightly higher than those achieved with the oversampling technique. However, the performance derived from the without sampling technique is unreliable as it is based on a skewed dataset. In the undersampling technique, most classifier models exhibited poor performance due to insufficient samples for effective learning during the training process.

	Without Sampling		**With Oversampling**		**With Undersampling**	
Classifiers	**Accuracy**	**F1-Score**	**Accuracy**	**F1-Score**	**Accuracy**	**F1-Score**
Decision Tree	99.46	99.45	99.63	99.63	95.71	95.68
SVM	94.63	93.91	97.83	97.83	74.28	74.30
Logistic Regression	96.78	96.60	98.76	98.76	84.28	84.70
KNN	95.70	95.29	94.60	94.58	72.85	73.23
Gradient Boost	98.81	98.79	99.76	99.76	97.14	97.13
MLP	97.85	97.85	99.40	99.30	84.24	84.49

Table VII: Model Comparison For Sampling Techniques

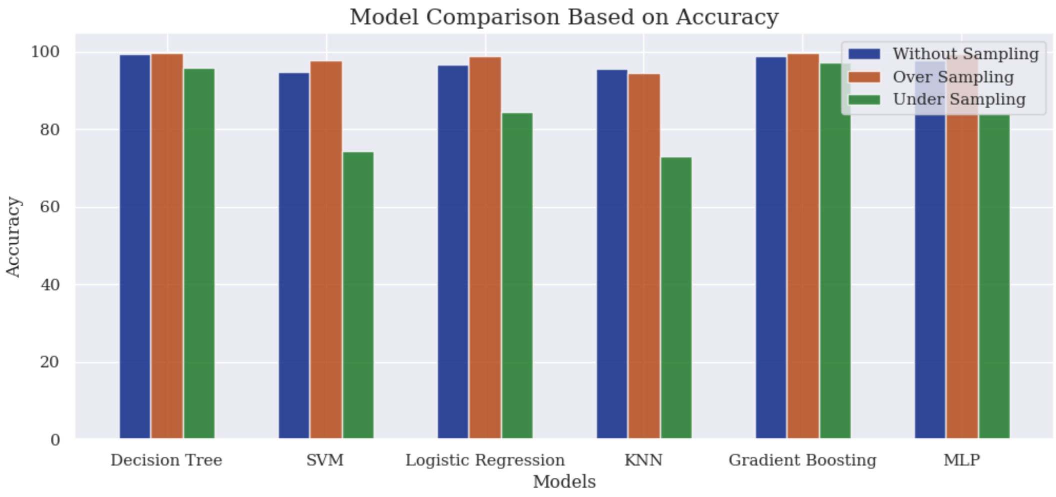


Figure 21: Model Comparison Based on Accuracy Score.

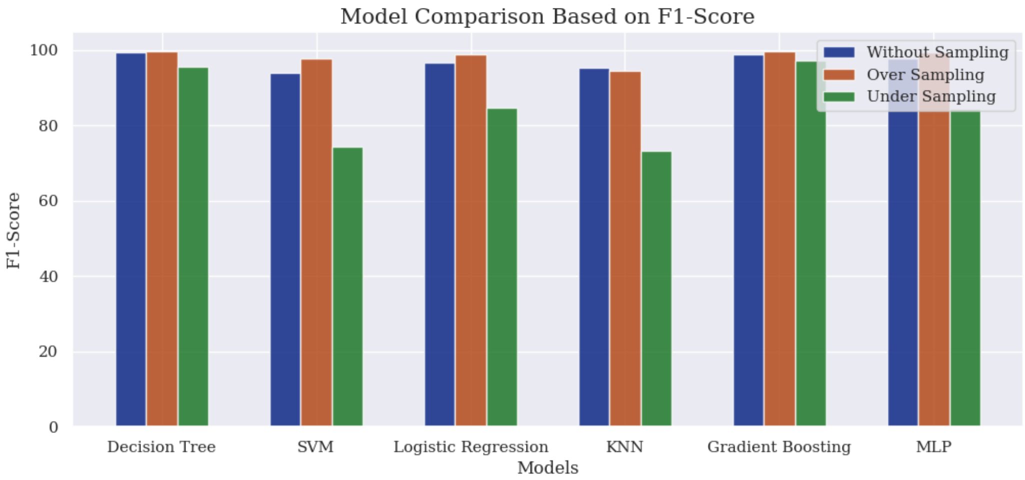


Figure 22: Models Comparison Based on F1-Score

4.2.3 Web Application Performance

The web application accurately classified the different classes of thyroid diseases. Figure 23 shows the web application's start page, while Figure 24 depicts the index page for input data. Figures 25 and 26 illustrates the different classification outcomes for "No thyroid," and "Hyperthyroid."

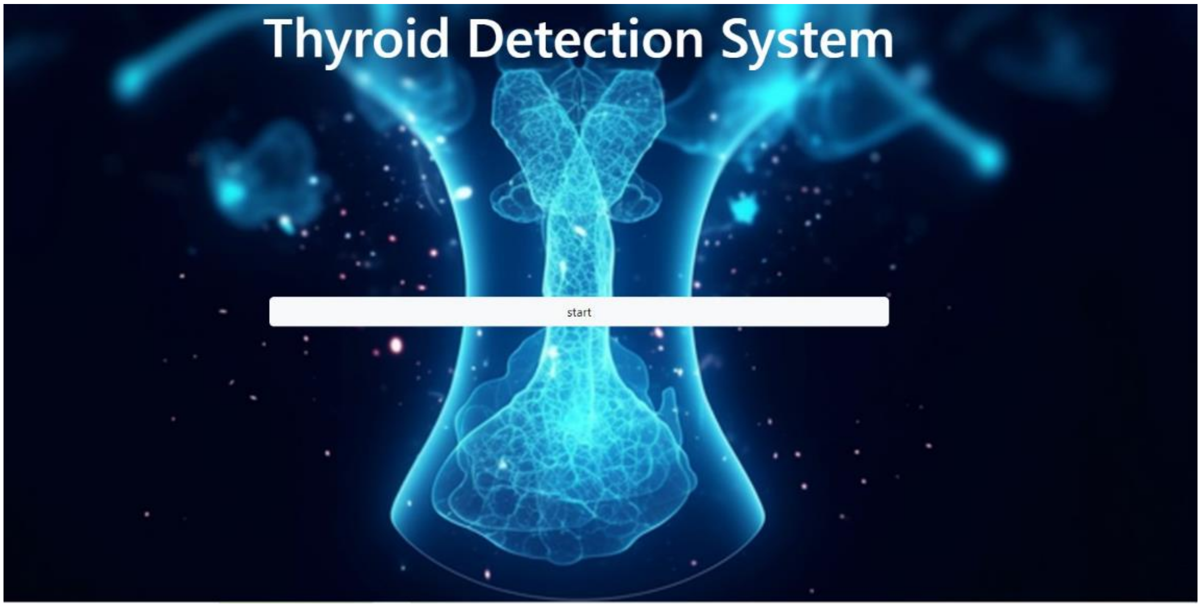
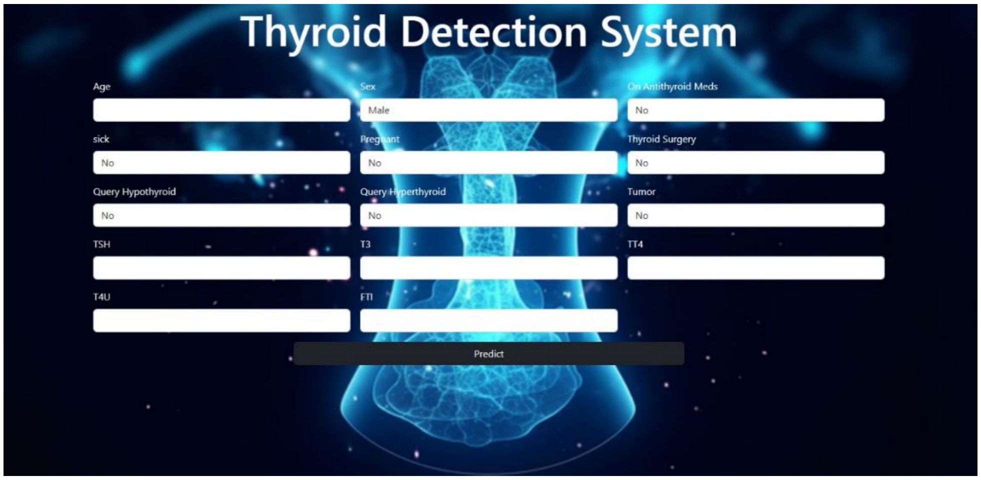


Figure 23: Web App Start Page


Thyroid Detection System

Age:

Sex: Male

On Antithyroid Meds: No

sick: No

Pregnant: No

Thyroid Surgery: No

Query Hypothyroid: No

Query Hyperthyroid: No

Tumor: No

TSH:

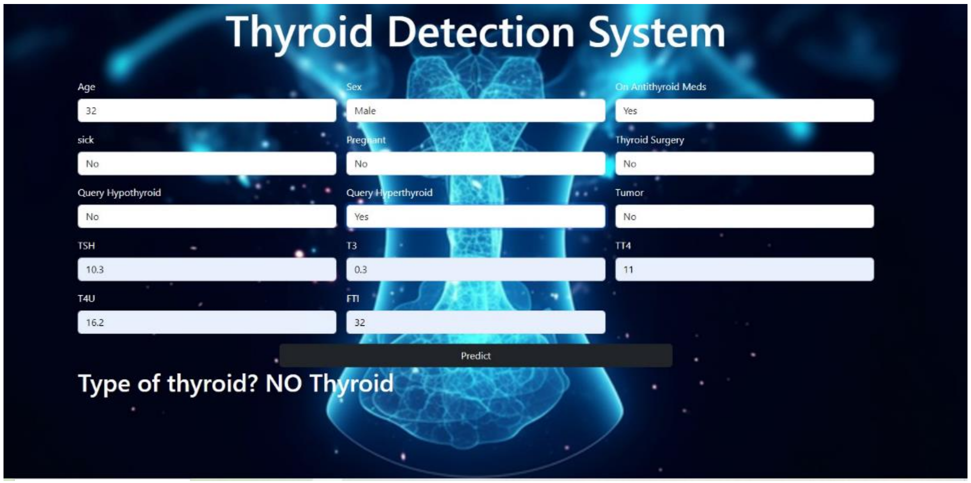
T3:

TT4:

T4U:

FTI:

Figure 24: Web App Index Page For Data Input.


Thyroid Detection System

Age: 32

Sex: Male

On Antithyroid Meds: Yes

sick: No

Pregnant: No

Thyroid Surgery: No

Query Hypothyroid: No

Query Hyperthyroid: Yes

Tumor: No

TSH: 10.3

T3: 0.3

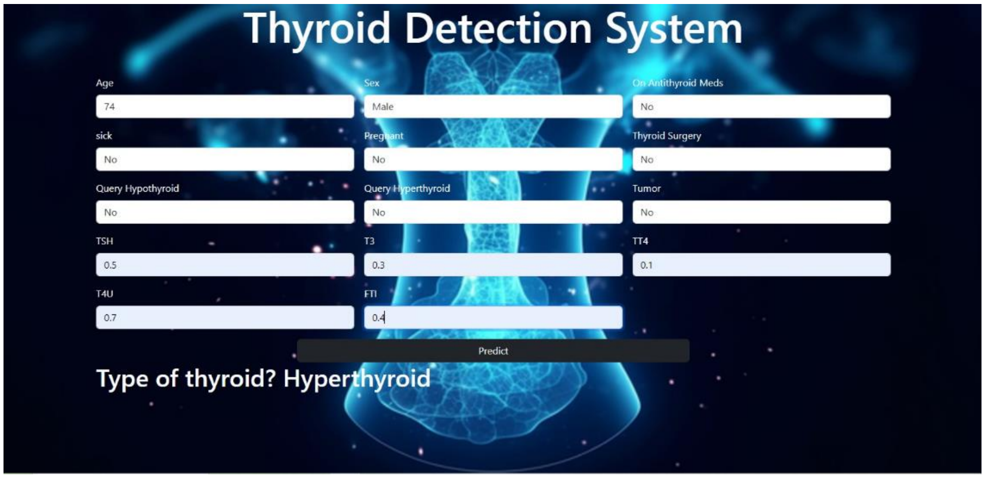
TT4: 11

T4U: 16.2

FTI: 32

Type of thyroid? NO Thyroid

Figure 25: Screenshot of Web App Classification of No thyroid.


Thyroid Detection System

Age: 74

Sex: Male

On Antithyroid Meds: No

sick: No

Pregnant: No

Thyroid Surgery: No

Query Hypothyroid: No

Query Hyperthyroid: No

Tumor: No

TSH: 0.5

T3: 0.3

TT4: 0.1

T4U: 0.7

FTI: 0.4

Type of thyroid? Hyperthyroid

Figure 26: Screenshot of Web App Classification of Hyperthyroid.

4.3 Research Questions Discussion

Research Question 1

Which biomarkers are the most significant contributors to classifying different types of thyroid diseases?

Answer: Biomarkers are measurable indications that offer insights into distinct biological states or conditions and are vital in categorizing various thyroid illnesses (Strimbu & Tavel, 2010). The dataset examined in this study comprises five key biomarkers: Thyroid Stimulating Hormone (TSH), Triiodothyronine (T3), Total Thyroxine (TT4), Thyroxine Utilization (T4U), and Free Thyroxine Index (FTI).

The analysis of the feature importance chart in Figure 27 demonstrates that TSH is the most relevant biomarker in classifying various thyroid diseases, with a substantial importance value of 0.78. This observation aligns with the research findings of Duggal et al. (2020) and Savi and Nuriyeva (2022), emphasizing the increasing significance of the TSH test as the primary factor in laboratory thyroid function testing for classifying thyroid diseases.

In addition, the feature importance chart indicates that FTI and T3 substantially impacted the categorization of thyroid illnesses, with importance values of about 0.15 and 0.08, respectively. These findings also highlight the relevance of FTI and T3 in understanding and differentiating thyroid conditions, thereby enhancing the diagnostic procedure.

Recognising the most significant biomarkers is crucial for expediting the diagnostic process for healthcare professionals. Equipped with this information, healthcare practitioners can precisely select the appropriate tests and examinations, resulting in more efficient and accurate diagnoses. Ultimately, this focused strategy improves patient outcomes by enabling rapid and personalised medical interventions.

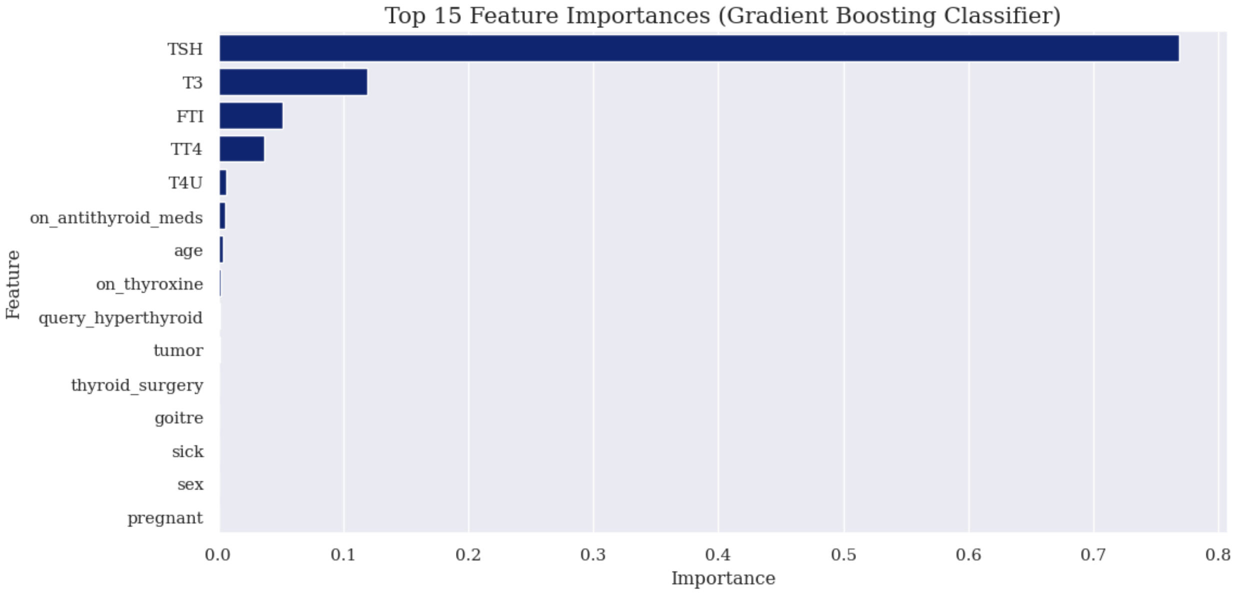


Figure 27: Feature Importance Graph Based on Gradient Boosting Classifier.

Research Question 2

How do resampling techniques impact models' performance in accurately classifying thyroid diseases?

Answer: First, resampling techniques foster unbiased model performance, especially when dealing with imbalanced datasets, as in this study. Applying models to an imbalanced dataset will yield mistaken classification performance, as the model will only perform appropriately on the class with a more significant number of instances while performing poorly on the class with fewer instances.

SMOTE oversampling and random undersampling techniques were used to balance the dataset's target variable. Analysis of Table VI above revealed that the oversampling technique yielded superior results compared to the undersampling technique in terms of accuracy and F1-Score for all classifiers. This is because synthetic instances of the target class have been generated, enhancing the learning process of models and enabling them to make precise and dependable categorizations of thyroid disease. Undersampling results in data loss, which leads to fewer instances for the model to learn from, thereby impeding its performance. Modelling without resampling technique may have exhibited good accuracy and F1-score, but it cannot be considered reliable since the models were primarily learned from the majority class during training.

Another impact of resampling techniques on the model's performance is that it enhances the model's generality. This implies that the model not only learned from the instances of the dominant class but also from those of the lesser class. Thus, improving the model's performance on unfamiliar data and mitigating the problem of overfitting during training is crucial when developing models for real-world applications.

In summary, resampling techniques aid in mitigating bias and enhancing the generality of models when classifying various types of thyroid disease, resulting in improved and dependable model performance.

Research Question 3

How does the integration of a machine learning model into a web-based application enhance the prompt classification of thyroid disease?

Answer: Firstly, the analysis performed by the web app is automated, utilizing the machine learning model that has been trained to recognise patterns related to the different types of thyroid diseases. As a result, the classification process is faster than the manual analysis of laboratory test results. The model quickly analyses the input data, detects relevant patterns, and delivers prompt feedback. The reduction in processing time allows healthcare practitioners to make faster and more informed decisions when treating patients with thyroid conditions.

Secondly, the web app eliminates human lapses in accurately classifying thyroid diseases, reducing delays and the need to re-evaluate test results. The integrated machine learning model is able to eliminate classification errors by leveraging its training to make accurate and consistent predictions based on the recognition of learned patterns. This enhances the efficiency of the classification process.

Thirdly, the web app has a friendly interface that aids effortless navigation and lets users enter necessary data from the front end. The machine learning model efficiently processes the inputted data at the backend, which provides transparent feedback on the front end. The user-friendly interface promotes regular utilization of the web application, thereby delivering expedited outcomes.

Finally, the web-based applications can be accessed remotely from any location, alleviating the need to send test results to specialised physicians for analysis. The fact that healthcare practitioners can access the web app from different locations facilitates prompt diagnosis without being limited by geographic constraints.

5. CONCLUSION

5.1 Research Summary

This research utilized a range of machine learning models, including a deep learning model, to accurately classify three thyroid disease classes: No thyroid, hyperthyroid, and hypothyroid. Resampling techniques were employed to tackle the problem of class imbalance, and it was found that the SMOTE oversampling technique was the most effective. The experimental results proved that the Gradient Boosting Classifier (GBC) utilizing the oversampling technique achieved superior performance compared to other classifier models, with an accuracy and F1-score of 99.76%. Although the performance of models utilizing the without Sampling technique appeared satisfactory, it may yield deceptive outcomes due to modelling on an imbalanced dataset. The utilization of the undersampling technique resulted in poor outcomes across the majority of the models. The study further presents the deployment of the best-performing model into a web-based application for thyroid disease classification. This deployment aims to offer users an attractive interface and efficient diagnostics. The study also identified Thyroid-stimulating hormone (TSH) as the most significant biomarker, providing valuable insights into thyroid diseases and emphasising the importance of TSH in diagnosis. Overall, this study presented valuable insights that can assist healthcare practitioners in promptly diagnosing thyroid diseases.

5.2 Limitations

The insights gained from the models' performance and this research's findings have been valuable. However, a few limitations require consideration and are crucial to the robustness of the proposed model. Firstly, the dataset used was relatively small in size and characterized by a high degree of imbalance. Another constraint is the unequal representation of categorical variables compared to continuous variables in the dataset. Furthermore, the diagnosis of thyroid disease typically involves a comprehensive assessment, including various clinical, pathological, and serological characteristics. This could comprise thyroid function tests to quantify hormone levels, ultrasound to ascertain the size and texture of the gland, and an examination of clinical symptoms like fluctuations in weight, tolerance to heat, irregularities in heart rate, BMI, and more. Additionally, antibody tests are frequently utilized to detect the existence of antibodies that might affect the thyroid gland. However, the dataset lacks essential information about most of these diagnostic components, limiting the proposed model's robustness.

5.3 Future Work

While this research has effectively met the study's objectives and performed adequately, there are still opportunities to broaden its scope. Potential areas for future endeavors involve incorporating Super Learners (SL) with the resampling techniques utilized in this research and evaluating their performance against individual models using the same metrics employed in this study. Additionally, the web app interface will be enhanced by introducing an option for users to upload test results in addition to manual input. This would provide increased flexibility and convenience for users. Lastly, the proposed model will be extended to a medical image dataset, thereby enhancing the model's robustness and application.

REFERNCES

Abbad Ur Rehman, H., Lin, C. and Mushtaq, Z. (2021) 'Effective K-Nearest Neighbor Algorithms Performance Analysis of Thyroid Disease', *Journal of the Chinese Institute of Engineers,* 44(1), pp. 77-87. Available at: https://doi.org/10.1080/02533839.2020.1831967

Akash, K.T. *et al.* (2023) 'Predicting Thyroid Dysfunction Using Machine Learning Techniques', *2023 12th International Conference on Advanced Computing (ICoAC).* IEEE

Alexandropoulos, S.N., Kotsiantis, S.B. and Vrahatis, M.N. (2019) 'Data preprocessing in predictive data mining', *The Knowledge Engineering Review,* 34, pp. e1. Available at: https://doi.org/10.1017/S026988891800036X(Accessed: 2023/11/30)

Alyas, T. *et al.* (2022) 'Empirical Method for Thyroid Disease Classification Using a Machine Learning Approach', *BioMed Research International,* 2022, pp. 1-10. Available at: https://doi.org/10.1155/2022/9809932

Aswathi, A.K. and Antony, A. (2018) 'An Intelligent System for Thyroid Disease Classification and Diagnosis', IEEE Available at: 10.1109/ICICCT.2018.8473349.

Bahaj, A.S. *et al.* (2020) 'Role of fine-needle aspiration cytology in evaluating thyroid nodules. A retrospective study from a tertiary care center of Western region, Saudi Arabia', *Saudi Medical Journal,* 41(10), pp. 1098-1103. Available at: https://doi.org/10.15537/SMJ.2020.10.25417

Balasree, K. and Dharmarajan, K. (2023) 'Thyroid classification using Deep Learning Techniques', IEEE Available at: 10.1109/ICPCSN58827.2023.00145.

Bonjoc, K. *et al.* (2020) 'Thyroid cancer diagnosis in the era of precision imaging', *Journal of Thoracic Disease,* 12(9), pp. 5128.

Borzouei, S. *et al.* (2020) 'Diagnosing thyroid disorders: Comparison of logistic regression and neural network models', *Journal of Family Medicine and Primary Care,* 9(3), pp. 1470.

Brereton, R.G. and Lloyd, G.R. (2010) 'Support vector machines for classification and regression', *Analyst,* 135(2), pp. 230-267.

Breu, F., Guggenbichler, S. and Wollmann, J. 'No title', *Utilization of Neural Network for Disease Forecasting.Vasa [Internet].2008,*

Brindha, V. and Muthukumaravel, A. (2023) 'Efficient Method for Predicting Thyroid Disease Classification using Convolutional Neural Network with Support Vector Machine', in 'Efficient Method for Predicting Thyroid Disease Classification using Convolutional Neural Network with Support Vector Machine', *Computational Intelligence for Clinical Diagnosis.* Springer, pp. 77-85.

Brownlee, J. (2020) *Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python.* Machine Learning Mastery.

Charbuty, B. and Abdulazeez, A. (2021) 'Classification based on decision tree algorithm for machine learning', *Journal of Applied Science and Technology Trends,* 2(01), pp. 20-28.

Chen, D. *et al.* (2020) 'Diagnosis of thyroid nodules for ultrasonographic characteristics indicative of malignancy using random forest', *BioData Mining,* 13(1), pp. 14. Available at: https://doi.org/10.1186/s13040-020-00223-w

Chiasera, J.M. (2013) 'Back to the basics: thyroid gland structure, function and pathology', *Clinical Laboratory Science,* 26(2), pp. 112.

Cutler, A., Cutler, D.R. and Stevens, J.R. (2012) 'Random forests', *Ensemble Machine Learning: Methods and Applications,* , pp. 157-175.

Deng, J. *et al.* (2014) 'Comparison of diagnostic efficacy of contrast-enhanced ultrasound, acoustic radiation force impulse imaging, and their combined use in differentiating focal solid thyroid nodules', *PloS One,* 9(3), pp. e90674. Available at: https://doi.org/10.1371/journal.pone.0090674

Dixit, R. *et al.* (2023) 'Thyroid Disorder Classification using Machine Learning', IEEE Available at: 10.1109/ICETET-SIP58143.2023.10151522.

Dogantekin, E., Dogantekin, A. and Avci, D. (2011) 'An expert system based on Generalized Discriminant Analysis and Wavelet Support Vector Machine for diagnosis of thyroid diseases', *Expert Systems with Applications,* 38(1), pp. 146-150. Available at: https://doi.org/10.1016/j.eswa.2010.06.029

Duggal, P. and Shukla, S. (2020) 'Prediction Of Thyroid Disorders Using Advanced Machine Learning Techniques', IEEE Available at: 10.1109/Confluence47617.2020.9058102.

Farling, P.A. (2000) 'Thyroid disease', *British Journal of Anaesthesia*, 85(1), pp. 15-28.

Freitas, P.A.V.C.J. *et al.* (2016) 'STUDY OF THE PREVALENCE OF AUTOIMMUNE THYROID DISEASE IN WOMEN WITH BREAST CANCER', *Endocrine Practice,* 22(1), pp. 16-21. Available at: https://doi.org/10.4158/EP14445.OR

Gosain, M., Gupta, S. and Kaur, S. (2022) 'Machine and Deep Learning Techniques to Classify and Predict Thyroid Diseases', IEEE Available at: 10.1109/ICIEM54221.2022.9853067.

Gou, J. *et al.* (2012) 'A new distance-weighted k-nearest neighbor classifier', *J.Inf.Comput.Sci,* 9(6), pp. 1429-1436.

Islam, S.S. *et al.* (2022) 'No title', *Application of Machine Learning Algorithms to Predict the Thyroid Disease Risk: An Experimental Comparative Study.PeerJ Comput Sci,*

Koulouri, O. *et al.* (2013) 'Pitfalls in the measurement and interpretation of thyroid function tests', *Best Practice & Research Clinical Endocrinology & Metabolism,* 27(6), pp. 745-762. Available at: https://doi.org/10.1016/j.beem.2013.10.003

Kumar, R.R. *et al.* (2023) 'Thyroid Disease Classification using Machine Learning Algorithms', *E3S Web of Conferences.* EDP Sciences

Mollica, G. *et al.* (2022) 'Classification of Thyroid Diseases Using Machine Learning and Bayesian Graph Algorithms', *IFAC PapersOnLine,* 55(40), pp. 67-72. Available at: https://doi.org/10.1016/j.ifacol.2023.01.050

Montagna, C. and Zangelidis, A. (2023) 'People's Experience with Thyroid Disease: Survey Report', *People's Experience with Thyroid Disease,*

N. Ananthi *et al.* (2022) 'Detecting six different types of Thyroid Diseases using Deep Learning approaches', *2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI).* Available at: 10.1109/ACCAI53970.2022.9752581.

Nguyen, Q.T. *et al.* (2015) 'Diagnosis and treatment of patients with thyroid cancer', *American Health & Drug Benefits,* 8(1), pp. 30.

Nugroho, A. *et al.* (2023) 'Web based application system for cancerous object detection in ultrasound images', *AIP Conference Proceedings.* AIP Publishing

Olatunji, S.O. *et al.* (2021) 'Early diagnosis of thyroid cancer diseases using computational intelligence techniques: A case study of a Saudi Arabian dataset', *Computers in Biology and Medicine,* 131, pp. 104267. Available at: https://doi.org/10.1016/j.compbiomed.2021.104267

Q. Pan *et al.* (2016) 'Improved Ensemble Classification Method of Thyroid Disease Based on Random Forest', *2016 8th International Conference on Information Technology in Medicine and Education (ITME).* Available at: 10.1109/ITME.2016.0134.

R. Mohammed, J. Rawashdeh and M. Abdullah (2020) 'Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results', *2020 11th International Conference on Information and Communication Systems (ICICS).* Available at: 10.1109/ICICS49469.2020.239556.

Rao, A.R. and Renuka, B.S. (2020) 'A Machine Learning Approach to Predict Thyroid Disease at Early Stages of Diagnosis', IEEE Available at: 10.1109/INOCON50539.2020.9298252.

S. Balasubramanian, V. Srinivasan and A. Thomo (2022) 'Identifying Important Features for Clinical Diagnosis of Thyroid Disorder', *2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).* Available at: 10.1109/ASONAM55673.2022.10068712.

Saini, A., Guleria, K. and Sharma, S. (2023) 'Machine Learning Approaches for Early Identification of Thyroid Disease', IEEE Available at: 10.1109/WCONF58270.2023.10235086.

Sankar, S. *et al.* (2022) 'Thyroid Disease Prediction Using XGBoost Algorithms', *Journal of Mobile Multimedia,* 18(3), pp. 917-934. Available at: https://doi.org/10.13052/jmm1550-4646.18322

Savcı, E. and Nuriyeva, F. (2022) 'DIAGNOSIS OF THYROID DISEASE USING MACHINE LEARNING TECHNIQUES.', *Journal of Modern Technology & Engineering,* 7(2)

Scholkopf, B. (1998) 'Support vector machines: A practical consequence of learning theory', *IEEE Intelligent Systems,* 13

Shahid, A.H. *et al.* (2019) 'A Study on Label TSH, T3, T4U, TT4, FTI in Hyperthyroidism and Hypothyroidism using Machine Learning Techniques', IEEE Available at: 10.1109/ICCES45898.2019.9002284.

Srivastava, R. and Kumar, P. (2021) 'BL_SMOTE ensemble method for prediction of thyroid disease on imbalanced classification problem', *Proceedings of Second International Conference on Computing, Communications, and Cyber-Security: IC4S 2020.* Springer

Strimbu, K. and Tavel, J.A. (2010) 'What are biomarkers?', *Current Opinion in HIV and AIDS,* 5(6), pp. 463.

Swain, P.H. and Hauska, H. (1977) 'The decision tree classifier: Design and potential', *IEEE Transactions on Geoscience Electronics,* 15(3), pp. 142-147.

Tabassum, S., Rumky, S.F.F. and Shahariar, M.F. (2022) 'No title', *Thyroid Disease Analysis and Prediction by using Machine Learning and Deep Learning: A Comparative Approach,*

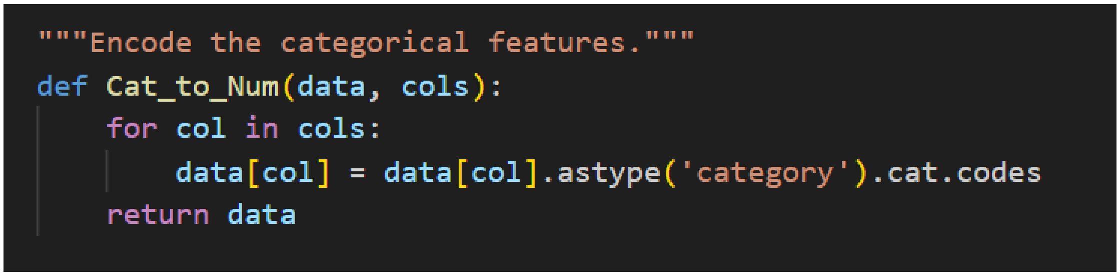
Vanderpump, M.P. (2011) 'The epidemiology of thyroid disease.', *British Medical Bulletin,* 99(1)

Vasan, C.R.C. *et al.* (2018) 'Thyroid detection using machine learning', *Computing (PDGC-2018),* 20, pp. 22.

Vijayvargiya, A. *et al.* (2021) 'Human knee abnormality detection from imbalanced sEMG data', *Biomedical Signal Processing and Control,* 66, pp. 102406.

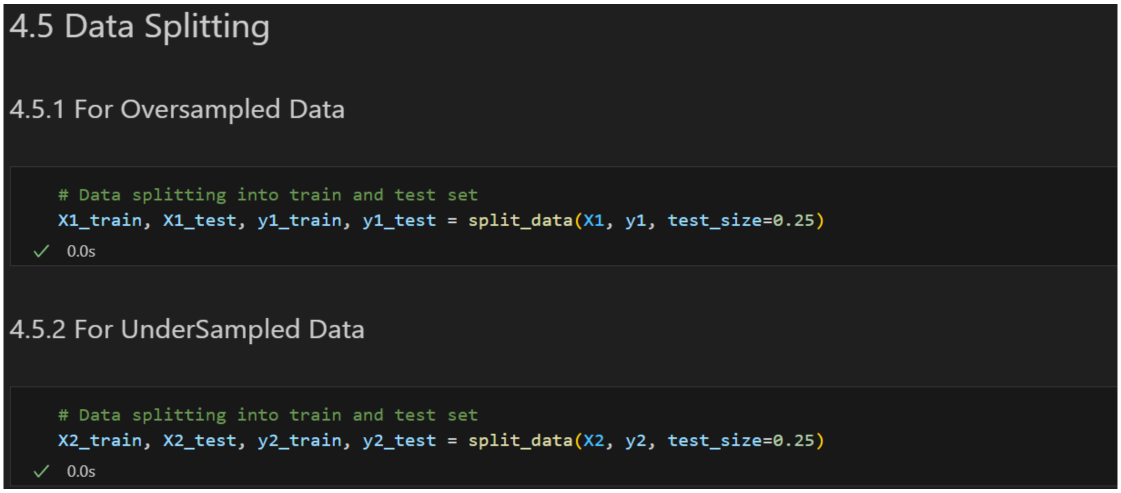
Zhang, T. *et al.* (2021) 'Improving convection trigger functions in deep convective parameterization schemes using machine learning', *Journal of Advances in Modeling Earth Systems,* 13(5), pp. e2020MS002365.

APPENDICES



```
"""Encode the categorical features."""
def Cat_to_Num(data, cols):
    for col in cols:
        data[col] = data[col].astype('category').cat.codes
    return data
```

Appendix 1: Code For Encoding Categorical Variables.



4.5 Data Splitting

4.5.1 For Oversampled Data

```
# Data splitting into train and test set
X1_train, X1_test, y1_train, y1_test = split_data(X1, y1, test_size=0.25)
```

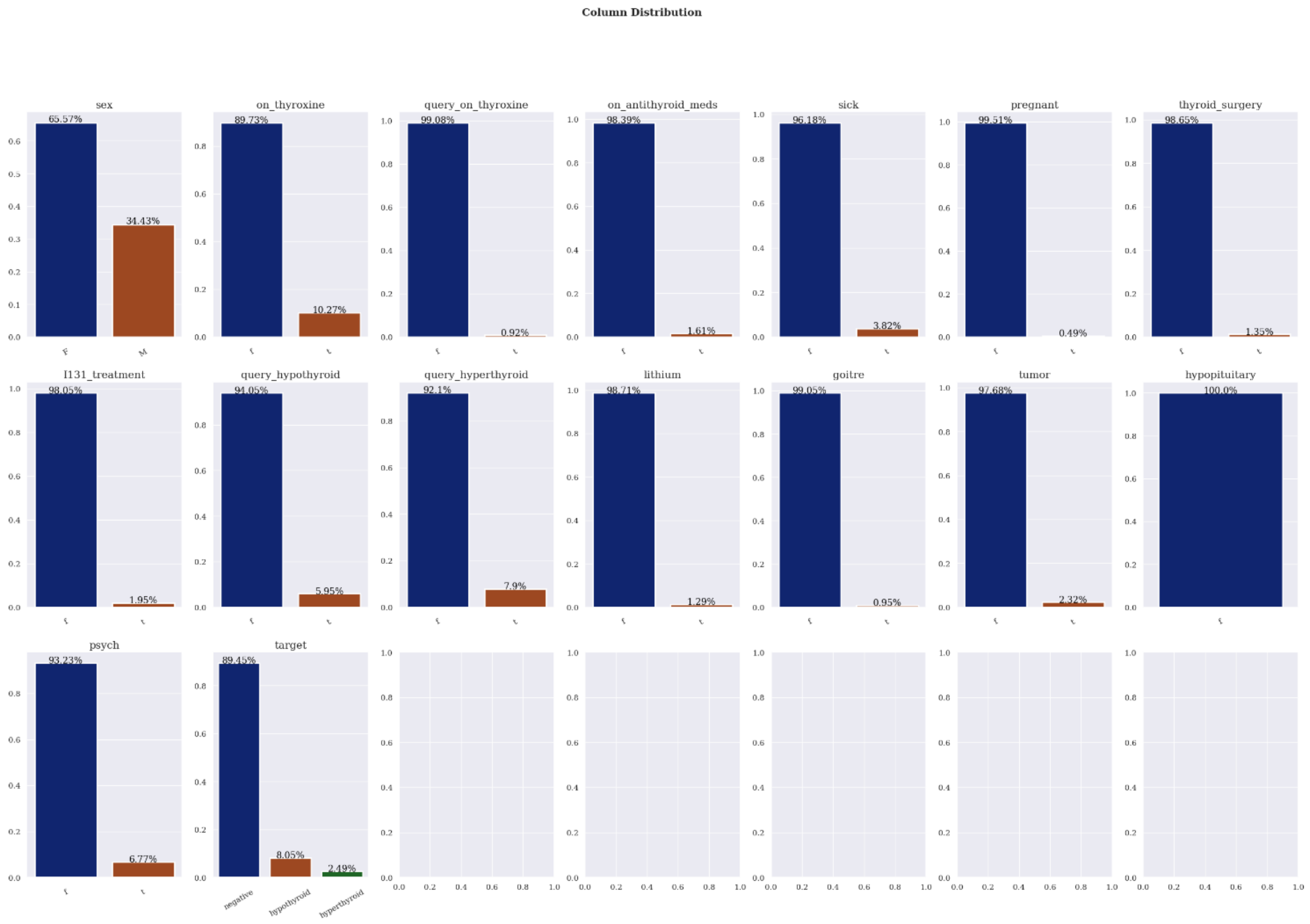
✓ 0.0s

4.5.2 For UnderSampled Data

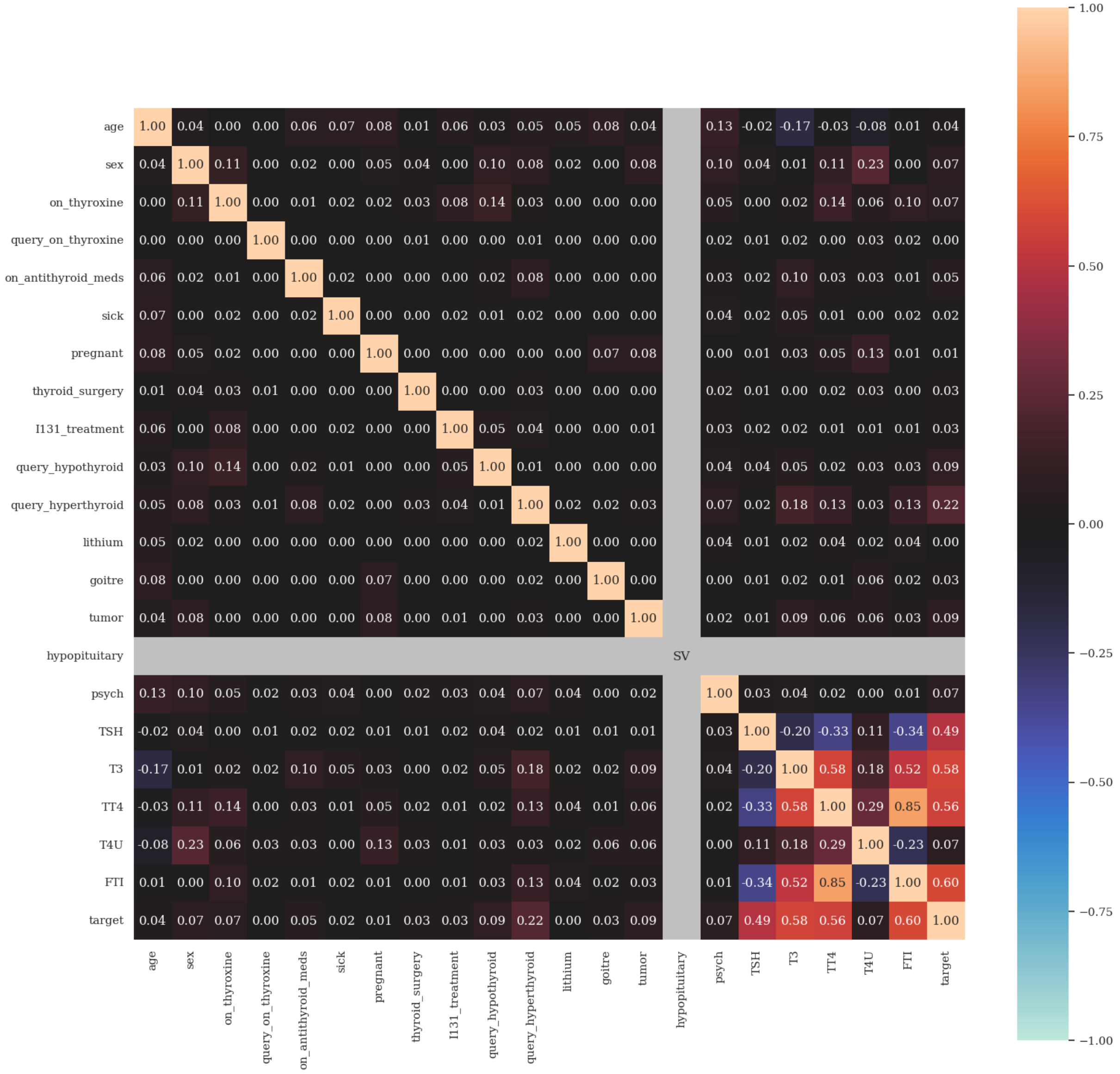
```
# Data splitting into train and test set
X2_train, X2_test, y2_train, y2_test = split_data(X2, y2, test_size=0.25)
```

✓ 0.0s

Appendix 2: Data Splitting Code for Oversampled and Unsampled data.



Appendix 3: Plot of All Categorical Variables.



	age	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_meds	sick	pregnant	thyroid_surgery	I131_treatment	query_hypothyroid	query_hyperthyroid	lithium	goitre	tumor	hypopituitary	psych	TSH	T3	TT4	T4U	FTI	target
age	1.00	0.04	0.00	0.00	0.06	0.07	0.08	0.01	0.06	0.03	0.05	0.05	0.08	0.04		0.13	-0.02	-0.17	-0.03	-0.08	0.01	0.04
sex	0.04	1.00	0.11	0.00	0.02	0.00	0.05	0.04	0.00	0.10	0.08	0.02	0.00	0.08		0.10	0.04	0.01	0.11	0.23	0.00	0.07
on_thyroxine	0.00	0.11	1.00	0.00	0.01	0.02	0.02	0.03	0.08	0.14	0.03	0.00	0.00	0.00		0.05	0.00	0.02	0.14	0.06	0.10	0.07
query_on_thyroxine	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00		0.02	0.01	0.02	0.00	0.03	0.02	0.00
on_antithyroid_meds	0.06	0.02	0.01	0.00	1.00	0.02	0.00	0.00	0.00	0.02	0.08	0.00	0.00	0.00		0.03	0.02	0.10	0.03	0.03	0.01	0.05
sick	0.07	0.00	0.02	0.00	0.02	1.00	0.00	0.00	0.02	0.01	0.02	0.00	0.00	0.00		0.04	0.02	0.05	0.01	0.00	0.02	0.02
pregnant	0.08	0.05	0.02	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.07	0.08		0.00	0.01	0.03	0.05	0.13	0.01	0.01
thyroid_surgery	0.01	0.04	0.03	0.01	0.00	0.00	0.00	1.00	0.00	0.00	0.03	0.00	0.00	0.00		0.02	0.01	0.00	0.02	0.03	0.00	0.03
I131_treatment	0.06	0.00	0.08	0.00	0.00	0.02	0.00	0.00	1.00	0.05	0.04	0.00	0.00	0.01		0.03	0.02	0.02	0.01	0.01	0.01	0.03
query_hypothyroid	0.03	0.10	0.14	0.00	0.02	0.01	0.00	0.00	0.05	1.00	0.01	0.00	0.00	0.00		0.04	0.04	0.05	0.02	0.03	0.03	0.09
query_hyperthyroid	0.05	0.08	0.03	0.01	0.08	0.02	0.00	0.03	0.04	0.01	1.00	0.02	0.02	0.03		0.07	0.02	0.18	0.13	0.03	0.13	0.22
lithium	0.05	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	1.00	0.00	0.00		0.04	0.01	0.02	0.04	0.02	0.04	0.00
goitre	0.08	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.02	0.00	1.00	0.00		0.00	0.01	0.02	0.01	0.06	0.02	0.03
tumor	0.04	0.08	0.00	0.00	0.00	0.00	0.08	0.00	0.01	0.00	0.03	0.00	0.00	1.00		0.02	0.01	0.09	0.06	0.06	0.03	0.09
hypopituitary															SV							
psych	0.13	0.10	0.05	0.02	0.03	0.04	0.00	0.02	0.03	0.04	0.07	0.04	0.00	0.02		1.00	0.03	0.04	0.02	0.00	0.01	0.07
TSH	-0.02	0.04	0.00	0.01	0.02	0.02	0.01	0.01	0.02	0.04	0.02	0.01	0.01	0.01		0.03	1.00	-0.20	-0.33	0.11	-0.34	0.49
T3	-0.17	0.01	0.02	0.02	0.10	0.05	0.03	0.00	0.02	0.05	0.18	0.02	0.02	0.09		0.04	-0.20	1.00	0.58	0.18	0.52	0.58
TT4	-0.03	0.11	0.14	0.00	0.03	0.01	0.05	0.02	0.01	0.02	0.13	0.04	0.01	0.06		0.02	-0.33	0.58	1.00	0.29	0.85	0.56
T4U	-0.08	0.23	0.06	0.03	0.03	0.00	0.13	0.03	0.01	0.03	0.03	0.02	0.06	0.06		0.00	0.11	0.18	0.29	1.00	-0.23	0.07
FTI	0.01	0.00	0.10	0.02	0.01	0.02	0.01	0.00	0.01	0.03	0.13	0.04	0.02	0.03		0.01	-0.34	0.52	0.85	-0.23	1.00	0.60
target	0.04	0.07	0.07	0.00	0.05	0.02	0.01	0.03	0.03	0.09	0.22	0.00	0.03	0.09		0.07	0.49	0.58	0.56	0.07	0.60	1.00

Appendix 4: Correlation Matrix of All Datasets Features.