Predicting Obesity: A Comparative Analysis of Machine Learning Models Incorporating Different Features

by

Md.Sakibur Rahman
19101319
Kaosar Ahmed
19101328
Tanvir Alam Nafis
19101575
Md. Ridwan Hossain
19101305
Swapnil Majumder
19101572

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
May 2023

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md.Sakibur Rahman
19101319

Kaosar Ahmed
19101328

Tanvir Alam Nafis
19101575

Md. Ridwan Hossain
19101305

Swapnil Majumder
19101572

Approval

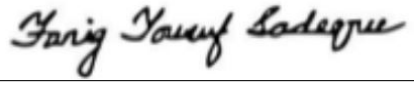
The thesis/project titled "Predicting Obesity: A Comparative Analysis of Machine Learning Models Incorporating Different Features" submitted by

1. Md.Sakibur Rahman (19101319)
2. Kaosar Ahmed (19101328)
3. Tanvir Alam Nafis (19101575)
4. Md. Ridwan Hossain (19101305)
5. Swapnil Majumder (19101572)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 25, 2023.

Examining Committee:

Supervisor:
(Member)



Dr. Farig Yousuf Sadeque
Assistant Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Associate Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Ethics Statement

The research presented here complies to the most stringent ethical requirements and was carried out in accordance with BRAC University's criteria. We affirm that the data gathered for this investigation is genuine and appropriately reflects the findings of our study. Every piece of information was collected and processed with the utmost care and accuracy, and any discrepancies or inconsistencies were notified. We have also mentioned any potential conflicts related to interests that may have impacted how the data was interpreted.

Abstract

Obesity, the excessive accumulation of body fat, is a significant health risk associated with various detrimental impacts, including the development of chronic diseases, metabolic abnormalities, joint problems, sleep apnea, mental health issues, reproductive health difficulties, respiratory disorders, liver disease, and surgical risks. The emergence of machine learning, which offers potent analytical tools and high-performance computing capabilities, has revolutionised the interdisciplinary health industry. Through improved understanding and therapeutic interventions, this technology offers opportunities to address and overcome the severe harm that obesity causes. This thesis aims to develop an automated system that utilises machine learning techniques to predict obesity based on different eating habits and relevant features. A comprehensive research methodology will be presented to categorise risk factors associated with an unhealthy lifestyle using machine learning. To effectively handle and anticipate various types of obesity, our AI system will analyse user data, including height, weight, daily food consumption habits, and more. The system will consider both weight-related and non-weight-related variables, as well as other features, to provide comprehensive insights into this health condition. Additionally, our technology will assist individuals by accurately classifying different forms of obesity, such as overweight I, overweight II, and beyond. Coefficient and correlation matrices have been utilised in the analysis to further enhance predictability. Therefore, by employing our obesity prediction algorithm, individuals can obtain estimates regarding various levels of obesity. Empowered with this information, individuals can actively improve their health status by modifying their eating habits in accordance with their specific obesity condition. The primary objective of this research is to include and exclude features associated with predicting different levels of obesity and to see how this affects the accuracy scores. A secondary dataset and a range of machine learning techniques were employed to accomplish this goal, resulting in improved predictability and accuracy of the obesity-related outcomes.

Keywords: Supervision; Machine Learning; Unsustainable lifestyle; AI system; Self Monitoring; Pre-existing diseases

Dedication (Optional)

All of the group members have worked very hard to complete this research. We are grateful to each one of us. We also devote our modest efforts to our beloved parents, whose passion, commitment, motivation, and prayer across the day and night enable us, as do all the committed and respected educators, deserving of such success and distinction.

Acknowledgement

First and foremost, we thank Allah for his bounties, which have allowed us to carry on our study without any setbacks. Furthermore, we wanted to thank all supportive faculty members, especially our thesis supervisor, Dr. Farig Yousuf Sadeque sir, for enduring our mistakes and offering continuous feedback to assist us more effectively in our research. We also wish to thank our family members and group mates for their constant encouragement throughout the semester.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AI Artificial Intelligence

BMI Body Mass Index

KNN K-Nearest Neighbors

ML Machine Learning

SVC Support Vector Classifier

WHO World Health Organization

Chapter 1

Introduction

The rising prevalence of obesity and its associated health issues have grown into a significant global health concern. Numerous factors contribute to this escalating problem, from unhealthy diets and sedentary lifestyles to a lack of awareness about individual obesity types and the severity of their conditions. The World Health Organisation (WHO) reports that obesity rates have nearly tripled since 1975, emphasising the critical need for initiatives to combat this worrying trend. Among the diseases linked to obesity are various types of cancer, diabetes, hypertension, stroke, osteoarthritis, and heart disease. Machine learning algorithms provide an innovative solution to predicting obesity, offering an opportunity to forecast a person's obesity level based on different factors. By analysing physical characteristics, dietary habits, transportation modes, and other relevant factors, machine learning algorithms can categorise individuals into obesity or normal weight classes. Leveraging these algorithms will provide insights into the severity of obesity, enabling individuals to tailor their diet plans and lifestyle changes accordingly. This personalization is crucial to mitigating the risks associated with obesity and promoting healthier living. One notable approach in this regard is to apply machine learning to caloric intake and expenditure data. By determining an individual's daily caloric needs based on their specific obesity level, we can tailor recommendations for weight maintenance and reduction. A comprehensive dataset encompassing daily food consumption, height, and dietary habits can be used to generate accurate predictions of different types of obesity. In this way, machine learning models can predict obesity based on a wide array of features, both related and unrelated to weight, as well as other key features. Moreover, our proposed models cater to individuals with specific medical conditions that necessitate restrictions on certain foods or food groups. By incorporating these considerations into our models, we aim to provide a comprehensive obesity prediction system that accounts for a broad spectrum of individual needs. The final objective is to significantly contribute to obesity mitigation efforts and improve the quality of life for individuals struggling with this health issue. Different machine learning strategies, such as content-based techniques, collaborative-based approaches, and hybrid approaches, can be utilised in these models. These techniques, if appropriately tweaked, can provide personalised food selection recommendations, adding another layer of customization to the model. Ultimately, this level of personalization fosters a heightened sense of health consciousness among individuals and encourages healthier choices. The thesis's focus on a comparative analysis of different machine learning models aims to identify the most effective algorithm or combination of algo-

rithms for predicting obesity. This comparative analysis is essential to ensuring the chosen models provide the highest level of accuracy and usefulness in real-world applications. By comparing different models, we aim to further refine our approaches and provide more precise obesity prediction, contributing significantly to the fight against obesity, both in Bangladesh and globally.

1.1 Motivation

The principal objective of our research is to harness the power of machine learning to determine if an individual is obese. Obesity is a severe health issue that is intricately associated with a multitude of chronic diseases, resulting in adverse impacts and far-reaching implications for both patients and their loved ones. Obesity, a prominent lifestyle condition, often leads to severe comorbidities. According to the World Health Organisation (WHO), diseases related to avoidable lifestyles are projected to account for 30% of all fatalities by 2030. This statistic is particularly poignant in regions like Southeast Asia, where nutrition-related issues, or malnutrition, present a double burden. While many individuals in this region still suffer from malnutrition, the prevalence of obesity has also seen a drastic upswing in recent years. In our research, we utilised a dataset with more than 16 attributes, ranging from physical characteristics to lifestyle habits, to conduct our study. This extensive list of features allows our machine learning models to generate comprehensive and accurate obesity predictions. Early identification of obesity is a critical factor in managing the condition. If an individual is diagnosed with obesity at an early stage, appropriate interventions, including dietary modifications, physical activity recommendations, or medication, can be implemented. This proactive approach prevents further health complications that can arise when obesity is left unaddressed. Given this context, our research aims to pinpoint the key elements that could determine obesity and enhance awareness of this condition. We undertake a comparative analysis of different machine learning models that incorporate these features to predict obesity. The thesis aims to determine which machine learning models, either independently or in combination, deliver the most accurate predictions and thus contribute significantly to the early identification and subsequent management of obesity. The importance of this research lies in its potential to influence individual health trajectories, inform public health initiatives, and potentially shape policy responses to the obesity epidemic. The ultimate goal is not only to develop an effective obesity prediction model but also to elevate the level of consciousness around obesity risks and preventative measures within society. By successfully achieving these objectives, our research can significantly contribute to the global fight against obesity.

1.2 Problem Statement

In the face of growing obesity prevalence, maintaining a balanced diet and a healthy lifestyle has become paramount, especially for those suffering from illnesses. Despite this, numerous individuals overlook the importance of proper dietary practices, often prioritising taste over health benefits. Alongside this issue is a substantial lack of awareness about personal obesity status and its associated health risks. While

some individuals recognise the harmful health consequences of obesity, many lack an understanding of their specific obesity stage or even the fact that they may be obese. Current research indicates a continuous preference for unhealthy lifestyles, marked by overconsumption of junk food, sugary drinks, and unprocessed red meat, leading to an escalating risk of obesity. This situation contributes to the emergence of severe diseases such as liver cancer, heart disease, and strokes, among others. However, despite the seriousness of the issue, a significant part of the population remains indifferent to the importance of a healthy lifestyle. This stark reality calls for the development of an intelligent obesity prediction system that employs machine learning methodologies. While most BMI or obesity predictions heavily rely on weight characteristics, our proposed model aims to adopt a novel approach by predicting obesity levels even without weight characteristics. Such an approach can potentially address the gaps in current prediction models, thereby offering a more comprehensive tool for individuals to assess their obesity levels and proactively address weight-related health issues. Given this context, the main problem this thesis seeks to address is: how can we develop and comparatively analyse various machine learning models, which incorporate diverse features beyond weight characteristics, to accurately predict obesity levels? This problem involves not just the creation of an efficient model but also the comparison of different models to identify the most effective one. The aim is to provide individuals with the ability to predict and understand their obesity level, which could then motivate them to pursue a healthier lifestyle. In tackling this problem, the proposed project will ethically and responsibly gather essential health-related data and highlight the critical distinction between increasing food intake for protein and vitamin enrichment versus overeating. By offering tailored guidance, such as customised dietary plans, exercise routines, and lifestyle modifications based on predicted obesity severity, the model empowers individuals to make informed decisions about their health. Consequently, the proposed obesity prediction model becomes an essential tool for healthcare professionals, enabling them to provide tailored dietary advice and targeted interventions. With access to obesity predictions, individuals can monitor progress, set goals, and celebrate milestones, promoting adherence to healthier habits. The aggregated data from the model can also aid in informing public health initiatives, policy-making, and resource allocation, addressing the broader social impact of obesity. Lastly, this thesis seeks to advance our understanding of obesity through the comparative analysis of machine learning models, thereby playing a significant role in risk assessment, personalised advice, public health impacts, and research progress.

1.3 Research Objective

In the face of the increasing prevalence of sedentary lifestyles and obesity, this thesis titled "Predicting Obesity: A Comparative Analysis of Machine Learning Models Incorporating Different Features" aims to construct, refine, and evaluate an advanced machine learning model to predict obesity levels in individuals. The challenge is not only developing such a model but also incorporating a wide array of factors beyond just physical characteristics and caloric intake.

To fulfill this overarching goal, the following research objectives have been set:

Identification and Analysis of Predictive Factors: The first objective is to conduct an exhaustive review of the existing literature and empirical studies to identify a comprehensive set of risk factors associated with obesity. These factors may include genetic predispositions, lifestyle aspects, dietary habits, and varying levels of physical activity. Each identified factor will be carefully assessed for its potential predictive power and suitability as inputs to the model.

Data Acquisition and Preprocessing: The second objective is to collect high-quality, relevant datasets necessary for developing an efficient model. This could involve both structured and unstructured data. Rigorous preprocessing of the data will be conducted to address missing values, outliers, and inconsistencies, thus ensuring it is ready for further analysis.

Designing and Training the Machine Learning Model: The third objective involves designing a machine learning model using appropriate algorithms such as KNN, Decision Trees, Random Forest, among others, chosen based on their suitability to the problem and the data. The processed dataset will be used to train the model, helping it understand the complex relationships between the predictive factors and the outcome, i.e., obesity.

Model Evaluation and Validation: The fourth objective focuses on evaluating the performance of our machine learning model. A portion of the dataset will be reserved for validation purposes, and performance metrics such as precision, recall, and F1-score will be utilized to assess the model's predictive accuracy.

Interpretability and Insights Extraction: Beyond just accurate predictions, the fifth objective is to extract valuable insights from the model's decision-making process. This interpretability can lead to a deeper understanding of the complex relationships between risk factors and obesity, offering novel perspectives on obesity prevention strategies.

Ultimately, this research aims to create a reliable, scalable, and interpretable machine learning model for predicting obesity. By accomplishing these objectives, this research can significantly contribute to early detection, prevention strategies, and global public health efforts concerning obesity.

Chapter 2

Literature Review

The motive of this paper [17] is to progress toward a machine-learning-based approach for identifying risk of obesity through applying machine-learning algorithms. The excellent thing about this study is that individuals will be able to understand the risks of obesity as well as the causes of being overweight. More than 1100 data sets were obtained from persons of various ages who are both obese and non-obese. In this paper they have used nine well known algorithms and they are KNN, random forest, logistic regression, multi layer perception (MLP), support vector machine (SVM), naive Bayes, adaptive boosting (ADA boosting), decision tree, and gradient boosting classifier, and their efficiency has been assessed using several well-known performance indicators. Obesity levels of high, medium, and low can be deduced from the experimental data. The Logistic Regression Algorithm has the highest accuracy of 97.09 percent in comparison to the other classifiers. In addition, the gradient boosting method had the lowest accuracy (64.08%) and metric values.

Food clustering analysis was utilized in this study [4] to offer a food recommendation system for diabetes patients by the authors Phanich, Pholkul and Phimoltares. Nutrition Therapy is thought to be a fantastic way to prevent, manage, and control diabetes through nutrition management, based on the notion that food is both medicine and a way of life. Their proposed system aids in the management of appropriate food substitutes from the food pyramid for diabetic patients. In this scenario, the food clustering analysis is done in two stages. The Self-Organizing Map (SOM) is built and trained first, and then clustered using the K-mean clustering approach in the second portion. The dataset which was used here is 'Nutritive Values for Thai Food' provided by the Nutrition Division of the Health Department under Thailand's Ministry of Public Health. The ultimate result is food clusters, which contain foods that provide an approximate amount of each of the eight nutrients suggested by nutritionists. This model receives a reasonably good score of 3.64 on a scale of 1 to 5 after being examined by the invited dietitians.

Machine Learning Techniques are applied in this study [16] to diagnose diabetes and propose an appropriate diet for diabetic individuals using a Diet Recommendation System (DRS). It illustrates machine learning techniques for diabetes prediction in the hospital management system for patients. The Hospital Management System and the Patients Database are the two main components of the model. Patient arrives at the hospital and enters information such as body parameters, glucose levels,

cholesterol levels, patient database, patient characteristics such as Type 1 and Type 2 diabetes, and so on. The attributes are then selected using Clustering and Machine Learning approaches. Diabetes is also predicted for a person's medical data or results using the Learning model. The result illustrates the accuracy of diabetes dataset Machine Learning algorithms with regard to different research techniques and without preprocessing method (WPP) (WOPP).

Article [11] depicts malnutrition in pregnant moms. It's been a troublesome point for quite a while. It has an immediate association with the preceding generation. Current strategies for deciding the nourishing status of pregnant ladies by estimating and classifying them utilizing the Weight Record and upper arm circle indicators.Health administrations for pregnant ladies during pregnancy with pre-birth care are standard: measure weight and level, measure circulatory strain, vaccination, measure the level of the foundation of the uterus, and regulate tablets of somewhere around 80+ tablets enriched with iron (Fe) during pregnancy. The information utilized in this study can be partitioned into two groups.First information is utilized as preparing information. Preparing information is information utilized in the order cycle. The information was taken from the anthropometric information of the pregnant ladies, consisting of: gestational age, weight, level, arm boundary, fundus and circulatory strain. The procedures utilized in this examination are: handling of clinical history information, recording of the nourishing status of pregnant ladies, assortment of information from pregnant ladies, extraction of attributes. Dietary status discovery process utilizing the Support Vector machine (SVM).The nutritional status acknowledgment process is performed through a grouping cycle. One characterization strategy is the Support Vector Machine (SVM).Using five situations, it is analyzed that the determination in view of qualities, for example, gestational age, weight, level and upper arm outline influences the presentation of the created maternal healthful status identification technique. The consequences of the main test show a precision pace of 94.6, the aftereffect of the subsequent test shows an exactness of 91.5%, the third test shows an exactness of 97.2 percent, the fourth test shows a precision of 93.5% and the fifth test shows an exactness of 95.3%.

The article authored by Gonzalo Colmenarejo presents a comprehensive survey of the utilization of machine learning methodologies for the purpose of forecasting obesity in the demographic of children and adolescents [15]. The objective of this investigation is to assess the extant scholarly works pertaining to this subject matter and ascertain the capacity of machine learning algorithms in prognosticating and averting obesity among this demographic. The author initiates the discussion by underscoring the escalating incidence of obesity among children and adolescents globally, along with its adverse impact on health. The conventional methods for forecasting obesity predominantly hinge on demographic and behavioral determinants. Machine learning methodologies present a new and auspicious strategy that can exploit a diverse array of data sources to augment the precision of predictions. The present study provides an overview of diverse machine learning algorithms that have been employed in prior research endeavors. These algorithms encompass decision trees, support vector machines, logistic regression, random forests, and neural networks. The paper provides a comprehensive analysis of the efficacy of various algorithms in predicting obesity during childhood and adolescence. The study exam-

ines the strengths and weaknesses of each algorithm and evaluates their performance in this specific context. Additionally, the significance of data pre-processing and feature selection in enhancing model performance is emphasized by the author. The identification of the most pertinent predictors of obesity is a critical task, and feature selection techniques, such as genetic algorithms and recursive feature elimination, are instrumental in this regard. The review additionally examines the obstacles and restrictions encountered by prior research endeavors, including inadequate sample sizes, absence of uniform protocols, and complexities in result comparison due to disparities in data collection and model execution. The author proposes the necessity of expanding the size and enhancing the strength of datasets, as well as establishing uniform evaluation metrics to facilitate forthcoming research and enable comparisons across various studies. To conclude, the paper highlights the potential of machine learning models in predicting obesity in children and adolescents. The utilization of models can yield significant insights into the intricate interplay among diverse factors that contribute to obesity, thereby facilitating the implementation of focused interventions and preventive measures. Additional research is required to enhance the efficacy of models, establish uniform procedures, and guarantee the pragmatic implementation of machine learning methodologies in real-life scenarios to tackle the escalating public health issue of obesity among young demographics.

This study [21] investigates the efficacy of machine learning algorithms in forecasting obesity by taking into account daily lifestyle habits and other associated factors. The authors underscore the importance of this methodology in enhancing preventative measures and tackling the worldwide obesity crisis. This study examines a range of machine learning algorithms, such as decision trees, support vector machines, logistic regression, random forests, and neural networks, and evaluates their efficacy, limitations, and predictive accuracy in the context of obesity. The significance of integrating lifestyle habits, such as dietary patterns, physical activity, sedentary behavior, and sleep patterns, as predictors in these models is emphasized by the authors. The author acknowledges the challenges that have been encountered in previous studies, including limited sample sizes and inconsistencies in data. The necessity of augmenting the reliability and generalizability of obesity prediction models is emphasized by the authors through the advocacy for larger and more diverse datasets. The paper emphasizes the potential of machine learning algorithms to predict obesity by considering daily lifestyle habits and other factors. Through the utilization of diverse data sources and sophisticated computational methodologies, these models have the potential to furnish significant insights into the identification of obesity risk factors and the formulation of efficacious preventative measures. Nevertheless, additional investigation is necessary to tackle data-related obstacles, enhance model efficacy, and guarantee the pragmatic implementation of these forecasting models in real-life scenarios. The objective of this study is to utilize machine learning methods to identify and prioritize predictor variables associated with childhood obesity.

Conventional epidemiological analyses in health-related research have proven effective in identifying specific risk factors for adverse health outcomes, such as the link between cigarettes and lung cancer [10]. However, these approaches have been less successful when it comes to conditions that are influenced by multiple factors or

where multiple variables interact to affect risk. In contrast, machine learning approaches, particularly classifiers, offer the potential to enhance risk prediction by empirically identifying patterns of variables that are indicative of a particular outcome. This differs from the conventional approach of examining isolated relationships between variables that are statistically independent and specified in advance.

The emergence of childhood obesity as a noteworthy public health issue underscores the importance of comprehending the underlying factors that contribute to its onset [19]. This knowledge is critical for the development of effective prevention and intervention measures. The research is centered on an extensive array of predictor variables that encompass various domains, such as socio-demographic, dietary, physical activity, and genetic factors. The researchers gathered information from a heterogeneous sample of minors and synthesized an extensive corpus of data. The present study employed machine learning algorithms, namely Random Forest and Gradient Boosting, to conduct an analysis of the dataset and ascertain the significance of each predictor variable with respect to childhood obesity. The study findings indicated a group of highly ranked predictor variables that exhibited a significant correlation with childhood obesity. The variables encompassed in the study comprised of sociodemographic factors, namely parental education and socioeconomic status, dietary factors such as the intake of sugary beverages and fast food, physical activity levels, and specific genetic markers. The results underscore the complex and diverse origins of childhood obesity and offer valuable perspectives on the varying degrees of significance of different factors. The research underscores the capacity of machine learning methodologies to discern crucial factors and comprehend intricate interconnections within an extensive data set. The findings of this study hold potential to make significant contributions towards the advancement of focused interventions and policies that are geared towards the prevention and mitigation of childhood obesity. By prioritizing efforts and resources effectively, healthcare professionals and policymakers can concentrate on the most influential factors. To summarize, this study presents a thorough examination of predictor variables linked to childhood obesity through the utilization of machine learning methodologies. This research highlights the significance of various factors across multiple domains and provides valuable perspectives for tackling this urgent matter in public health.

This paper presents a comprehensive survey of the utilization and progress of machine learning methodologies in the domain of obesity investigation [9]. The condition of being obese is a notable worldwide health concern that is linked to a range of unfavorable health consequences. Historically, conventional methodologies in the field of obesity research have frequently utilized statistical models and predetermined hypotheses. Machine learning has emerged as a potent instrument for analyzing intricate and extensive datasets, enabling more comprehensive and data-driven insights into obesity. The present study provides a comprehensive overview of the existing literature regarding the application of machine learning techniques in the field of obesity research, covering a diverse array of subject matters. The authors deliberate on the utilization of machine learning methodologies, namely supervised learning, unsupervised learning, and deep learning, in diverse domains of obesity, encompassing prognostication and diagnosis, hazard evaluation, optimization of treatment, and individualized interventions. The study emphasizes the

capacity of machine learning algorithms to amalgamate heterogeneous data sources, including electronic health records, genetic data, and wearable device data, to detect correlations and patterns that could potentially contribute to the onset and advancement of obesity. The aforementioned statement also tackles the obstacles linked with machine learning in the domain of obesity research, including but not limited to, the quality of data, interpretive aspects, and ethical considerations. The review provides a comprehensive overview of various studies that have employed machine learning methodologies in the field of obesity research, highlighting their efficacy and potential. In its entirety, this manuscript presents a thorough examination of the function of machine learning in the field of obesity research. The text underscores the capacity of machine learning methodologies to propel our comprehension in the domain and provides perspectives on forthcoming avenues for inquiry and the utilization of machine learning in addressing the obesity crisis.

Obesity is a global disease that affects individuals of all ages and genders. As a result, researchers have been diligently working to identify early factors that contribute to this condition. In this study, they propose an intelligent approach utilizing supervised and unsupervised data mining techniques, specifically Simple K-Means, Decision Trees (DT), and Support Vector Machines (SVM), to detect obesity levels [14]. The aim is to assist individuals and healthcare professionals in adopting healthier lifestyles to combat this widespread epidemic.To gather data, we focused on students aged 18 to 25 from educational institutions in Colombia, Mexico, and Peru. Our dataset encompasses key factors associated with obesity, including high caloric intake, reduced energy expenditure due to physical inactivity, eating disorders, genetic predisposition, socioeconomic factors, and mental health conditions such as anxiety and depression. A total of 178 students participated in the study, comprising 81 males and 97 females.By employing Decision Trees, Support Vector Machine (SVM), and Simple K-Means algorithms, we have demonstrated the efficacy of these methods through a comparative analysis. Our findings highlight the significance of these algorithms as valuable tools for examining factors related to obesity.

Chapter 3

Dataset Description

For this work, we are using a secondary dataset that was acquired from the Dataset Repository of Machine Learning at the University of California, Irvine's website. The dataset is titled, "Estimation of obesity levels based on eating habits and physical condition." As the name suggests this dataset was made with the intention of predicting obesity based on features which is based on a person's dietary practices and bodily states. This dataset is based on people from the nations of Mexico, Peru, and Colombia, according to its creators. The information was first gathered for a month using online surveys and questionnaires. Of the entire data, 485 records, or around 23%, were gathered. The Weka tool and SMOTE filter, on the other hand, helped create the remaining 77% of the data artificially. There are overall 17 characteristics in this collection. Among these, some of the traits are connected to eating practices, while others are connected to physical circumstances. To begin with, the following traits are related to eating behaviors: FAVC, FCVC, NCP, CAEC, CH20, and CALC. The following qualities, on the other hand, are related to physical factors: SCC, FAF, TUE, and MTRANS. The remaining characteristics are: Gender, Age, Height, Weight, Family history with overweight, Smoke, Obesity. They have used these names as the columns in their dataset which is difficult to interpret. So, to understand this in a clearer manner, we have changed their column names in a more meaningful way which is mentioned in the pre-processing part of this paper. Basically, the features which are associated with dietary habits of a person are based on these scenarios: Regular intake of calorie-dense foods, Consumption of vegetables on a regular basis, Quantity of major meals, Food consumption in between meals, daily consumption of water, and drinking alcohol. And the features which are associated with dietary habits of a person are based on these scenarios: Monitoring of caloric intake, Regularity of exercise, Time spent on technological gadgets, and Utilization of transportation [13].

At the moment, this collection has 2111 records. Data from this dataset are both numerical and categorical. Only 8 qualities include numerical information, whereas the other attributes all have categorical information. As we dive deeper into this information, we discover that the recordings were gathered from individuals between the ages of 14 and 61. According to the description of the dataset, there should be only 3 numerical variables which are a person's Age, Height, and Weight. The rest of the numerical variables here are categorical variables which have been already encoded. Additionally, according to the data, the features CAEC and CALC are split

into the following options: No, Sometimes, Frequently and Always. These are basically the renamed columns in our pre-processing part, 'Food Consumption Between Meals' and 'Consumption of Alcohol'. The Gender column has 2 options: Male and Female. The categories in MTRANS also have 5 options which includes Walking, along with four additional types of transportation usage: Public_Transportation, Bike, Motorbike and Automobile. This is basically the 'Type of Transportation Use' column in our pre-processed dataset. The classes in the attribute NObesity are: Insufficient weight, Normal weight, Obesity type 1, Obesity type 2, Obesity type 3, and Overweight level 1, and Overweight level 2. This is basically the 'Obesity Level' column in our pre-processed dataset. Multiple columns which have Yes and No as options are: 'family_history_with_overweight, 'FAVC', 'SMOKE', and 'SCC'. The renamed columns for these in our pre-processed are given as well: 'Family Overweight History', 'Frequent High Calory Food Consumption', 'Smoke', 'Monitor Calorie Consumption'.

There are no blank/null values in this dataset. Removing the duplicate rows, there are a total 2087 instances. Among them 1052 people are of Male gender and 1035 people are of Female gender. Also, total 1722 person has family history with overweight and the rest 365 people have no family history of overweight. Out of all, 1844 people said that they intake high calorie-dense food on a regular basis and remaining 243 people do not take them on a regular basis. Subsequently, the number of people who smoke is only 44 and the remaining 2043 people are not smokers here in this dataset. Following that, in this dataset 1991 people do not monitor their calorie consumption and only 96 people monitor their consumption. According to the dataset, around 1380 people consume alcohol sometimes. Also, 1761 people consume foods between their major meals sometimes. Around 1558 people use public transportation service and the rest of the 529 people use other modes of transportation including walking their way. Among the class labels in 'Obesity Level' the most frequent type is of Obesity Type 1 which consists of 351 instances.

The features in this dataset will be: Gender, Age, Height, Weight, Family history with overweight, FAVC, FCVC, NCP, CAEC, CH20, CALC Smoke, SCC, FAF, TUE, MTRANS. And the labels in this dataset are the categorical values of the column NObseity. So, basically the thing which we will be trying to predict here is the Obesity Levels based on the features. The novelty of the dataset has been discussed on the pre-processing part.

Chapter 4

Data Pre-Processing

An essential step in ensuring accuracy is pre-processing the data. Preparing raw data for additional analysis or modeling is the primary task of data preprocessing in the data analysis pipeline. It includes several methods designed to improve the quality of the data by cleaning, transforming, and other ways. Pre-processing is necessary to deal with missing and erroneous values since most data is noisy. Errors, missing numbers, outliers, and inconsistent results are frequently present in raw data. To assure data quality, preprocessing techniques help in locating and resolving these problems. Data may be scaled differently or come from different sources. By scaling data to a common range, preprocessing normalize data, allowing fair comparisons and lowering analytical bias. Specific criteria apply to various analytical approaches. Preprocessing converts data into an appropriate format, such as encoding categorical variables or using logarithmic transformations, to guarantee data compatibility. Leading pre-processing techniques, such as the selection of features and feature extraction, have a direct bearing on the model's accuracy. Not every feature in complicated datasets may be important. By lowering the dimension of the data and identifying key characteristics, preprocessing techniques increase the effectiveness of analysis. We can select the important features from the data and toss the rest by choosing features. With the use of feature extraction, we may turn unprocessed data into numerical options and process it while keeping the original dataset's data intact. A key component of data pre-processing where the number of attributes keeps growing is dimension reduction. Dimensionality reduction can improve our ability to perceive data as greater dimensions become more difficult to do so. It is also required to forecast missing values or fill in the mean when there are gaps in the data. Analysis precision can be hampered by class imbalances, when one class dominates the dataset. In order to produce a dataset that is more balanced in terms of classes, data preparation techniques can over- or under-sample the data or produce synthetic data. Data preprocessing is an essential stage in the data analysis process. Researchers can provide dependable and accurate results by changing, cleaning, and improving the quality of their data. It is utilized in a variety of situations, including real-world datasets, machine learning tasks, and exploratory data analysis. The effectiveness of future analysis and modeling is increased by using the right preprocessing techniques, which also improve data compatibility, feature selection, and management of imbalanced data.

4.1 Novelty of Data

The term "novelty of the data" refers to the distinctive or novel features of the dataset being examined or used in the study. It draws attention to the traits, components, or aspects of the data that set it apart from other datasets or earlier research. It describes the distinctive and novel features of the dataset under analysis. The unique traits, components, or properties that distinguish the data apart from already-existing datasets or earlier research are highlighted basically. There are several ways that novelty in the data might show up. Utilizing unique factors that have not been fully researched, merging many datasets to show new linkages, or gathering data in a particular context that yields new insights are some examples of methods that may be used. Researchers can emphasize the unique contribution of their study, indicating its relevance and possible influence in furthering knowledge in the area, by highlighting the originality of the data. The dataset used here is a secondary dataset. This was taken from the University of California's Machine Learning Repository website. As this is a secondary dataset, some works were done on this previously. To be more specific, the researchers had previously used this dataset to figure out obesity levels in a manner that was quite like ours. But there are differences between the way we are using this dataset and the way the previous authors used this dataset. For example, while we worked on this dataset, this was pre-processed in a certain manner. However, we have already taken different pre-processing procedures. As a result of pre-processing, we are utilizing the data that is substantially different from what they utilized. We have encoded the categorical variables in a different way than the previous users where we made significant differences, especially the floating values in many categorical variables. We have used scaling techniques which possibly were different from theirs. Overall, the way we handled things especially in pre-processing, this makes the dataset novel in terms of theirs. The models we have used here are somewhat different. And, we have incorporated different features in order to predict the obesity levels which was not done in previous works. We have used all the features and then tweaked the number of features to show the differences we see in the prediction scores so that we can say how some features have an impact in the obesity prediction. We will look at the methods we followed to pre-process the data before sending them towards the model for prediction.

4.2 Column Name Transformation

Renaming column names which are not understandable enough, to acceptable and relevant names is a crucial component of data preparation in the field of machine learning. The data is transformed throughout this procedure to improve its comprehensibility, interoperability, and usefulness. The data is made more accessible and makes it easier to do efficient analysis and modeling when the columns are given meaningful and descriptive titles. Renaming column names speeds up the data preparation stage and lays the groundwork for later activities, allowing researchers to gather insightful knowledge and create precise machine learning models based on correctly labeled and intelligible features.

In our used dataset, most of the columns had names which are not easily under-

standable and it required us to investigate the paper related to the dataset a lot of times. Therefore, we changed the name of the columns which can be easily interpreted for the rest of our work and we will be using the renamed column names in the rest of the part of our paper as well. The column names of the dataset after we made changes are given below:

'Gender', 'Age', 'Height', 'Weight' are kept as it is.

'family_history_with_overweight' is renamed to 'Family Overweight History'.

'FAVC' is renamed to 'Frequent High Calory Food Consumption'.

'FCVC' is renamed to 'Frequency of Vegetable Consumption'.

'NCP' is renamed to 'Frequency of Main Meals Consumption'.

'CAEC' is renamed to 'Food Consumption Between Meals'.

'SMOKE' is renamed to 'Smoke'.

'CH2O' is renamed to 'Frequency of Daily Water Consumption'.

'SCC' is renamed to 'Monitor Calorie Consumption'.

'FAF' is renamed to 'Frequency of Physical Activity'.

'TUE' is renamed to 'Frequency of Technology Usage Time'.

'CALC' is renamed to 'Consumption of Alcohol'.

'MTRANS' is renamed to 'Type of Transportation Use'.

'NObeyesdad' is renamed to 'Obesity Level'.

4.3 Handling/Missing Null Values

A major part of data preparation is dealing with missing or null values. Missing figures can happen for a few different causes, including mistakes made during data collection, equipment failures, or survey response rates that were too low. To maintain the accuracy and dependability of the dataset, missing value issues must be resolved. A variety of approaches, including complex algorithms like k-nearest neighbors (KNN) imputation, mean imputation, regression imputation, and other imputation methods, can be used to deal with missing values during the data preparation step. These methods assist in replacing missing values with approximated or anticipated values, resulting in more full and employable data for later analysis or modeling. Researchers can lessen the impact of incomplete data and avoid potential biases in the study findings by managing missing/null values correctly. Our used

dataset did not have any missing values or null values from the beginning. So, we did not need to process that part.

```
Gender                                  0
Age                                     0
Height                                  0
Weight                                  0
Family Overweight History               0
Frequent High Calory Food Consumption   0
Frequency of Vegetable Consumption      0
Frequency of Main Meals Consumption     0
Food Consumption Between Meals          0
Smoke                                   0
Frequency of Daily Water Consumption    0
Monitor Calorie Consumption             0
Frequency of Physical Activity          0
Frequency of Technology Usage Time      0
Consumption of Alcohol                  0
Type of Transportation Use              0
Obesity Level                           0
dtype: int64
```

Figure 4.1: No missing values in any of the columns.

4.4 Handling Duplicate Instances

In order to guarantee data correctness and consistency, managing duplicate rows is an essential step in the data preparation phase. Multiple datasets being combined, data input mistakes, system flaws, and other factors can all result in duplicate rows. Finding and resolving records that are the same or very similar inside the dataset is the first step in dealing with duplicate rows. This procedure involves thorough analysis and decision-making to choose the best course of action, such as eliminating duplicates, integrating them in accordance with predetermined criteria, or giving them distinctive identities to preserve their originality. Researchers can get rid of duplicate rows, improve data quality, and stop inaccurate analytic findings by properly handling duplicate rows. In our case, we tried to find out duplicate values from the dataset. We got 24 instances which are duplicates in our dataset. We removed all those instances. So, from the beginning, we had a total of 2111 instances. The shape of the dataset was (2111, 17). After removing the duplicate values, we had 2087 instances and the current shape became (2087, 17).

4.5 Managing Improper and Out-of-Range Values

During the data preprocessing stage, erroneous and out-of-range values must be found and corrected in order to ensure the correctness and dependability of a dataset. Out-of-range values represent data points that are outside of expected limits or logical limitations, whereas improper values might result from inaccuracies in measurement, data input, or system operation. To preserve data integrity and avoid biased

analytical findings, it is crucial to correct these numbers. It is possible to successfully manage incorrect and out-of-range values at the preprocessing step by using a variety of approaches. These approaches can involve the identification and elimination of outliers, the use of imputation to fill in blanks or inconsistent data, or the scaling of data to fit within predetermined ranges. Researchers must improve the dataset's correctness and reliability by taking the right steps to deal with erroneous and out-of-range numbers, which will lead to more reliable and trustworthy data analysis and modeling. In our work, this was a major task to correct the improper values and out-of-range values in the dataset. If we talk about how we detected the whole improper data situation, we must look at the datatypes of all the columns of this dataset. There were some data types given for some columns which did not go correctly with that column. For instance, the 'Age' column was of datatype float which is logically not possible. Because a person's age can never be a float number, most importantly a value which should never have more than 4 digits after the decimal point. Moreover, the categorical variables like 'Frequency of Vegetable Consumption', 'Frequency of Main Meals Consumption', 'Frequency of Daily Water Consumption', 'Frequency of Physical Activity', 'Frequency of Technology Usage Time' were showing that they were of type float which also is not logical cause they are categorical variables and there should not be any float values here. The datatypes are shown below for better understanding as well:

```
Gender                                object
Age                                  float64
Height                               float64
Weight                               float64
Family Overweight History             object
Frequent High Calory Food Consumption object
Frequency of Vegetable Consumption   float64
Frequency of Main Meals Consumption  float64
Food Consumption Between Meals        object
Smoke                                 object
Frequency of Daily Water Consumption float64
Monitor Calorie Consumption           object
Frequency of Physical Activity       float64
Frequency of Technology Usage Time   float64
Consumption of Alcohol                object
Type of Transportation Use            object
Obesity Level                         object
dtype: object
```

Figure 4.2: Data types of the columns at the very beginning.

Also, we looked at the unique values in all the columns in our dataset and that also strengthens our claim that we had improper and out-of-range values in our dataset. For instance, the 'Age' column had 1402 unique values which is not possible due to some reasons. According to the description of all the columns, the highest and lowest value in 'Age' column is 61 and 14 respectively. And it is also not possible to have this large number of unique values in this range unless there are floating point values of different ranges. But also, at the same time age can not be a floating-point value, it must be of type int. We noticed this type of issue in the case of previously

mentioned categorical variables as well. There were many unique values in those variables which were supposed to be within a limited range. 'Frequency of Vegetable Consumption', 'Frequency of Main Meals Consumption', 'Frequency of Daily Water Consumption', 'Frequency of Physical Activity', 'Frequency of Technology Usage Time' had 810, 635, 1268, 1190, 1129 unique values respectively which is simply not possible for a categorical variable. It is pretty much clear that these issues needed to be solved in order to have a proper dataset to work with. The unique values are shown below as well:

```
Gender - 2
Age - 1402
Height - 1574
Weight - 1525
Family Overweight History - 2
Frequent High Calory Food Consumption - 2
Frequency of Vegetable Consumption - 810
Frequency of Main Meals Consumption - 635
Food Consumption Between Meals - 4
Smoke - 2
Frequency of Daily Water Consumption - 1268
Monitor Calorie Consumption - 2
Frequency of Physical Activity - 1190
Frequency of Technology Usage Time - 1129
Consumption of Alcohol - 4
Type of Transportation Use - 5
Obesity Level - 7
```

Figure 4.3: Number of unique values of the columns at the very beginning.

And below is given a snippet of some portion of the dataset to have a better understanding of what we are trying to explain here and the thing which we will try to correct in the upcoming stages sequentially. It can be seen that there are floating values in some columns which is inappropriate.

	Gender	Age	Height	Weight	Family Overweight History	Frequent High Calory Food Consumption	Frequency of Vegetable Consumption	Frequency of Main Meals Consumption
743	Male	18.381382	1.722547	53.783977	yes	yes	2.000000	3.131032
744	Male	18.000000	1.738702	50.248677	yes	yes	1.871213	3.000000
745	Male	26.698580	1.816298	86.963765	no	yes	2.341133	1.578521
746	Male	21.125836	1.638085	70.000000	no	yes	2.000000	1.000000
748	Male	21.963457	1.697228	75.577100	yes	yes	2.204914	3.623364

Figure 4.4: Columns with improper values at the beginning.

Now we will fix the values and will keep them in a proper range as per logic and according to the description from the dataset, in all the columns.

Starting with the 'Age' column, here we adjusted the whole column in such a way so that it becomes of type int. For the values which were less than 14.5, we kept them to 14 as this was the minimum value. We then iterated throughout the values and changed them to the next integer values in this way. For example, values which were greater than 14.5 and less than 15.5 were kept to 15, values which were greater than 15.5 and less than 16.5 were kept to 16 and so on. Lastly, the column was correctly converted to type int.

Then, as per dataset description, in the 'Frequency of Vegetable Consumption' column there are only 3 categories. We checked and found that there are 102 instances which are less than 1.5 value and we kept them to 1. There are 994 instances where the values are greater or equal to 1.5 and less than 2.5. We kept them to 2. Lastly, there are 991 instances where the values are greater or equal to 2.5 and less or equal to 3 and we kept them to 3.

After that, as per dataset description, in the 'Frequency of Main Meals Consumption' column there are only 3 categories. We checked and found that there are 296 instances which are less than 1.5 value and we kept them to 1. There are 176 instances where the values are greater or equal to 1.5 and less than 2.5. We kept them to 2. Lastly, there are 1387 instances where the values are greater or equal to 2.5 and less or equal to 3. Also, there are 228 instances which are greater than 3. We kept them to 3.

Following that, as per dataset description, in the 'Frequency of Daily Water Consumption' column there are only 3 categories. We checked and found that there are 478 instances which are less than 1.5 value and we kept them to 1. There are 1107 instances where the values are greater or equal to 1.5 and less than 2.5. We kept them to 2. Lastly, there are 502 instances where the values are greater or equal to 2.5 and less or equal to 3. We kept them to 3.

Subsequently, as per dataset description, in the 'Frequency of Physical Activity' column there are only 4 categories. We checked and found that there are 714 instances which are less than 0.5 value and we kept them to 5 first and then 1. There are 759 instances where the values are greater or equal to 0.5 and less than 1.5. We kept them to 6 first and then 2. There are 495 instances where the values are greater or equal to 1.5 and less than 2.5. We kept them to 7 first and then 3. Lastly, there are 119 instances where the values are greater or equal to 2.5 and less or equal to 3. We kept them to 8 first and then 4.

Afterwards, as per dataset description, in the 'Frequency of Technology Usage Time' column there are only 3 categories. We checked and found that there are 932 instances which are less than 0.5 value and we kept them to 5 first and then 1. There are 912 instances where the values are greater or equal to 0.5 and less than 1.5. We kept them to 6 first and then 2. Lastly, there are 243 instances where the values are greater or equal to 1.5 and less or equal to 2. We kept them to 7 first and then 3.

Now, if we check below the unique values of all these columns, we will see that they are in a proper range now and it also makes sense that there are the correct number

of categories.

```
Gender - 2
Age - 40
Height - 1574
Weight - 1525
Family Overweight History - 2
Frequent High Calory Food Consumption - 2
Frequency of Vegetable Consumption - 3
Frequency of Main Meals Consumption - 3
Food Consumption Between Meals - 4
Smoke - 2
Frequency of Daily Water Consumption - 3
Monitor Calorie Consumption - 2
Frequency of Physical Activity - 4
Frequency of Technology Usage Time - 3
Consumption of Alcohol - 4
Type of Transportation Use - 5
Obesity Level - 7
```

Figure 4.5: Number of unique values of the columns after corrections were made.

Here, the 'Height' and 'Weight' column has a big range of unique values but that is not much of an issue because they can be float values in real life as well. Yes, there is a limitation in digits after decimal point in real life, but we ignored that issue considering they are in float format. So, now in terms of range the whole dataset is correct as per our work so far.

4.6 Categorical Encoding

Categorical variables must be transformed into a numerical format that can be analyzed, and categorical encoding is a crucial part of data preparation. Categorical variables like gender, color, or product category must be converted into a numerical representation that machine learning algorithms can understand. The several encoding methods used to achieve this transition. One-hot encoding generates fresh binary columns for each category, indicating whether that category is present in a certain observation or not. In order to express categories as numbers, label encoding gives each category a distinct numerical label. In order to represent the relative order or ranking, ordinal encoding assigns number values depending on the ordinal connection between categories. By using the right categorical encoding techniques, researchers may efficiently include categorical variables into their studies and use their important data for precise and insightful machine learning results.

In our work, we have used Ordinal Encoding to encode the categorical variables which were not encoded previously. Ordinal encoding categorizes the different types in a sequential number order. So, in the "Gender" column, we used 0 and 1 respectively to encode the options Female and Male. In 'Family Overweight History', 'Frequent High Calory Food Consumption', 'Smoke', 'Monitor Calorie Consumption' column, we used 0 and 1 respectively to encode the options Yes and No.

In the 'Food Consumption Between Meals', 'Consumption of Alcohol' column, we used 1, 2, 3, and 4 respectively to encode the options No, Sometimes, Frequently and Always. Lastly, in the 'Type of Transportation Use' column, we used 0, 1, 2, 3, and 4 respectively to encode the options Walking, Public_Transportation, Bike, Motorbike and Automobile. We also properly categorized all the variables in the column properly as can be seen below. Label Encoder has been used to encode our target variable 'Obesity Level. The options 'Normal_Weight', 'Overweight_Level_I', 'Overweight_Level_II', 'Obesity_Type_I', 'Insufficient_Weight', 'Obesity_Type_II', 'Obesity_Type_III' has been encoded with the value 1, 5, 6, 2, 0, 3, and 4 respectively.

4.7 Scaling

In the process of standardizing or normalizing a dataset's numerical properties, scaling is a critical step in the preparation of data. The scaling procedure makes sure that variables with various scales of measurement units are placed on an equivalent level, avoiding some characteristics from monopolizing the analysis or modeling process due to their size. Standardization (or z-score normalization) is a common scaling approach that changes the data to have a mean of 0 and a standard deviation of 1, as well as min-max scaling, which rescales the data to a predefined range, usually between 0 and 1. Researchers can efficiently scale numerical characteristics to a common scale, enabling fair comparisons, enhancing algorithm performance, and guaranteeing accurate and valuable analytic results.

In our proposed work, we have used the MinMax scaling method which scaled the values within 0 to 1. We basically scaled 3 of our columns which are 'Age', 'Height' and 'Weight'. These 3 columns had values which were required to be scaled so that models could learn properly while being as less biased as possible. We made a copy of the dataframe and applied scaling. Also, it is mentioned here how the MinMax scaler works in a column.

	Gender	Age	Height	Weight	Family Overweight History	Frequent High Calory Food Consumption	Frequency of Vegetable Consumption	Frequency of Main Meals Consumption	Food Consumption Between Meals	Smoke
0	0	0.148936	0.320755	0.186567	0	1	2	3	2	1
1	0	0.148936	0.132075	0.126866	0	1	3	3	2	0
2	1	0.191489	0.660377	0.283582	0	1	2	3	2	1
3	1	0.276596	0.660377	0.358209	1	1	3	3	2	1
4	1	0.170213	0.622642	0.379104	1	1	2	1	2	1

Figure 4.6: A Pre-processed portion of the full dataset.

value_std = (value - value.min(axis=0)) / (value.max(axis=0) - value.min(axis=0))
value_scaled = (value_std * (max - min)) + min
Here,
value: Value of an instance in that column.
value.min(axis=0): Minimum value of that column.

value.max(axis=0): Maximum value of that column.
min, max: range (by default it is 0 and 1)

If we look at our pre-processed dataset now, it will look like the image below. Here, we have shown the head of the dataframe or the first 5 instances and a portion of the full pre-processed dataset.

Moreover, we also categorized all the variables properly now as it can be seen below in the image given. All the categorized variables are now given the category type. Also, the other numerical variables are also properly categorized as int and float.

```
Gender                                  category
Age                                        int64
Height                                   float64
Weight                                   float64
Family Overweight History               category
Frequent High Calory Food Consumption   category
Frequency of Vegetable Consumption      category
Frequency of Main Meals Consumption     category
Food Consumption Between Meals          category
Smoke                                   category
Frequency of Daily Water Consumption    category
Monitor Calorie Consumption             category
Frequency of Physical Activity          category
Frequency of Technology Usage Time      category
Consumption of Alcohol                  category
Type of Transportation Use              category
Obesity Level                           category
dtype: object
```

Figure 4.7: Proper categorization of all the variables of the dataset.

4.8 Data Splitting

Splitting the dataset into distinct subsets is a crucial step in data preparation that enables efficient model training, validation, and assessment. Data splitting is used to evaluate how well machine learning models perform on new data and to avoid over- or underfitting. Typically, the dataset is split into a training set and a testing set, which are both used to pick the best-performing model and assess the model's performance when it will be exposed to new unseen data. Sometimes a validation set is constructed as a second subset for the purpose of fine-tuning model parameters and evaluating for finding the best fitted final model. To assure representative samples and prevent adding bias, the distribution of data to each subgroup must be well thought out. Researchers can employ data splitting to improve model parameters, analyze model performance objectively, and make defensible choices about the deployment of models. Splitting of a dataset can be done in either Random split or Stratified split. By using Stratified split technique, it is guaranteed that the percentage of each class or category remains constant while comparing various subsets. When a dataset is stratified split, the relative frequencies of classes or categories

within each subset are preserved. Stratified splitting helps to avoid the introduction of bias during model training, validation, and testing by maintaining the class distribution. This method is frequently used in classification jobs to make sure the model has enough exposure to every class and can successfully identify patterns and generate reliable predictions throughout the whole dataset. By implementing a stratified split, the assessment process becomes more robust and reliable, producing measures for model performance that are more reliable and accurate. In our case, we divided the pre-processed dataset into 2 sets: Training and Testing. The target variable was the column 'Obesity Level' and all the rest of the columns were taken as possible feature variables. We made a split in such a way that 70% of the data will be chosen for training and the rest 30% will be chosen for testing. As a result, 1460 instances were for training and the shape for the feature set was (1460, 16) and for the class set (1460, 1). On the contrary, 627 instances were for testing and the shape for the feature set was (627, 16) and for the class set (627, 1). Here, we also used stratified splitting as it distributes the dataset evenly while splitting for training and testing so that the models will be able to learn patterns for every class. We did not make a Validation or Developmental split separately because we later used Grid Search CV to fine tune the model parameters and in there a validation split is done by the machine from the training set.

Chapter 5

Model Description

5.1 Machine Learning

Machine learning is a branch of AI that deals with computers' capacity to learn new tasks by inferring patterns and relationships in data without being explicitly programmed. These models take in information, process it, and then draw judgments about the world based on what they've learned. The primary goal of machine learning is to examine data and build understandable models for humans.
Algorithms for machine learning may be broken down into the three main types of learning: supervised, unsupervised, and reinforcement. Supervised learning is a kind of machine learning that requires labeled data to train systems so that they may predict future events based on the annotations supplied. In contrast, unsupervised learning algorithms seek to discover hidden patterns in data without the use of labels. Algorithms based on reinforcement learning learn from their surroundings by accumulating rewards and punishments [2].
Many steps go into the process of putting a machine learning algorithm into action. The preparation phase includes activities such as data cleansing, missing value imputation, and translating qualitative data into quantitative representations. Once the data has been cleaned and organized, it will be split into a training set and a test set. After the training set has been created, the method may be used to discover patterns and correlations in the testing set. The predicted performance of the trained model is then measured against a testing set.
K-Nearest Neighbors (KNN), decision trees (DT), random forests (RF), and gradient boosting are all machine learning techniques that might be useful in the context of obesity prediction. Assuming that people with comparable characteristics tend to have similar health outcomes, KNN makes predictions about obesity based on the similarities between 'K' people in the dataset. Predictions may be made using individual qualities, such as exercise levels or food habits, with the help of a decision tree's hierarchical tree-like structure. In order to improve accuracy and resistance to overfitting, Random Forest, an ensemble approach, integrates the results of numerous decision trees. Gradient boosting is a method for improving prediction accuracy by incrementally optimizing any differentiable loss function using a series of basic prediction models, most often decision trees.
Machine learning methods such as KNN, decision trees, random forests, and gradient boosting may be included in obesity prediction models to make them more inclusive and accurate. These models are essential for the early detection and prevention of

obesity and their application to individualizing healthcare and therapy [18].

5.2 Supervised Learning

A key idea in machine learning called supervised learning is vital in addressing a variety of issues in the real world. This sort of learning uses labeled training data to teach the machine learning algorithm how to anticipate or categorize brand-new, unforeseen events. Each training sample in supervised learning contains input characteristics and their matching target labels, giving the algorithm a clear picture of what to expect as a result. The algorithm can learn patterns and correlations between input attributes and output labels using this supervised framework, which enables it to generalize and make precise predictions on unobserved data. The main goal of supervised learning is the development of a mapping function that can accurately connect input characteristics to matching output labels. By analyzing the patterns and correlations between the input attributes and the known output labels in the training data, the algorithm may learn to recognize underlying patterns and make educated predictions on new, unforeseen cases. Supervised learning is extremely effective and extensively applicable across a variety of fields due to its capacity to generalize from labeled data.

When creating predictions or categorizing data is crucial, supervised learning is essential in many different disciplines and industries. For instance, it may be used to estimate market trends or stock prices in finance using past data. It can help in the diagnosis of diseases or the forecasting of patient outcomes. It can be used in natural language processing to categorize text or carry out sentiment analysis. Also, supervised learning is widely used in a variety of fields, including recommendation engines, fraud detection systems, and image recognition jobs. Within the field of supervised learning, several techniques and algorithms exist, each with particular advantages and applicability for particular sorts of tasks. Algorithms that are often utilized include gradient boosting, neural networks, support vector machines, decision trees, random forests, and logistic regression. These approaches differ in terms of their complexity, interpretability, and effectiveness with respect to various types of data in terms of modeling the link between input attributes and target labels using mathematical and statistical techniques. The value of supervised learning is found in its capacity to automate decision-making procedures, spot complex patterns in data, and generate predictions with a high level of accuracy. Employing data with labels, supervised learning enables companies and researchers to get insightful knowledge, take sensible judgments, and resolve complicated issues that would have been difficult or time-consuming. Additionally, semi-supervised learning and reinforcement learning, two of the most sophisticated machine learning approaches, are built on the basis of supervised learning.

In general, supervised learning is a strong technique that may be applied to many different tasks. To develop precise and trustworthy models, it is crucial to be aware of the difficulties associated with supervised learning, such as data collecting and data quality. In the end, supervised learning is a cornerstone of machine learning that leverages the power of training data with labels to allow computers to learn from examples and provide precise predictions or classifications. It is an essential

tool for resolving issues in the real world due to its adaptability and efficacy across several areas. Researchers and practitioners may maximize the value of their data, find hidden patterns, and make wise judgments that promote innovation and advancement in their respective professions by utilizing the concepts of supervised learning.

5.3 Hyperparameter Tuning

Finding the ideal setting for a given model's hyperparameters is known as Hyperparameter Tuning, which is an essential component of machine learning. Hyperparameters are parameters that are chosen in advance of the learning process and control the model's performance and behavior. Hyperparameters are manually specified by the practitioner or researcher, as opposed to the model's intrinsic parameters, which are determined by the data. In order to maximize the model's performance on unobserved data or to achieve the ideal balance between accuracy and computing efficiency, hyperparameter tuning aims to determine the hyperparameter values that work best together. Because the default settings for hyperparameters might not always produce the greatest performance for a specific situation, hyperparameter customization is required. A model's performance may be greatly enhanced by fine-tuning its hyperparameter values, which are dependent on the datasets and learning tasks being used. The model's capacity to recognize complicated patterns in the data may be improved by making the proper choice of hyperparameters, which can also reduce overfitting and increase generalization.

The tuning of hyperparameters may be accomplished using a variety of techniques, such as grid search, random search, Bayesian optimization, and evolutionary algorithms. Grid search thoroughly examines each combination of a predetermined set of hyperparameter variables to assess the model's performance. A more effective search in high-dimensional hyperparameter spaces is made possible by random search, which chooses hyperparameter values at random from predetermined ranges. In Bayesian optimization, the search process is guided by a statistical model that iteratively explores interesting locations. In order to evolve and improve the hyperparameters across generations, evolutionary algorithms imitate the process of natural selection. Hyperparameter tuning's significance cannot be emphasized enough. Through careful tweaking, it enables academics and practitioners to get the most out of their machine learning models and ensure the greatest performance. The accuracy, generalization, and robustness of a model may all be enhanced with proper hyperparameter adjustment. It is essential for identifying hyperparameter combinations that balance model performance and efficiency, which is important for maximizing computing resources.

In conclusion, finding the ideal configuration of hyperparameters to improve model performance is known as hyperparameter tuning, which is a crucial step in machine learning. Practitioners can strengthen the generalization capacities of the model, reduce overfitting, and improve model accuracy by fine-tuning hyperparameters. Researchers can test multiple search tactics to find the optimal hyperparameter settings because there are several tuning techniques accessible. The performance and efficacy of machine learning models may be greatly impacted by the careful tuning of hyperparameters, enabling practitioners to fully utilize their algorithms in the solution of challenging real-world issues. Overall, as it may considerably enhance

the model's performance, hyperparameter tweaking is a crucial step in the machine learning process. It is crucial to remember that hyperparameter adjustment may be time-consuming and costly in terms of computing.

5.4 Grid Search CV

Machine learning practitioners frequently utilize Grid Search CV, commonly referred to as Grid Search Cross-Validation, for hyperparameter tuning. To determine the ideal configuration, a model's performance is systematically assessed over a grid of hyperparameter combinations. Grid Search CV is a potent technique for hyperparameter optimization since it combines the ideas of grid search with cross-validation. Grid Search CV is required since hyperparameters have a substantial impact on a model's performance and generalization abilities. However, determining the ideal hyperparameter values may be difficult and time-consuming. This procedure is automated by Grid Search CV, which does a thorough search across a predetermined grid of hyperparameter variables and assesses the model's effectiveness using cross-validation.

A list of potential values for each of the hyperparameters of interest in Grid Search CV is specified in order to create the hyperparameter grid. After that, the approach uses k-fold cross-validation to train and test the model with every conceivable set of hyperparameters. In order to compare and choose the optimum configuration, the performance measure, such as accuracy or mean squared error, is computed for each combination. The value of Grid Search CV rests in its capacity to optimize model performance by fine-tuning hyperparameters. It aids in determining the ideal combination that produces the greatest outcomes by methodically examining the hyperparameter space. Grid Search CV also uses cross-validation to measure the performance of the model in a trustworthy and objective manner, reducing the danger of overfitting and helping to ensure adaptation [12].

The straightforwardness and ease of deployment of Grid Search CV are two benefits. It is an easy technique that does not need any significant adjustments to be used with different machine learning algorithms. Grid Search CV also enables individuals to evaluate the effectiveness of various hyperparameter configurations, offering information on the effects of each hyperparameter on the performance of the model. On the contrary, the computational expense of Grid Search CV is a drawback, particularly when working with several hyperparameters or an extensive number of values. It may take a long time and use a lot of resources to conduct a comprehensive search. Furthermore, Grid Search CV makes the assumption that the space of hyperparameter is discrete and finite, which might not be as appropriate in some situations involving continuous or infinite regions.

In conclusion, Grid Search CV is an effective method for fine-tuning hyperparameters in machine learning. Grid search and cross-validation are used to methodically explore the hyperparameter space and pinpoint the ideal configuration. Grid Search CV helps experts fine-tune their models, improve performance, and increase generalization by automating the hyperparameter tweaking process. Although it benefits from being straightforward and easy to understand, it may be hindered by compu-

tational costs and a restricted range of continuous or infinite hyperparameter spaces that it may be applied to. Grid Search CV is still a well-liked and successful technique for optimizing hyperparameters in a variety of machine learning problems. For fine-tuning the hyperparameters of machine learning models, Grid Search CV is a powerful tool. The benefits and drawbacks of employing Grid Search CV should also be understood before utilizing it.

5.5 Decision Tree

The decision tree includes a structure made up of trees that resembles a flowchart, with each leaf node serving as the conclusion, an internal node serving as the characteristic (or attribute), and a branch serving as a decision rule. A decision tree's base node is a node located at the very top. It gains the ability to split using attribute value. Recursive partitioning is the process of repeatedly splitting a tree. This format, which resembles a flowchart, helps you make judgments. It is a representation in the form of a flowchart diagram that accurately mimics humanlevel thinking. Decision trees are, therefore, easy to understand and comprehend. Moreover, White box ML methodology takes the form of decision trees. It possesses an inherent logic of decision-making that black box methods that include neural network algorithms may not. It takes less time to train than the neural network method. The number of records and qualities in the input data are directly correlated with the time complexity of decision trees. In a not parametric or distribution-independent approach, the decision tree does not make use of any assumptions about probability distributions. High-dimensional data can be dealt with by decision trees with incredible precision [7]
One of the greatest advantages of using decision trees is that they are intuitive and simple to understand. When compared to more complex black-box models like neural networks, decision trees provide a transparent and interpretable representation of the decision-making process. The tree structure, which consists of nodes and branches, helps analysts and stakeholders understand and defend estimates. This aspect of interpretability is of paramount importance in areas where openness in decision-making is crucial, such as healthcare and finance. It's also worth noting that decision trees can handle both categorical and quantitative data with similar ease. Their flexibility makes them useful in a wide range of contexts. In addition, decision trees can handle outliers and missing data during tree building or through imputation methods, making them robust against these common problems. Decision trees are robust against real-world challenges, which can be found in many different types of data, because of their flexibility. Another benefit is that it may do feature selection in an implicit fashion. Since the value of the variables is assessed at each stage of the splitting process in decision trees, the most predictive variables can be identified. In addition to enhancing the model's explainability, this property aids in reducing its dimensionality, which boosts computational efficiency and decreases the likelihood of overfitting.Even though decision trees contain many useful features, it is important to be mindful of their constraints. Their tendency to overfit the training data is a major downside, especially in cases where the tree depth is poorly maintained. When the tree overfits by accumulating noise or irrelevant patterns in the training data, it has a hard time generalizing to unseen examples. By employing regularization strategies, such as cost-complexity pruning, the tree's complexity can

be kept in check, and generalization can be enhanced. The situation can then be fixed thanks to this.
Decision trees may perform poorly when presented with imbalanced data. In cases of unequal class distribution, decision trees may favor the class with more data, leading to subpar results for classes with fewer examples. Using methods like stratified sampling, class weighting, or ensemble approaches like Random Forests, which produce more nuanced predictions, can help relieve this issue to some extent.

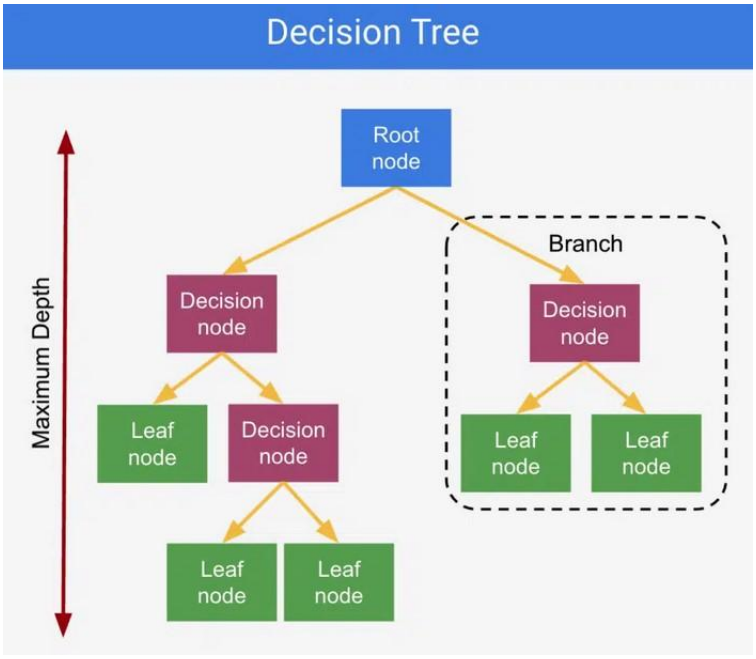


Figure 5.1: Decision Tree Classifier.

5.6 RandomForest

The Random Forest technique, a popular ensemble learning method, has been widely recognized as a powerful tool for predictive modeling. By aggregating the outputs of multiple decision trees, Random Forest can effectively improve the accuracy and robustness of predictions. The construction of decision trees involves the use of distinct subsets of the training data and a varied selection of features from the feature space. This approach is commonly employed in machine learning and data mining applications. The use of different subsets of training data and features is intended to enhance the accuracy and robustness of the decision tree model. By incorporating diverse data and features, the decision tree can better capture the underlying patterns and relationships in the data, thereby improving its predictive power. The integration of individual forecasts from a decision tree ensemble is a crucial step in producing a final prediction. This integration can be achieved through either a voting mechanism or an averaging mechanism. The decision trees in the ensemble work together to produce a more accurate prediction than any individual tree could achieve on its own. The Random Forest algorithm is a popular ensemble learning method that is composed of multiple decision trees. These decision trees are the fundamental building blocks of the Random Forest model. Each decision tree in

the Random Forest is constructed using a random subset of the training data and a random subset of the features. The final prediction of the Random Forest model is obtained by aggregating the predictions of all the individual decision trees. Therefore, the components that constitute a Random Forest are commonly referred to as decision trees. The process of making predictions in machine learning involves partitioning the input space according to various qualities. This is followed by utilizing the majority class or the average value at the leaf node to make predictions. The Random Forest algorithm employs the bagging technique, which involves generating multiple bootstrap samples from the training dataset. This method is widely used in the construction of Random Forest models. The present study highlights the importance of ensuring diversity in decision trees, which is achieved through the utilization of a distinct bootstrap sample for each tree during the training process. This approach is widely recognized as a crucial factor in enhancing the accuracy and robustness of decision tree models. In the process of generating decision trees, the Random Forest algorithm employs a technique that involves selecting a subset of features at each split in a completely random manner. This approach is intended to introduce a higher degree of variation in the model, thereby enhancing its overall performance. The implementation of a random selection process has resulted in a reduction of the interconnectivity between the trees, thereby mitigating the issue of overfitting [1].

The Random Forest algorithm is a popular ensemble learning method that com-

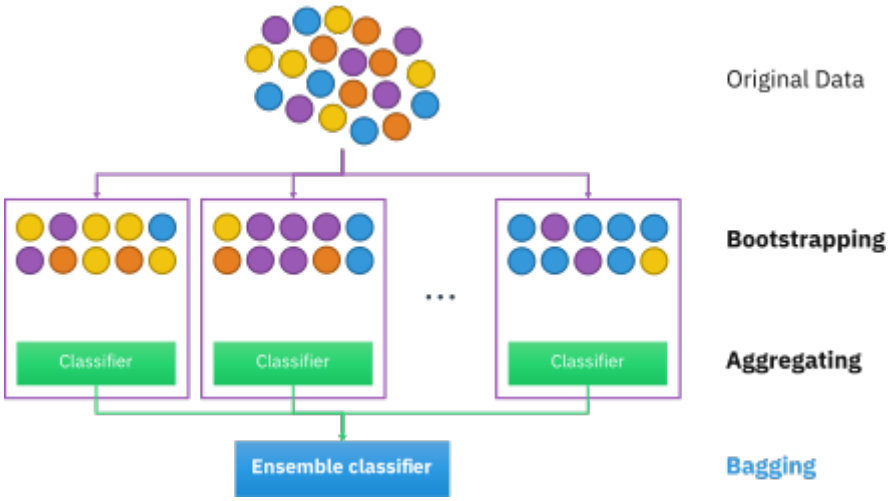


Figure 5.2: Random Forest Classifier

bines the predictions of multiple decision trees to generate a final output. Specifically, the individual decision trees' forecasts are aggregated through ensemble voting or averaging to produce an overall prediction. This approach has been shown to improve the accuracy and robustness of machine learning models, particularly in high-dimensional and noisy datasets. In the context of classification tasks, the category that garners the highest number of votes is typically employed to generate the ultimate prediction. This approach is commonly utilized in various machine learning applications, where the classification of data points into distinct categories is a critical task. The selection of the category with the most votes is based on the principle of maximizing the accuracy of the classification model. This technique has been shown to be effective in numerous studies and is widely employed in practice.

In regression analysis, the process involves determining the mean of the predicted values for a given set of tasks. The Random Forest algorithm has been widely recognized for its numerous benefits, including its exceptional predictive accuracy, robustness to noise and outliers, scalability to handle large datasets, and ability to effectively manage high-dimensional feature spaces. These advantages have made Random Forest a popular choice in various fields, including machine learning and data mining.

5.7 The Support Vector Classifier (SVC)

The Support Vector Classifier (SVC), a renowned machine learning algorithm, serves as a potent tool for classification tasks, including predicting the risk of obesity in the context under discussion. As a supervised learning model, SVC belongs to the broader category of Support Vector Machines (SVMs), renowned for their efficacy in binary classification tasks. Furthermore, SVC can be extended to address multi-class classification scenarios.

Fundamentally, the SVC algorithm endeavors to discover an optimal hyperplane that successfully demarcates data into distinct classes while maximising the inter-class margin. This hyperplane, termed as the decision boundary, segregates the data points. Meanwhile, the margin signifies the distance separating the hyperplane from the nearest data points in each class. The data points in closest proximity to the decision boundary are denoted as support vectors and play a crucial role in determining the optimal hyperplane [5].

Here's a detailed elucidation of the steps involved in deploying an SVC classifier to predict obesity risk:

Data Preprocessing: Prior to applying the SVC technique, the data necessitates preprocessing, which involves handling missing data, encoding categorical variables, and scaling numerical features, as required. This ensures that the data conforms to a format amenable to the classifier.

Selection: Identifying relevant features that can accurately predict obesity risk is imperative. Various strategies, ranging from domain expertise and statistical analysis to feature importance algorithms, can facilitate this task.

Data Splitting: The dataset is partitioned into a training set and a test set. The training set facilitates the training of the SVC classifier, whereas the test set enables the evaluation of its performance on unlabeled data.

Training the SVC Model: The SVC classifier is initialized with pertinent hyper-parameters, encompassing kernel type (e.g., linear, polynomial, radial basis functions), regularization parameter (C), and kernel coefficient (gamma). Subsequently, the model undergoes training on the training set, striving to identify the optimal hyperplane to segregate the data points into their respective obesity risk labels.

Model Evaluation: Upon training the SVC model, its performance is assessed via a gamut of evaluation metrics, including accuracy, precision, recall, and the F1 score. These metrics provide insights into the classifier's proficiency in predicting obesity risks.

Hyperparameter Tuning: To enhance the SVC model's performance, hyperparameter tuning may be employed. This involves identifying the optimal combination of hyperparameter values, potentially via methods like grid search or random search. The goal is to discover the hyperparameters that yield the best performance on the evaluation metrics.

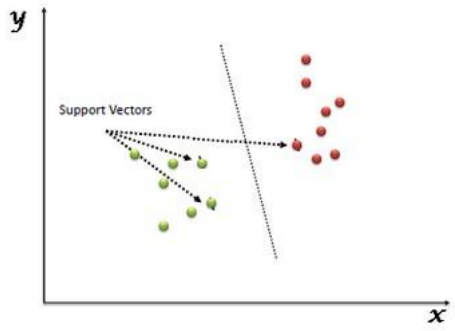


Figure 5.3: Support Vector Classifier.

Once trained and fine-tuned, the SVC classifier can be utilized to predict obesity risk in new, unseen instances. Upon receiving the input feature vectors of these instances, the classifier produces the corresponding predicted obesity risk labels.Known for its prowess in handling complex decision boundaries and operating effectively in high-dimensional spaces, the SVC classifier does pose challenges in terms of computational cost for large datasets. Additionally, achieving optimal performance hinges on kernel selection and hyperparameter adjustment.
Overall, the SVC classifier emerges as a valuable tool for predicting obesity risk as it can accurately categorize individuals based on presented features by discerning the best decision boundary between various risk classes.

5.8 K-nearest Neighbors(KNN)

The supervised machine learning technique known as the k-nearest neighbors (KNN) may be used to handle classification and regression issues. It is straightforward and simple to implement. It classifies or predicts how a set of individual data points will be arranged using proximity. The KNN algorithm believes that comparable objects may be found nearby. In other words, related objects are close to one another. The KNN algorithm saves all the information that is available and categorizes new input based on similarity. This implies that as fresh data is generated, it may be quickly categorized into a suitable category using the K-NN method [3].

The K-NN algorithm is non-parametric and implying it makes no assumptions regarding the underlying data. Because it saves the information and executes an action on it when it is time to classify, this method is also known as a lazy learner.

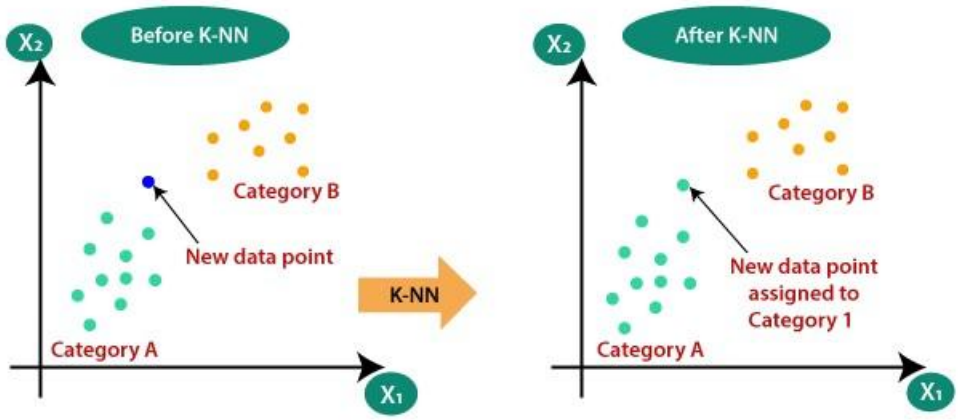


Figure 5.4: KNN Classifier.

In order to apply this algorithm, at first, we have to choose the number of neighbors K. Then we have to identify the Euclidean distance or other chosen distance between K neighbors. After that we have to pick the K closest neighbors based on the Euclidean distance or other chosen distance estimations. Compute the total amount of data points belonging to each category within these K neighbours. Then we assign the additional data points to the category where the neighbor count is at its highest. Thus, this model is completed.

5.9 AdaBoost Classifier

The AdaBoost (adaptive boosting) classifier, a highly reliable machine learning technique, is frequently employed within the realm of predictive modeling tasks, serving as a vital component in the research thesis entitled "Obesity Risk Prediction."AdaBoost, an ensemble learning technique, effectively transforms numerous "weak learners" into a unified "strong learner" over a progressive learning process. The resultant strong learner represents the ultimate prediction model, constructed upon the foundation of individual weak learners, which take the form of straightforward decision tree models.The distinctive aspect of AdaBoost lies in its adept handling of training data. Initially, all training examples are assigned equal weight. As the algorithm progresses and educates the weak learners, the weights of incorrectly classified examples progressively increase. This iterative process compels the weak learner to focus more intently on challenging instances during subsequent rounds. The term "adaptive boosting" aptly captures this process, wherein the algorithm dynamically adapts to the errors made by underperforming learners. The utilization of AdaBoost within the context of "obesity risk prediction" proves to be highly advantageous [8].

Given the multitude of variables that can influence the risk of obesity, including genetics, dietary factors, physical activity, and others, a single model may struggle to accurately forecast obesity risk. By leveraging AdaBoost, a more robust model is constructed, capable of effectively incorporating these diverse variables and their intricate interdependencies. Consequently, predictions are rendered more accurate,

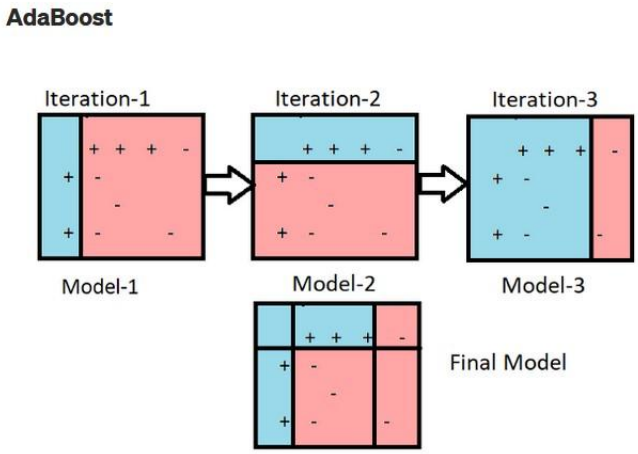


Figure 5.5: AdaBoost Classifier.

as the model adeptly handles the wide array of factors that influence obesity risk. AdaBoost also possesses notable resilience against overfitting, a significant concern in the realm of predictive modeling. Moreover, compared to certain other machine learning algorithms, AdaBoost requires less parameter tuning, making it easier to install and employ. Within our thesis research, the AdaBoost classifier played a pivotal role within the machine learning pipeline employed to predict obesity risk. Leveraging a dataset encompassing genetic information, dietary habits, physical activity levels, and other relevant attributes of individuals, we trained the system. The outcomes demonstrated that the AdaBoost classifier exhibited remarkable predictive capabilities concerning the risk of obesity, underscoring its suitability for tackling this intricate and multifaceted health issue.

5.10 GradientBoosting

The thesis research on the development of an obesity prediction model extensively employed the potent machine learning technology known as the gradient boosting algorithm [6].

Gradient boosting, an advanced technique in ensemble learning, effectively combines an ensemble of weak learners, predominantly decision trees, to yield a robust and reliable prediction model. The algorithm proceeds iteratively, creating a series of weak learners that progressively address the deficiencies of their predecessors. The process commences by initializing the ensemble with a simple model, often a single-node decision tree referred to as a stump. Denoted as F_0(x), the stump's initial prediction represents the starting point, with x signifying the input features. During each iteration, the algorithm determines the negative gradient of a pre-defined loss function concerning the ensemble's current predictions. This negative gradient provides the direction in which the ensemble's predictions need to be adjusted to minimize the loss. The loss function quantifies the disparity between the actual labels associated with the training instances and the ensemble's predicted values. To approximate the negative gradient, a new weak learner, denoted as h_t(x), is

trained using the residual errors between the true labels and the ensemble's current predictions. These scaled predictions, multiplied by a learning rate denoted as , control the contribution of each weak learner to the ensemble. Employing a small positive value for the learning rate ensures a gradual convergence of the model.
At each iteration t, the ensemble is updated by adding the scaled predictions of the new weak learner to the previous ensemble, yielding:

$$F\_t(x) = F\_t - 1(x) + \alpha^* h\_t(x). \tag{5.1}$$

This iterative process continues until a predefined stopping criterion is met or a specified number of iterations is reached. The final ensemble model, F(x), is the sum of the individual weak learners' predictions, each weighted by the learning rate:

$$F(x) = \Sigma\alpha^* h\_t(x) \tag{5.2}$$

Through a carefully orchestrated interplay of weak learners, each addressing the shortcomings of their predecessors, the gradient boosting algorithm adeptly combines predictions to capture intricate relationships and interactions within the obesity prediction model.

In the context of the thesis study on obesity prediction, the employment of the gradient boosting algorithm exemplifies its efficacy as a powerful tool. Leveraging a comprehensive dataset encompassing genetic information, dietary patterns, physical activity levels, and other pertinent factors, the algorithm is adeptly trained to predict the likelihood of obesity. The findings of this research highlight the remarkable predictive capabilities of the gradient boosting algorithm, offering valuable insights for addressing the multifaceted challenges associated with obesity.

Chapter 6

Implementation Procedures

6.1 Performance Metrics

After the data pre-processing, we have put the models into test.The test results were in a form of probability scores that are Accuracy, Precision, F1-Score and recall. We have employed the use of Confusion Matrix that evaluates the classes having Dataset values and Machine generated values. This yields a depiction of the maximum success probability.

6.2 Confusion Matrix

The confusion matrix comprises data pertaining to the factual and anticipated classifications of the model. The number of correct and incorrect predictions is tallied using count values and categorized by class. Subsequently, the numerical values are presented in a matrix format. The generation of a confusion matrix involves a comparison between the anticipated and factual class labels, with a tabulation of the number of correct and incorrect predictions for each class [20]. Following that, the tallies are organized in a tabular format, where the factual categories are represented as horizontal rows and the predicted categories as vertical columns. The elements along the diagonal of the matrix are indicative of precise predictions, while the remaining entries correspond to imprecise estimates. The utilization of a confusion matrix is a valuable method for evaluating the efficacy of a model, especially in cases where the dataset exhibits an imbalanced distribution. The table is given below:

	Predicted Value(0)	Predicted Value(1)
Actual Value(0)	True Negative(TN)	False Positive(FP)
Actual Value(1)	False Negative(FN)	True Positive(TP)

Accuracy: The concept of accuracy pertains to the ratio of accurate predictions generated by a model in comparison to the overall number of predictions made. The determination of accuracy is achieved through the division of the total number of correct predictions by the overall number of predictions made. The metric of accuracy provides a clear and concise measure of the performance of a model, indicating its overall effectiveness. Nonetheless, the precision of the results may be

misleading in the presence of imbalanced data, where one category is significantly more prevalent than the others. In scenarios of this nature, a predictive model that consistently forecasts the majority class may achieve high levels of accuracy, but it may not be deemed feasible in practice. In the present study, the dataset exhibits an imbalanced distribution, thus necessitating the incorporation of alternative metrics in addition to accuracy. The following equation(6.1) is the formula to calculating the accuracy:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Sum of all predictions}} \tag{6.1}$$

Precision: Precision can be defined as the proportion of true positive predictions generated by a model in relation to the overall number of positive predictions made. The metric evaluates the capacity of the model to refrain from generating erroneous positive predictions. A precise numerical value indicates a reduced occurrence of false positives in the model. In scenarios where the expense associated with false positives is significant, precision holds a distinct advantage. In the context of imbalanced datasets, relying solely on precision as a performance metric can be misleading. In our given scenario, we will employ it in combination with recall and F1-Score. The equation(6.2) is :

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{6.2}$$

Recall: Recall is referred to as sensitivity or true positive rate, measures the ratio of correct positive predictions made by a model in comparison to the overall count of positive instances that exist in the data. The metric evaluates the model's capacity to precisely identify all affirmative instances. A high recall value indicates that the model exhibits a low rate of false negatives. In cases where the negative consequences of false negatives are substantial, the metric of recall holds significant value. When dealing with datasets that are unbalanced, relying solely on recall metrics may be misleading. Therefore, the F1-Score is also utilized in our approach. The recall calculation is shown in the equation(6.3) below :

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{6.3}$$

F1-Score: The F1-Score is a statistical measure that represents the harmonic mean of precision and recall. The scale ranges from zero to one, where a higher value denotes superior performance. The purpose of its usage is to achieve equilibrium between the trade-off of precision and completeness. The F1-Score is a valuable optimization statistic when the expenses associated with false positives and false negatives are significant. As a result of an imbalanced distribution of classes, with a

higher frequency of negative cases than positive cases, the F1-Score will be utilized as the primary performance metric. F1-Score is equation(6.4) :

$$F1 \text{ Score } = 2 \times \frac{\text{Precision } \times \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{6.4}$$

6.3 Support Vector Classifier

The Support Vector Classifier (SVC) is a potent supervised learning methodology that is particularly adroit at addressing binary classification challenges, such as predicting the propensity to develop obesity. Herein, we delve deeper into the deployment of the SVC classifier within the obesity risk prediction model: Model Initialization: An instance of the SVC class, devoid of any explicitly supplied hyperparameters, is instantiated to initialize the SVC model. By default, it utilizes the Radial Basis Function (RBF) kernel, which is a commonly adopted choice for classification tasks.

Model Training: The SVC model, once initialized, is trained utilizing the training dataset. This dataset comprises feature vectors (x_train_new) and their corresponding obesity risk labels (y_train). By feeding the training data as input, the fit() method is employed to educate the SVC model and discern the most suitable decision boundary segregating the distinct obesity risk classes.

Test Data Prediction: Subsequent to the training of the SVC model, it is leveraged to forecast the labels indicating obesity risk within the test data (x_test_new). By inputting the test data into the SVC model via the predict() method, the anticipated obesity risk labels (svc_test_predict) are generated.

Performance Evaluation: Metrics such as accuracy, precision, recall, and the F1 score are calculated to appraise the performance of the SVC classifier on the test data. This process utilizes scikit-learn's functions including accuracy_score(), precision_score(), recall_score(), and f1_score(). These metrics elucidate the classifier's proficiency in detecting obesity risk.

Classification Report: Utilizing the classification_report() method, a comprehensive classification report is produced, offering an in-depth performance analysis of the classifier. It furnishes metrics for each obesity risk class, including precision, recall, F1 score, and support. The target_names parameter is assigned the value of Target_labels, representing the class names of obesity risk.

Confusion Matrix: A confusion matrix is constructed to visually depict the SVC classifier's performance in predicting obesity risk. This matrix, illustrating the count of true positives, true negatives, false positives, and false negatives for each obesity risk class, is created through the confusion_matrix() function. The plot_confusion_matrix() function is employed to represent the confusion matrix with

appropriate labels and color mapping.

The obesity risk prediction model harnesses the SVC classifier, capitalizing on the algorithm's ability to discern the optimal decision boundary via the use of support vectors. This capability facilitates the accurate classification of obesity risk using the available features. The confusion matrix assists in pinpointing the strengths and drawbacks of the classifier in predicting distinct obesity risk classes, and an array of metrics is employed to gauge the model's performance.

6.4 Decision Tree Implementation

The Decision Tree Classifier, a supervised learning algorithm, is primarily employed for resolving classification problems. This algorithm systematically fragments the dataset into smaller subsets, all the while incrementally constructing an associated decision tree. In our model, the decision tree classifier is harnessed to predict obesity risks based on a diverse array of factors. The model is initially trained on x_train and y_train data through the fit() function. The dataset x_train embodies independent variables, encapsulating attributes like height, occupational hours, and daily caloric intake, all of which contribute to obesity. Conversely, y_train, the dependent variable, signifies different levels of obesity. Post-training, the model is applied to the test data (x_train), utilizing the predict() method. This technique yields the predicted values (decision_tree_test_predict) for obesity risk, predicated on the correlations discerned in the training data. Subsequent to the predictions, the model's performance on the test data is assessed by determining key metrics, including accuracy, precision, recall, and the F1 score. Accuracy characterizes the ratio of accurate predictions out of all predictions made. Precision quantifies the proportion of correct positive outcomes relative to all predicted positives. Recall, also termed sensitivity, measures the ratio of correctly identified true positives. The F1 Score presents a balanced measure of the model's performance by computing the harmonic mean of precision and recall. The classification_report() function divides the test data into groups like overweight I, overweight II, and others to provide a thorough analysis of the model's performance. This classification expedites comprehension of the model's predictive prowess for each obesity risk category. Moreover, the confusion matrix graphically delineates the classification model's performance. It elucidates the correlation between the model's predictions and the actual values by spotlighting true positives, true negatives, false positives, and false negatives.

By scrutinizing the confusion matrix and the computed metrics, one can accurately evaluate the performance of the decision tree classifier in predicting obesity risk in this machine learning model.

6.5 Random Forest Implementation

An ensemble learning model called the Random Forest Classifier combines the predictions of many decision trees to provide precise predictions. The steps it takes to function are as follows: The number of decision trees, indicated by the variable "n_estimators," and other hyperparameters that regulate how each decision tree be-

haves inside the forest are first initialized into the Random Forest Classifier. These hyperparameters include things like the minimum number of samples needed for a split and the maximum depth of the trees. A training dataset with labels is then entered into the Random Forest Classifier. There are input characteristics (X) and target labels (Y) for each occurrence in the dataset. Next, the decision trees are constructed by the model. As part of a process known as bootstrap aggregating or bagging, a portion of the training data is randomly chosen for each decision tree in the forest. By employing a technique known as recursive partitioning, this subset is utilized to construct a unique decision tree. The method chooses the appropriate feature and threshold for each decision tree node that maximizes information gain or minimizes Gini impurity. A node's level of impurity is quantified by the Gini impurity, whereas the information gain measures the rise in purity or reduction in entropy following the split. Until a stopping requirement is satisfied, such as reaching the maximum depth or the minimal number of samples needed for a split, the decision tree keeps splitting the data recursively. The Random Forest Classifier aggregates the predictions made by each decision tree once they have all been constructed. The Random Forest Classifier runs the test instances through each decision tree in the forest while making a prediction on the test data. Each tree casts a vote, and the class label receiving the most votes is used as the instance's predicted label. The majority voting concept serves as the foundation for this prediction method. In conclusion, the Random Forest Classifier constructs a collection of decision trees from bootstrapped samples taken from the training set. In order to create precise predictions on hypothetical test data, it aggregates the forecasts of these trees by majority voting. Incorporating several trees and adding randomization to feature and data sampling improve the model's robustness, generalizing, and resistance to overfitting.

6.6 K-nearest Neighbors(KNN) Implementation

The K-nearest neighbor (KNN) model is an instance-based classifier that may execute the classification of unknown instances by connecting the unidentified ones to the known using specified distance or similarity functions. It is important to note that KNN is a non-parametric technique since it does not need the calculation of parameters for an imagined function, as does logistic regression. In general, the KNN model will first find the k closest neighbors of the latest data point u=(u1,u2,—uq) with an unidentified class label, followed by allocating the class label that corresponds to the majority of the k nearest neighbors as the label. In general, neighbors are discovered using distance functions. There are two extensively used distance functions: the Euclidean distance and Manhattan distance. As demonstrated in (5), we used weighted Euclidean distance in our KNN model depending on the relevance of the predictive factors. It is worth noting that the amount of weight linked with each predictive variable is proportionate to the value given by the adjusted odds ratio for that variable when implementing logistic regression.

$$\sqrt{w_1 \left(x_1^i - u_1\right)^2 + w_2 \left(x_2^i - u_2\right)^2 + \cdots + w_Q \left(x_Q^i - u_Q\right)^2}. \tag{6.5}$$

The weighted KNN model's pseudo code is as outlined below:

Input: k as the number of intended adjacent neighbors; S is the collection of training data points with defined class labels (x1,y1),(x2,y1)...(xn,yn); u is represented as a new data point with an undetermined class identifier.

1. Determine the distance that lies between each data point in S and u using the formula (6.5).
2. Sort each data point in S according to its distance calculated in the preceding phase.
3. Mark the initial k points of data within the sorting list as u's nearest neighbors.
4. Determine the predominant class designation of k selected points of data and designate it to u.

6.7 AdaBoost Implementation

The thesis project focuses on developing a reliable prediction model for assessing the risk of obesity by employing the AdaBoost classifier, an effective ensemble approach in machine learning. AdaBoost aims to construct a robust classifier by aggregating multiple weak classifiers. It iteratively builds models, with each subsequent model focusing on correcting the errors made by the previous ones, thereby improving the overall prediction performance. There is no inherent restriction on the number of models to be included, as they are added until the training set is accurately predicted. In the provided code, the default configuration of the AdaBoostClassifier is utilized. The model is then trained on the available training data, consisting of feature vectors (x_train) and corresponding labels (y_train). By fitting the model to this training data, it learns to capture patterns and relationships between the features and the target variable. Once trained, the model is applied to make predictions on the test data (x_test). The performance of the AdaBoost classifier is evaluated using various metrics, including accuracy, precision, recall, and the F1 score. Accuracy measures the proportion of correctly predicted labels, while precision assesses the accuracy of positive predictions among all positive predictions made by the classifier. Recall quantifies the model's ability to accurately identify positive instances relative to all actual positive instances. The F1 score combines precision and recall into a single metric that balances both aspects. The classification report provides a comprehensive overview of the model's performance for each individual class. It includes precision, recall, F1 score, and support (the number of instances) for each class. These metrics offer insights into how well the classifier performs for each specific label. Additionally, the report presents macro and weighted averages, providing an overall summary of the model's performance. To gain further understanding of the classifier's performance, a confusion matrix is generated. The confusion matrix visually displays the predicted labels versus the actual labels. The diagonal elements represent the correctly predicted instances, while the off-diagonal elements represent the misclassified instances. This matrix aids in identifying which labels the model predicts accurately and where it encounters challenges.

Lastly, the implementation of the AdaBoost classifier in this thesis project enhances the prediction of obesity risk. By leveraging its ability to iteratively correct errors and improve predictions, the AdaBoost classifier demonstrates the potential to generate more accurate and trustworthy predictions, thereby aiding in the assessment of obesity risk.

6.8 Gradient Boosting Implementation

The methodology outlined within our thesis paper encapsulates a machine learning technique known as gradient boosting classification. This approach leverages an ensemble of decision trees to yield predictive outcomes. In a stage-wise fashion, this model is crafted, and each successive tree is meticulously fit onto a modified version of the original dataset. The modification incorporated at each iterative boosting stage involves the application of weights to instances within the dataset. This process enhances the effect of instances that have been inaccurately predicted, thereby enabling the model to allocate more focus to complex cases that are more challenging to predict.

Here's a detailed walk-through of the implementation within your script:

Model Initialization: The Gradient Boosting Classifier model is instantiated with default parameters through the invocation of GradientBoostingClassifier(). Although these parameters could be fine-tuned according to the specificities of certain tasks, your script currently operates with the pre-established settings.

Model Training: The fit function is employed to facilitate the training of the model with your designated training data. To conform to the expected format, the .ravel() function is invoked to convert the y_train values into a compatible 1D array.

Prediction: Following successful training, the model proceeds to generate class label predictions for the provided test dataset by employing the predict function. The model's output is an array encapsulating the predicted class labels for each instance within the test dataset.

Evaluation: Subsequent to prediction, the script progresses to compute several performance metrics to assess the quality of the test predictions:

Accuracy: This metric denotes the fraction of predictions that are entirely accurate, computed as (TP+TN)/(TP+FP+TN+FN), where TP represents true positives, FP signifies false positives, TN denotes true negatives, and FN stands for false negatives.

Precision: Precision indicates the fraction of positive predictions that are genuinely accurate, and it's computed as TP/(TP+FP).

Recall: Often referred to as sensitivity, recall quantifies the fraction of actual positive instances that the model has accurately discerned, computed as TP/(TP+FN).

F1 Score: This metric offers a more nuanced perspective of the inaccurately classified instances in comparison to the Accuracy metric. It signifies the harmonic mean of precision and recall and is computed as 2*((precision*recall)/(precision+recall)).

For multi-class classification problems such as the one at hand, these metrics are computed independently for each class before a weighted average is extracted, which explains the use of "weighted average. Classification Report: The classification_report function outputs a textual report detailing the principal classification metrics for each class, which encompass precision, recall, F1-score, and support (the count of instances within each class).

Confusion Matrix: This tabular representation offers a detailed analysis of the performance of the classification model (or classifier) against a set of test data wherein the true values are known. It provides an in-depth view of prediction errors by displaying not just the tally of correct and incorrect predictions but the specifics of their distribution as well.

On the whole, the script executes a standardised protocol for training and assessing a gradient-boosting classifier on a specific dataset. This application revolves around obesity risk prediction, implying that the data likely encompasses a variety of health and lifestyle indicators, with the target variable being a classification of obesity risk.

Chapter 7

Result Analysis

To predict distinct forms of obesity in our study, we applied a number of machine learning algorithms to the pre-processed dataset. The models we used were Decision Tree Classifier, Random Forest Classifier, Support Vector Classifier, K Nearest Neighbors Classifier, AdaBoost Classifier, and Gradient Boosting Classifier. We separated the dataset into a train split and a test split after preprocessing the data. The training data will make up 70% of our dataset, while the testing data will make up 30%. Afterwards, we predicted the Obesity Level including all possible features. We then excluded and included different features at a time and noticed how that impacts our prediction scores so that we can give a verdict on how those features overall impact obesity prediction in the proposed work.

7.1 All Features Included

We get the following accuracy scores when we first introduce the models with default hyperparameters.

Actual Column Names	Renamed Columns
family history with overweight	Family Overweight History
FAVC	Frequent High Calorie Food Consumption
FCVC	Frequency of Vegetable Consumption
NCP	Frequency of Main Meals Consumption
CAEC	Food Consumption Between Meals
SMOKE	Smoke
CH2O	Frequency of Daily Water Consumption
SCC	Monitor Calorie Consumption
FAF	Frequency of Physical Activity
TUE	Frequency of Technology Usage Time
CALC	Consumption of Alcohol
MTRANS	Type of Transportation Use
NObeyesdad	Obesity Level

Then we did hyperparameter tuning using Grid Search CV for every model and extracted the best fitted hyperparameters. For the decision tree, we got Log Loss as criterion and max depth of 14. For random forest, we got Log Loss criterion and

max depth of 13. For SVC, we got a Polynomial kernel with degree 5 and auto in gamma. For KNN, we got Manhattan distance in metric with K = 1 neighbors and uniform in weight. For AdaBoost, we got a 0.01 learning rate with 7 values of n_estimators and Random Forest Model as the estimator. For Gradient Boosting, we got a learning rate of 0.1, n_estimators of 75 and max depth of 6. And after that we used these best fitted models including the weight features and we got the following accuracy scores there.

Model Name	Accuracy Score
Decision Tree Classifier	95%
Random Forest Classifier	96%
Support Vector Classifier	90%
K Nearest Neighbor Classifier	80%
AdaBoost Classifier	95%
Gradient Boosting Classifier	97%

Table 7.1: Improved accuracy scores of models with best fitted parameters (Including Weight)

7.2 Weight Feature Excluded

After that we excluded the Weight feature from the training and testing set. Because in order to detect obesity, we use BMI and according to the formula of the BMI where a person's weight is divided by squared value of height, weight plays a big factor here. Moreover, if we think about real life scenarios, it is often we look at a person's weight and investigate whether a person is obese or not. So, we now predict obesity with the best fitted models but we excluded the weight column. We can see a huge drop in our accuracy scores here which is shown below.

Model Name	Weight Included Accuracy	Weight Excluded Accuracy
Decision Tree Classifier	95%	75%
Random Forest Classifier	96%	86%
Support Vector Classifier	90%	75%
KNN Classifier	80%	78%
AdaBoost Classifier	95%	87%
Gradient Boosting Classifier	97%	85%

Table 7.2: Accuracy scores of models with Including and Excluding weight.

Here, we can see that almost all the models have a big drop in their accuracy scores which explains how much impact the weight column had in the dataset. As mentioned earlier, the BMI is used for detecting obesity level in a dataset and in that equation, Height is also included. So, next we will see how much of a drop in the prediction we get excluding the columns which are responsible directly in the calculation of the Body Mass Index.

7.3 Height Feature Excluded

Here, we excluded the height feature as well and now we will try to predict the obesity level using the best fitted models. If we remember, we excluded the weight feature previously as well. So, this time it is basically a comparison between weight excluded and both height and weight excluded. There are more drop off in the accuracy scores.

Model Name	Weight Excluded Accuracy	Both Height and Weight Excluded Accuracy
Decision Tree Classifier	75%	75%
Random Forest Classifier	86%	83%
Support Vector Classifier	75%	74%
KNN Classifier	78%	76%
AdaBoost Classifier	87%	83%
Gradient Boosting Classifier	85%	81%

Table 7.3: Accuracy scores of models with Excluding weight and excluding both height and weight.

Here we can see that there is a small drop off in accuracy scores in almost all the models. Only the decision tree classifier gives us the same accuracy scores where weight was excluded and height was the dominant factor. Only 3 of the models give us a score more than 80% in this case where both the factors of BMI calculation are gone. So, in the prediction of obesity levels, the dominant factors here are Weight and Height which play a key role in determining the obesity levels. Among them, Weight is the most dominant one and Height is after that.

Lastly, we will look at prediction levels of two more cases where we picked the features which are based on habits of eating and the features which are based on the physical conditions of a person.

Model Name	Accuracy (Features of Physical Conditions)	Accuracy (Features of Eating Habits)
Decision Tree Classifier	41%	51%
Random Forest Classifier	42%	52%
Support Vector Classifier	39%	51%
KNN Classifier	25%	41%
AdaBoost Classifier	41%	51%
Gradient Boosting Classifier	42%	52%

Table 7.4: Accuracy scores of models based on features of physical condition and eating habits.

Here, we can say that the features which are related to eating habits are more impactful in terms of predicting obesity and the features which are related to physical conditions are less impactful.

Chapter 8

Conclusion

In a nutshell, the goal of this thesis was to examine and contrast machine learning models for predicting obesity while combining a variety of characteristics. Important knowledge has been acquired about the prediction effectiveness and feature significance of the analyzed models through thorough study and experimentation. The study emphasizes how important model and feature selection are to making precise predictions and how much impact some features can make on predicting obesity. Like, we saw that weight feature heavily impacts the score of accuracy. After it is excluded, the accuracy drops and we saw that further excluding some other features and predicting the obesity levels. Despite its drawbacks, which include potential dataset biases and a small number of available models, this study offers significant new knowledge in the field of obesity prediction and paves the way for future research aiming to increase prediction accuracy and create individualized obesity prevention plans.

Bibliography

[1] A. Liaw, M. Wiener, *et al.*, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[2] M. Cord and P. Cunningham, *Machine learning techniques for multimedia: case studies on organization and retrieval*. Springer Science & Business Media, 2008.

[3] A. Mucherino, P. J. Papajorgji, P. M. Pardalos, A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, "K-nearest neighbor classification," *Data mining in agriculture*, pp. 83–106, 2009.

[4] M. Phanich, P. Pholkul, and S. Phimoltares, "Food recommendation system using clustering analysis for diabetic patients," pp. 1–8, 2010.

[5] Y. Zhang, "Support vector machine classification algorithm and its application," in *Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, September 14-16, 2012. Proceedings, Part II 3*, Springer, 2012, pp. 179–186.

[6] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics*, vol. 7, p. 21, 2013.

[7] A. Priyam, G. Abhijeeta, A. Rathee, and S. Srivastava, "Comparative analysis of decision tree classification algorithms," *International Journal of current engineering and technology*, vol. 3, no. 2, pp. 334–337, 2013.

[8] C. Ying, M. Qi-Guang, L. Jia-Chen, and G. Lin, "Advance and prospects of adaboost algorithm," *Acta Automatica Sinica*, vol. 39, no. 6, pp. 745–758, 2013.

[9] K. DeGregory, P. Kuiper, T. DeSilvio, *et al.*, "A review of machine learning in obesity," *Obesity reviews*, vol. 19, no. 5, pp. 668–685, 2018.

[10] A. S. Selya and D. Anshutz, "Machine learning for the classification of obesity from dietary and physical activity patterns," *Advanced Data Analytics in Health*, pp. 77–97, 2018.

[11] S. Widodo and S. Farida, "Software development to monitor nutritional status of pregnant women using intelligent systems," *International Journal of Research in Engineering and Science (IJRES)*, vol. 6, no. 3, pp. 1–6, 2018.

[12] P. Liashchynskyi and P. Liashchynskyi, *Grid search, random search, genetic algorithm: A big comparison for nas*, 2019. arXiv: 1912.06059 [cs.LG].

[13] F. M. Palechor and A. de la Hoz Manotas, "Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from colombia, peru and mexico," *Data in brief*, vol. 25, p. 104 344, 2019.

[14] R. C. Cervantes and U. M. Palacio, "Estimation of obesity levels based on computational intelligence," *Informatics in Medicine Unlocked*, vol. 21, p. 100 472, 2020.

[15] G. Colmenarejo, "Machine learning models to predict childhood and adolescent obesity: A review," *Nutrients*, vol. 12, no. 8, p. 2466, 2020.

[16] S. S. Bhat and G. A. Ansari, "Predictions of diabetes and diet recommendation system for diabetic patients using machine learning techniques," pp. 1–5, 2021.

[17] F. Ferdowsy, K. S. A. Rahi, M. I. Jabiullah, and M. T. Habib, "A machine learning approach for obesity risk prediction," *Current Research in Behavioral Sciences*, vol. 2, p. 100 053, 2021.

[18] M. N. LeCroy, R. S. Kim, J. Stevens, D. B. Hanna, and C. R. Isasi, "Identifying key determinants of childhood obesity: A narrative review of machine learning studies," *Childhood Obesity*, vol. 17, no. 3, pp. 153–159, 2021.

[19] H. Marcos-Pasero, G. Colmenarejo, E. Aguilar-Aguilar, A. Ramırez de Molina, G. Reglero, and V. Loria-Kohen, "Ranking of a wide multidomain set of predictor variables of children obesity by machine learning variable importance techniques," *Scientific Reports*, vol. 11, no. 1, p. 1910, 2021.

[20] M. Heydarian, T. E. Doyle, and R. Samavi, "Mlcm: Multi-label confusion matrix," *IEEE Access*, vol. 10, pp. 19 083–19 095, 2022. DOI: 10.1109/ACCESS.2022.3151048.

[21] C. Suresh, B. Kiranmayee, M. Jahnavi, R. Pampari, S. R. Ambadipudi, and S. S. P. Hemadri, "Obesity prediction based on daily lifestyle habits and other factors using different machine learning algorithms," pp. 397–407, 2022.