EXPLORING THE DYNAMICS OF SLEEP AND DAILY HABITS: A COMPREHENSIVE INVESTIGATION USING MACHINE LEARNING

Project submitted to the University of Kerala

in partial fulfilment of the requirements

for the Degree of

MASTER OF SCIENCE IN APPLIED STATISTICS AND DATA ANALYTICS

Submitted By

FYSAL KASIM

(Reg no: 85621615009)

DEPARTMENT OF STATISTICS

UNIVERSITY OF KERALA

THIRUVANANTHAPURAM

2021-2023

DEPARTMENT OF STATISTICS

UNIVERSITY OF KERALA

Dr. A. Riyaz
Assistant Professor
Department of Statistics
University of Kerala

P.O. Kariavattom
Trivandrum-695581
Phone: 0471-18905

CERTIFICATE

I hereby certify that this project **"EXPLORING THE DYNAMICS OF SLEEP AND DAILY HABITS: A COMPREHENSIVE INVESTIGATION USING MACHINE LEARNING"** is bonafide project work carried out by Mr.FYSAL KASIM , M.Sc. (Applied Statistics and Data Analysis) student in the Department of Statistics, University of Kerala, Kariavattom, during 2021-2023 under my supervision and guidance, in partial fulfillment of the requirements for the M.Sc. Degree in Applied Statistics and Data Analytics of the University of Kerala.

Place: Thiruvananthapuram
Date: September 2023

Dr. A. Riyaz
Assistant Professor
Department of Statistics
University of Kerala

DECLARATION

I hereby declare that the Project Report entitled “**Exploring The Dynamics Of Sleep And Daily Habits: A Comprehensive Investigation Using Machine Learning”** bound herewith is the original record of the project work carried out by me under the supervision and guidance of Dr. A. Riyaz, Assistant Professor, Department of Statistics, University of Kerala, in partial fulfilment of the requirements for the award of the Degree of Master of Science in Applied Statistics and Data Analytics of the University of Kerala and further that no part thereof has been presented before for any other degree.

Thiruvananthapuram FYSAL KASIM

September 2023

ACKNOWLEDGEMENT

The success of anything needs co-operation and encouragement from different quarters. Words are inadequate to express my profound and deep sense of gratitude to those who helped me in bringing out this project successfully. I wish to express my profound and deep sense of gratitude to of Dr. A. Riyaz, Assistant Professor, Department of Statistics, University of Kerala for suggesting the topic, his guidance, supervision and constant encouragement throughout the study. His valuable and inspiring suggestions, constructive criticism and moral support helped me in the successful completion of this work. I would like to thank all the teachers, the administration officer, the librarian and research scholars of our department, my parents and all my friends for their help and encouragement that they have rendered to me during the course of this project work. Above all I thank the Almighty for His blessing showered on us throughout the course of this project work

Thiruvananthapuram FYSAL KASIM

September 2023

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In today's fast-paced world, the importance of sleep and daily habits on our overall health and well-being cannot be overstated. Sleep is a fundamental physiological process that plays a crucial role in maintaining our physical and mental health. Daily habits, including lifestyle choices and daily activities, can significantly impact the quality of our sleep and overall health.

The project titled " Exploring the Dynamics of Sleep and Daily Habits: A Comprehensive Investigation Using Machine Learning " is a comprehensive investigation that harnesses the power of machine learning to gain insights into how various factors influence sleep patterns and the presence of sleep disorders. This project utilizes a rich and diverse dataset that covers a wide range of variables related to sleep and daily habits, including gender, age, occupation, sleep duration, quality of sleep, physical activity level, stress levels, BMI category, blood pressure, heart rate, daily steps, and the presence or absence of sleep disorders.

The dataset used in this project is a valuable resource for understanding the complex interplay between lifestyle factors and sleep health. It contains information about individuals' characteristics, daily routines, and health indicators. Some key attributes in the dataset include:

Gender: Indicates the gender of each individual.

Age: Represents the age of the individuals participating in the study.

Occupation: Provides information about the occupation of each individual.

Sleep Duration: Specifies the average duration of sleep-in hours.

Quality of Sleep: A measure of the perceived sleep quality.

Physical Activity Level: Indicates the level of physical activity.

Stress Level: Measures the perceived stress levels.

BMI Category: Categorizes individuals based on their Body Mass Index.

Blood Pressure: Includes systolic and diastolic blood pressure readings.

Heart Rate: Represents the heart rate in beats per minute.

Daily Steps: Records the number of steps taken daily.

Sleep Disorder: Indicates the presence or absence of sleep disorders, such as insomnia or sleep apnea bv Data Pre-processing

INSOMNIA:

Insomnia is a common sleep disorder, and it is characterized by several key features. Individuals with insomnia often encounter difficulty falling asleep at night. They may also wake up frequently during the night and struggle to return to sleep once awakened. In some cases, people with insomnia wake up too early in the morning and find themselves unable to go back to sleep. This often results in feeling unrefreshed and fatigued after a night's sleep, leading to daytime sleepiness, irritability, and difficulties in concentrating.

There are various factors that can contribute to the development of insomnia, including stress, anxiety, depression, underlying medical conditions, the use of certain medications, and poor sleep habits. Fortunately, treatment options are available to address insomnia. These treatments may include making lifestyle changes, engaging in cognitive-behavioral therapy, and, in some situations, the use of medication to help individuals improve their sleep patterns and overall sleep quality.

SLEEP APNEA:

Sleep apnea is a sleep disorder characterized by frequent interruptions in breathing during sleep. The two main types are Obstructive Sleep Apnea (OSA) and Central Sleep Apnea (CSA), Sleep apnea can have severe health consequences, increasing the risk of cardiovascular problems, daytime fatigue, and cognitive impairment. Treatments include lifestyle changes, using devices like continuous positive airway pressure (CPAP) machines or oral appliances, surgery. These disorders can greatly affect quality of life, so it's crucial to seek medical help if symptoms are suspected.

Before applying machine learning algorithms, the dataset undergoes preprocessing steps to ensure its suitability for analysis. This includes data cleaning, encoding categorical variables, handling missing values, and feature scaling. Additionally, the dataset is balanced using oversampling techniques to address potential class imbalance issues, ensuring the model's robustness.The project explores the performance of several machine learning algorithms to predict sleep disorders based on the collected data.The performance of each machine learning algorithm is evaluated using metrics such as precision, recall, F1-score, and accuracy. Hyperparameter tuning is applied to select the best parameters for specific algorithms.The final model is saved for future use. It can take input data, process it, and predict whether an individual has a sleep disorder based on their characteristics and daily habits.

In summary, this project provides valuable insights into the relationship between sleep, daily habits, and sleep disorders using machine learning. It offers a predictive model that can assist individuals and healthcare professionals in identifying potential sleep issues early, promoting better sleep health, and ultimately enhancing overall well-being.

1.2 MOTIVATION OF THE STUDY

The primary motivation driving this study stems from the recognition of the far-reaching consequences of sleep on human lives. Sleep-related issues, ranging from transient sleep disturbances to debilitating sleep disorders, have a profound impact on individuals' physical and mental health, social interactions, and professional productivity. These issues not only affect individuals but also carry broader societal and economic implications.

The motivation to delve into the dynamics of sleep and daily habits is grounded in the desire to mitigate these adverse effects. By gaining a deeper understanding of how various factors interact and influence sleep, our aim is to provide individuals and healthcare professionals with actionable insights and solutions. Ultimately, our goal is to enhance sleep health, improve sleep quality, and contribute to the overall well-being of individuals and society at large.

1.3 DATASET INFORMATION

The vital part of machine learning is the dataset used. The dataset should be as concrete as possible because a little change in the data can perpetuate massive changes in the outcome. This dataset spans a wide spectrum of variables, encompassing demographic details such as gender and age, occupational information, sleep duration, sleep quality assessments, physical activity levels, stress evaluations, BMI categorizations, blood pressure measurements, heart rate data, daily step counts, and the presence or absence of diagnosed sleep disorders.

Gender: Indicates the gender of each individual.

Age: Represents the age of the individuals participating in the study.

Occupation: Provides information about the occupation of each individual.

Sleep Duration: Specifies the average duration of sleep-in hours.

Quality of Sleep: A measure of the perceived sleep quality.

Physical Activity Level: Indicates the level of physical activity.

Stress Level: Measures the perceived stress levels.

BMI Category: Categorizes individuals based on their Body Mass Index.

Blood Pressure: Includes systolic and diastolic blood pressure readings.

Heart Rate: Represents the heart rate in beats per minute.

Daily Steps: Records the number of steps taken daily.

Sleep Disorder: Indicates the presence or absence of sleep disorders, such as insomnia or sleep apnea bv Data Pre-processing:

This rich and diverse dataset stands as the bedrock of our research, allowing us to analyze, correlate, and derive meaningful insights regarding the intricate interplay between these variables and sleep patterns.

1.4 OBJECTIVE OF THE STUDY

Our study is driven by a set of well-defined objectives:

Explanatory Data Analysis (EDA): EDA is crucial to gain a deeper understanding of the dataset. The study will explore data distributions, correlations between variables, and visualize key insights using tools like histograms, Bar chart, Pie chart, Correlation heat map. EDA can reveal trends, outliers, and potential data quality issues.

Unravel Complex Relationships: We aim to elucidate the multifaceted relationships between various factors, including age, gender, occupation, stress levels, and sleep quality. By analyzing these interactions, we seek to uncover patterns and associations that influence sleep.

Predict Sleep Disorders: We endeavor to develop predictive models that can accurately identify and anticipate sleep disorders based on the wealth of data available. These models can serve as valuable tools for early detection and intervention.

Comparison of Model Performance: To assess the effectiveness of different prediction models, the study aims to compare their performance using various evaluation metrics such as accuracy, precision, recall, and F1-score.

Identify Key Influencers: We intend to identify critical variables that exert a significant influence on sleep patterns and quality. This knowledge can aid in targeted interventions and lifestyle modifications.

Offer Practical Recommendations: Through our findings, we aim to provide practical recommendations and insights for individuals and healthcare professionals to improve sleep quality and effectively manage sleep-related issues. These recommendations are grounded in data-driven insights.

1.5 METHODOLOGY IN BRIEF

To realize our research objectives, our study adopts a systematic and data-driven methodology. This methodology encompasses several pivotal steps:

1. Data Preprocessing: Our journey commences with the meticulous preparation and cleaning of the dataset, addressing missing values, and ensuring data integrity.
2. Exploratory Data Analysis (EDA): EDA is a critical phase that involves in-depth data exploration through statistical analysis and visualization. It allows us to gain an intimate understanding of variable distributions, unveil patterns, and discern correlations.
3. Data Scaling: We employ data scaling techniques, specifically Min-Max scaling, to guarantee that all features are harmoniously positioned within a consistent range.
4. Oversampling: Given the data's inherent imbalance concerning sleep disorders, we enlist the Synthetic Minority Over-sampling Technique (SMOTE) to rectify this disparity. This step ensures equitable representation of both classes.
5. Model Development: Our research canvasses multiple machine learning algorithms, including K-Nearest Neighbors, Support Vector Classifier (SVC), Random Forest, Decision Tree Classifier. Each algorithm undergoes rigorous evaluation, and hyperparameter tuning is conducted to optimize model performance.

1.6 LIMITATIONS

Acknowledging the limitations of our study is crucial to ensuring a transparent and comprehensive perspective:

1. Self-Reported Data: The dataset relies on self-reported information, which can introduce reporting bias. While this data serves as an invaluable source of insights, it is subject to individual perceptions and potential inaccuracies.
2. External Factors: External factors, such as environmental conditions and genetic predispositions, can exert significant influence on sleep patterns. These external influences may not be fully accounted for in our analysis.
3. Snapshot of Sleep Habits: Our study provides a snapshot of sleep habits and patterns; it may not encompass long-term lifestyle changes or chronic conditions that develop over extended periods.

1.7 SUMMARY OF THE STUDY

"Exploring the Dynamics of Sleep and Daily Habits" is a comprehensive project that embarks on a multifaceted journey. It seeks to unravel the intricate connections between sleep patterns and daily routines by leveraging advanced machine learning techniques. Our study aspires to provide valuable insights into sleep health, predict sleep disorders, and offer actionable recommendations. In doing so, we aim to elevate the quality of life for individuals by promoting healthier sleep habits and addressing the pressing sleep-related issues that impact us all.

CHAPTER 2

METHODOLOGY

2.1 MACHINE LEARNING

2.1.1 INTRODUCTION

Machine learning is a subset of AI, which enables the machine to automatically learn from data, improve performance from past experiences, and make predictions. The goal of machine learning is to derive knowledge from data. Predictive analytics or statistical learning are other names for the research area that lies at the nexus of statistics, AI, and computer science. Machine Learning is a statistical term that refers to the use of existing information through an artificial intelligence application method for handling or assisting with handling statistical data. Even if Machine Learning involves automation concepts, but it also needs human direction. In machine learning, there is a high amount of generalisation to create a system that works well with data that hasn't been seen before instances.

In recent years, the use of machine learning techniques has proliferated in daily life. Machine learning algorithms are at the foundation of many contemporary websites and devices, from automatic recommendations of movies to watch, foods to order, or things to buy, to tailored online radio and recognising your pals in your images. It is extremely possible that every section of a complicated website like Facebook, Amazon, or Netflix contains numerous machine learning models. Machine learning has had a significant impact on the way data-driven research is conducted today, even outside of commercial applications.

- Computer science's relatively young field of machine learning offers a variety of methods for data analysis. Principal component analysis and logistic regression are two examples of procedures that are based on well-known statistical techniques, although many more are not.
- Most statistical methods work by selecting a specific probabilistic model from a group of related models that best fits the observed data. Similar to this, the majority of machine learning approaches aim to identify the models that best match the data (i.e., they address specific optimization issues), with the exception that these machine learning models are no longer limited to probabilistic ones.

- Consequently, a benefit of machine learning approaches over statistical ones is that the former does not necessitate the use of underlying probabilistic models. The conventional statistical techniques are frequently too rigid for the upcoming Big Data era since data sources are becoming more complicated and multifaceted, even though certain machine learning techniques use probabilistic models. It may be extremely difficult, if not impossible, to prescribe statistically valid probabilistic models that link variables from different data sources.
- A broader class of more adaptable alternative analytical techniques that are more suited to contemporary sources of data may be made available via machine learning. In order to establish whether machine learning techniques could better meet their future needs than conventional ones, statistical agencies must investigate the potential usage of these approaches.

Machine learning contains a set of algorithms that work on a huge amount of data. Data is fed to these algorithms to train them, and on the basis of training, they build the model & perform a specific task. These ML techniques support the resolution of numerous business issues, including clustering, associations, forecasting, classification, regression, and others. Most commonly Python language is used as the Analytical tool for Machine Learning. It is a object oriented Programming language.Python programs are generally smaller compared to other language.Most of the Tech Giant companies like Facebook, Google etc uses this language.

2.1.2 TYPES OF MACHINE LEARNING ALGORITHMS

Machine Learning Algorithm can be broadly classified into three types:

1. Supervised Learning Algorithms
2. Unsupervised Learning Algorithms
3. Reinforcement Learning algorithm

SUPERVISED MACHINE LEARNING

Supervised machine learning is based on supervision, as its name suggests. In the supervised learning technique, this means that we train the machines using the "labelled" dataset, and then the machine predicts the output based on the training. Here, the labelled data indicates which inputs have already been mapped to which output. More precisely, we may state that after training the machine with input and related output, we ask it to predict the outcome using test dataset.

Let's use an illustration to clarify supervised learning. Assume we have a dataset of photos of dogs and cats as our input. Therefore, we will first train the machine to comprehend the photos, teaching it things like the size and shape of a dog's tail, the shape of a cat's eyes, their colour, and their height (dogs are taller than cats, for example). After training, we input a cat image and ask the computer to recognise the object and forecast the outcome. Now that the machine is educated, it will examine every characteristic of the thing, including height, form, colour, eyes, ears, tail, and so on, and determine that it is a cat. As a result, it will be classified as a cat. This is the technique of supervised machine learning.

Steps in Supervised Learning:

- First Determine the type of training dataset
- Collect/Gather the labelled training data.
- Split the training dataset into training **dataset, test dataset, and validation dataset**.
- Determine the input features of the training dataset, which should have enough knowledge so that the model can accurately predict the output.
- Determine the suitable algorithm for the model, such as support vector machine, decision tree, etc.
- Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of training datasets.
- Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

Categories of Supervised Machine Learning

Supervised machine learning can be classified into two types, Classification and Regression.

Classification

When a classification problem has a categorical output variable, such as "Yes" or "No," Male or Female, Red or Blue, etc., classification methods are employed to solve the problem. The categories that are present in the dataset are predicted by the categorization algorithms. Spam detection, email filtering, and other examples of categorization systems in use today.

Some popular classification algorithms are given below:

- Random Forest Algorithm
- Decision Tree Algorithm
- Logistic Regression Algorithm
- Support Vector Machine Algorithm
- KNN

There are two types of classifications;

- Binary classification
- Multi-class classification

Regression

Regression problems with a linear relationship between the input and output variables are solved using regression techniques. These are employed to forecast variables with continuous outputs, such as market trends, weather forecasts, etc.

Some popular Regression algorithms are given below:

- Simple Linear Regression Algorithm
- Multivariate Regression Algorithm
- Bayesian Linear Regression
- Lasso Regression

Classification Algorithm in Supervised Learning

We predicted the output for continuous values using Regression techniques, but we require Classification methods to predict the output for categorical values. The Classification method is a Supervised Learning technique that uses training data to identify the category of new observations. A software in Classification learns from a given dataset or observations and then classifies additional observations into one of several classes or groupings. For example, Yes or No, 0 or 1, Spam or Not Spam, cat or dog, and so on. Classes can also be referred to as targets/labels or categories. In contrast to regression, the outcome variable of Classification is a category rather than a value, such as "Green or Blue", "fruit or Animal", and so on. Since the Classification algorithm is a Supervised learning technique, hence it takes labeled input data, which means it contains input with the corresponding output.

The Classification algorithm's main purpose is to identify the category of a given dataset, and these are the algorithms mostly used to forecast the outcome of categorical data.

Binary Classifier: If the classification problem has just two possible outcomes, then this classifier is used. It is known as a Binary Classifier. Examples: YES or NO, MALE or FEMALE, SPAM or NOT SPAM, CAT or DOG, etc.

Multi-class Classifier.: If a classification problem includes more than two outcomes, then a multi-class classifier is used. It's known as a Multi-class Classifier. Examples include crop classifications and music classifications.

Classification algorithms can again be divided into 2 types: Linear and Non-linear Models

1) Linear Models

- Logistic Regression
- Support Vector Machines

2) Non-linear Models

- Kernel SVM
- K-Nearest Neighbours
- Naïve Bayes
- Decision Tree Classification
- Random Forest Classification

UNSUPERVISED MACHINE LEARNING

Unsupervised learning differs from the supervised learning technique in that no supervision is required, as the name implies. In unsupervised machine learning, this means that the system is trained on an unlabelled dataset and makes output predictions without any human supervision.

In unsupervised learning, the models are trained on data that has neither been classified nor labelled, and they are then allowed to behave autonomously on that data. The unsupervised learning algorithm's primary goal is to classify or group the unsorted dataset based on commonalities, patterns, and differences. The hidden patterns in the input dataset are to be found by the machines. To better comprehend it, let's use an example. Suppose we feed the machine learning model photographs of a basket of fruit. The model has no prior knowledge of the photos, and its job is to identify patterns and groups of items. As a result, when the machine is tested with the test dataset, it will now learn its patterns and distinctions, such as colour differences and form differences, and anticipate the output. Unsupervised Learning can be further classified into two types, which are given below:

- Clustering
- Association

REINFORCEMENT LEARNING

With reinforcement learning, an AI agent (a software component) automatically explores its surroundings by striking and trailing, acting, learning from experiences, and increasing performance. Reinforcement learning operates on a feedback-based method. The objective of a reinforcement learning agent is to maximise the rewards since the agent is rewarded for every good activity and penalised for every bad action.

In contrast to supervised learning, reinforcement learning relies only on the experiences of the agents. The method of reinforcement learning is comparable to that of a human being; for instance, a youngster learns different things through encounters in his daily life. Playing a game where the environment is the game, an agent's actions at each step establish states, and the agent's objective is to score highly is an example of reinforcement learning. Agent gets feedback in the form of sanctions and benefits. Due to the way it functions, reinforcement learning is used in a variety of disciplines, including multi-agent systems, game theory, operation research, and information theory. Reinforcement learning is categorized mainly into two types of methods/algorithms:

Positive Reinforcement Learning: Positive reinforcement learning refers to the process of adding something to the needed behaviour to make it more likely that it will happen again. It strengthens the agent's behaviour and has a favourable effect on it.

Negative Reinforcement Learning: This method of learning functions in direct opposition to positive RL. By avoiding the undesirable circumstance, it makes it more likely that the particular behaviour would recur.

2.2 DATA PREPROCESSING

Preparing raw data to be acceptable for a machine learning model is known as data preparation. In order to build a machine learning model, it is the first and most important stage. It is not always the case that we come across the clean and prepared data when developing a machine learning project. Additionally, any time you work with data, you must clean it up and format it. Therefore, we use a data preprocessing activity for this. Real-world data typically includes noise, missing values, and may be in an undesirable format, making it impossible to build machine learning models on it directly. Data preprocessing is necessary to clean the data and prepare it for a machine learning model, which also improves the model's accuracy and effectiveness.

It involves below steps:

- **Getting the dataset**
- **Importing libraries**
- **Importing datasets**
- **Finding Missing Data**
- **Encoding Categorical Data**
- **Splitting dataset into training and test set**
- **Feature scaling**

Get the dataset

To create a machine learning model, the first thing we required is a dataset as a machine learning model completely works on data. The collected data for a particular problem in a proper format is known as the dataset. The file format known as "Comma-Separated Values" (CSV) enables us to save tabular data, such as spreadsheets.

Importing libraries

Python predefined libraries must be imported in order to pre-process data using Python. Some specific tasks are carried out using these libraries. We will use the following three packages specifically for data pre-processing:

NumPy: NumPy Python library is used for including any type of mathematical operation in the code. It is the fundamental package for scientific calculation in Python. It also supports to add large, multidimensional arrays and matrices. So, in Python, we can import it as: import numpy as np

Matplotlib: The second library is matplotlib, which is a Python 2D plotting library, and with this library, we need to import a sub-library pyplot. This library is used to plot any type of charts in Python for the code. It will be imported as below: import matplotlib as mpt.

Pandas: The last library is the Pandas library, which is one of the most famous Python libraries and used for importing and managing the datasets. It is an open-source data manipulation and analysis library. It will be imported as below: import pandas as pd.

Seaborn: Seaborn is another popular Python library, particularly for data visualization. It is often used in conjunction with Matplotlib and Pandas.We can create more visually appealing and informative plots and charts for your data analysis tasks. Seaborn provides a high-level interface for drawing attractive and informative statistical graphics. It's commonly used to enhance Matplotlib visualizations and to simplify common data visualization tasks.

Scikit-learn: is a Python library for machine learning. It provides easy-to-use tools for tasks like classification, regression, clustering, and data pre-processing. It's great for beginners and experts alike, with consistent APIs and a wide range of machine learning algorithms

Importing datasets:

To import the dataset, we will use read_csv() function of pandas library, which is used to read a csv file and performs various operations on it. Using this function, we can read a csv file locally as well as through an URL.

Handling Missing data:

The handling of missing data in the datasets is the next stage of data preprocessing. Our machine learning model may run into serious issues if our dataset has some missing data. As a result, the dataset contains missing values, which must be handled. The two primary approaches to dealing with missing data are as follows:

By removing the specific row: The first approach is frequently used to handle null data. In this manner, we simply remove the particular row or column that contains null data. However, this approach is ineffective because deleting data can result in information loss, which would result in inaccurate output.

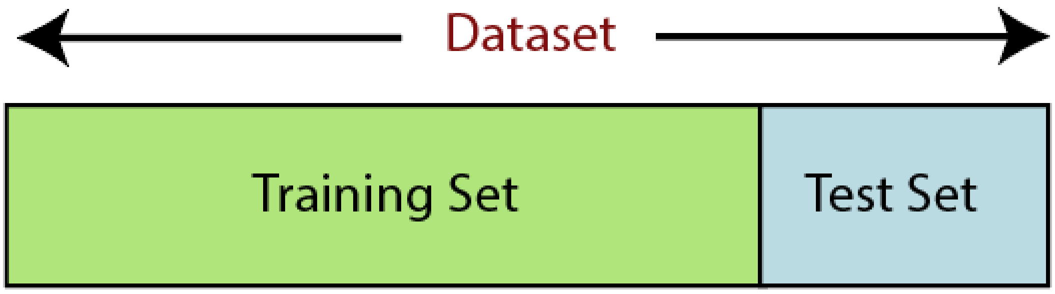
By computing the mean: In this manner, we will locate the missing value by calculating the mean of the column or row that contains the missing value. This method works well for characteristics that contain numerical information, such as age, salary, year, etc. Here, we'll apply this strategy. To handle missing values, we will use **Scikit-learn** library in our code, which contains various libraries for building machine learning models.

Encoding Categorical Data

Data with categories is referred to as categorical data.Since machine learning models are entirely based on arithmetic and numbers, if our dataset had a categorical variable, it might be difficult to construct the models. Therefore, these category variables must be encoded as numbers. we have imported **LabelEncoder** class of **sklearn library**. This class has successfully encoded the variables into digits

Splitting the Dataset into the Training set and Test set

We separate our dataset into a training set and test set during the machine learning data preprocessing phase. One of the most important data pretreatment stages, since it allows us to improve the functionality of our machine learning model.Imagine that we trained our machine learning model using one dataset, and then we tested it using a completely other dataset. The understanding of the correlations between the models will then become challenging for our model.If we train our model really well and it has a high training accuracy, but we give it a fresh dataset, the performance will suffer. Therefore, we constantly strive to create a machine learning model that excels with the training set and also with the test dataset.We can define the dataset as:



Training Set: A subset of dataset to train the machine learning model, and we already know the output.

Test set: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

For splitting the dataset, we will use the below lines of code:

```
1) from  sklearn.model_ selection import train_test_split

2)x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.2, random_state=0)
```

- In the above code, the first line is used for splitting arrays of the dataset into random train and test subsets.
- In the second line, we have used four variables for our output that are
    - **x_train:** features for the training data
    - **x_test:** features for testing data
    - **y_train:** Dependent variables for training data
    - **y_test:** Independent variable for testing data
- In **train_test_split() function**, we have passed four parameters in which first two are for arrays of data, and **test_size** is for specifying the size of the test set. The test_size maybe .5, .3, or .2, which tells the dividing ratio of training and testing sets.
- The last parameter **random_state** is used as a random generator so that you always get the same result, and the most used value for this is 42.

2.3 EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.

DATA VISUALIZATION TOOLS

2.3.1 HISTOGRAM

A histogram, which can be a bar plot where each bar reflects the frequency (count) or proportion (count/total count) of cases for various values, is the most fundamental type of graph. One of the simplest ways to immediately understand your data's central tendency, spread, modality, shape, and outliers is by using histograms..

2.3.2 HEATMAP

In the form of colourful maps, heatmaps provide a 2-dimensional visualisation of the data.The colour maps accomplish colour variation to represent different details by varying hue, saturation, or brightness. Readers are given visual indications regarding the magnitude of numerical numbers by this colour change. Because the human brain comprehends pictures better than numbers, text, or any other written data. Heatmaps replaces numbers with colours. Because people learn best visually, it makes more sense to visualise data in any format. Data is presented in an understandable way through heatmaps. HeatMaps and other visualisation techniques have so gained popularity. Heatmaps can display patterns, variance, and even anomalies while describing the density or intensity of various variables. Relationships between variables are depicted via heatmaps. On both axes, these variables are plotted. By observing the colour shift, we search the cell for patterns. Only numeric data is accepted, and it plots that data on a grid while presenting different data values using shifting colour intensity. Exploratory Data Analysis is used to highlight their key characteristics, frequently using visual techniques like Heatmaps. The visual representation of correlations between variables in high dimensional space using heatmaps is fascinating. The variable vs. itself on the diagonal and feature variables as row and column headings can be used to do this.

2.3.3 BAR GRAPH

The visual display of data (often grouped) in the shape of vertical or horizontal rectangular bars, with the length of the bars corresponding to the measure of the data, is called a bar graph. Bar charts are another name for them. One tool used in statistics for processing data is the bar graph.

2.3.4 PIE CHART

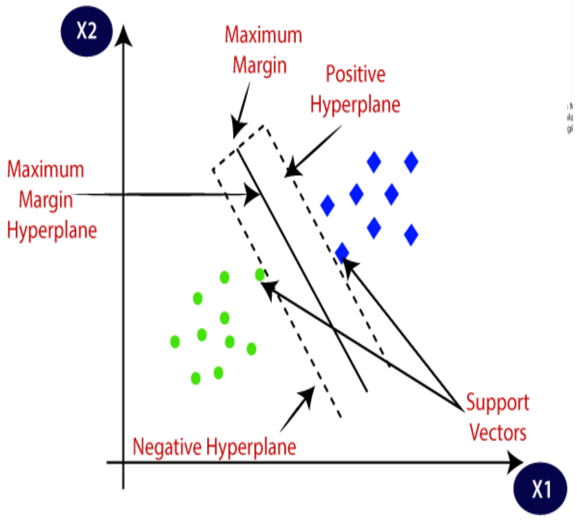
Pie charts are circular charts with segments that each indicate a value and are used in data handling. Pie charts have portions (or "slices") that each reflect a different size of value. Pie charts indicate percentages at a specific point in time and can be used to display percentages of an entire group. Pie charts do not depict changes over time, in contrast to bar graphs and line graphs. Pie charts and bar charts are both used to graph categorical data.

2.4 ML ALGORITHMS

2.4.1 SUPPORT VECTOR MACHINES

One of the most well-liked supervised learning algorithms, Support Vector Machine, or SVM, is used to solve Classification and Regression problems. However, it is largely employed in Machine Learning Classification issues.

The SVM algorithm's objective is to establish the best line or decision boundary that can divide n-dimensional space into classes, allowing us to quickly classify fresh data points in the future. A hyperplane is the name given to this optimal decision boundary. SVM selects the extreme vectors and points that aid in the creation of the hyperplane. Support vectors, which are used to represent these extreme instances, form the basis for the SVM method. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



SVM algorithm can be used for Face detection, image classification, text categorization, **etc.**

Hyperplane and Support Vectors in the SVM algorithm:

In n-dimensional space, there may be several lines or decision boundaries used to divide classes; however, the optimal decision boundary for classifying the data points must be identified. The hyperplane of SVM is a name for this optimal boundary.

The dataset's features determine the hyperplane's dimensions, therefore if there are just two features (as in the example image), the hyperplane will be a straight line. Additionally, if there are three features, the hyperplane will only have two dimensions.

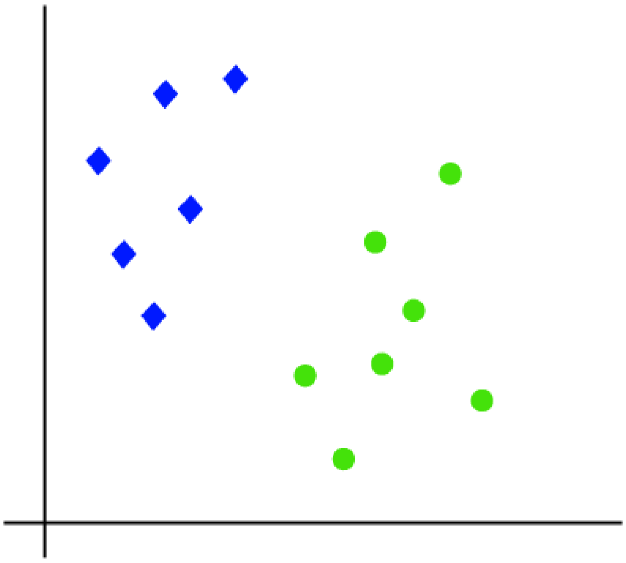
We always build a hyperplane with a maximum margin, or the greatest possible separation between the data points.

Support Vectors:

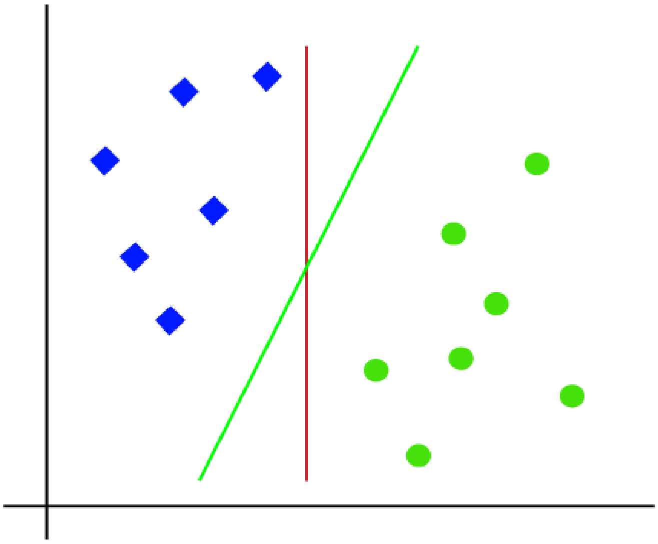
Support vectors are the data points or vectors that are closest to the hyperplane and have the greatest influence on where the hyperplane is located. These vectors are called support vectors because they support the hyperplane.

How does SVM works?

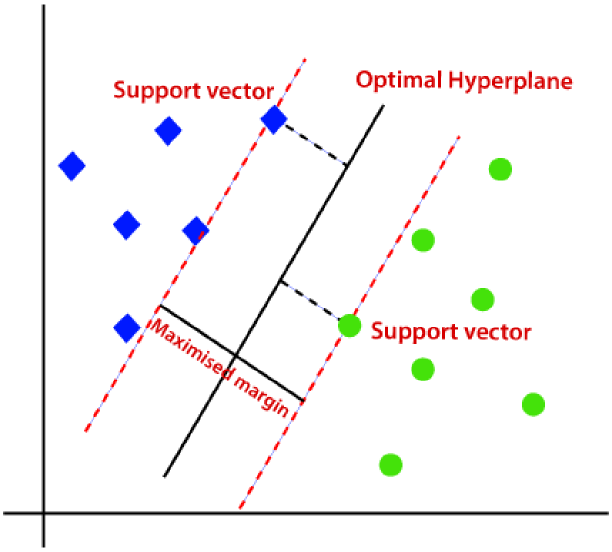
By presenting an example, the SVM algorithm's operation can be better understood. Consider a dataset with two tags (green and blue), two features (x1 and x2). We need a classifier that can identify whether the pair of coordinates (x1, x2) is blue or green**.** Consider the below image:



So as it is 2- dimentional space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the below image:



As a result, the SVM method aids in identifying the ideal decision boundary or region, often known as a hyperplane. The SVM algorithm determines which line from each class is closest to the other. Support vectors are the names for these points. Margin is the distance between the hyperplane and the vectors. Maximizing this margin is the aim of SVM. The term "optimal hyperplane" refers to the hyperplane with the largest margin.



2.4.2 K-NEAREST NEIGHBOUR

One of the simplest machine learning algorithms, based on the supervised learning method, is K-Nearest Neighbour.The K-NN algorithm makes the assumption that the new case and the existing cases are comparable, and it places the new instance in the category that is most like the existing categories.

A new data point is classified using the K-NN algorithm based on similarity after all the existing data has been stored. This means that utilising the K-NN method, fresh data can be quickly and accurately sorted into a suitable category.

Although the K-NN approach is most frequently employed for classification

problems, it can also be utilised for regression.

Since K-NN is a non-parametric technique, it makes no assumptions about the underlying data.

The KNN method simply saves the information during the training phase, and when it receives new data, it categorises it into a category that is quite similar to the new data.

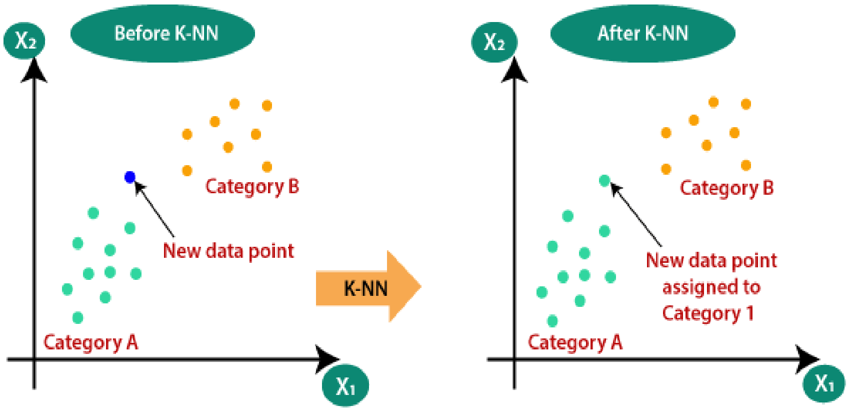
Example: Let's say we have a picture of a species that resembles both cats and dogs, but we aren't sure if it is one or the other. Therefore, since the KNN algorithm is based on a similarity metric, we can utilise it for this identification. Our KNN model will look for similarities between the new data set's features and those in the photos

of cats and dogs, and based on those similarities, it will classify the new data set as either cat- or dog-related.

Need of a K-NN Algorithm

If there are two categories, Category A and Category B, and we have a new data point, x1, which category does this data point belong in? We require a K-NN algorithm to address this kind of issue. K-NN makes it simple to determine the category or class of a given dataset.

Consider the below diagram:



WORKING OF KNN ALGORITHM

The following algorithm can be used to describe how the K-NN works:

Step-1: Select the number K of the neighbors

Step-2: Calculate the Euclidean distance of K number of neighbors

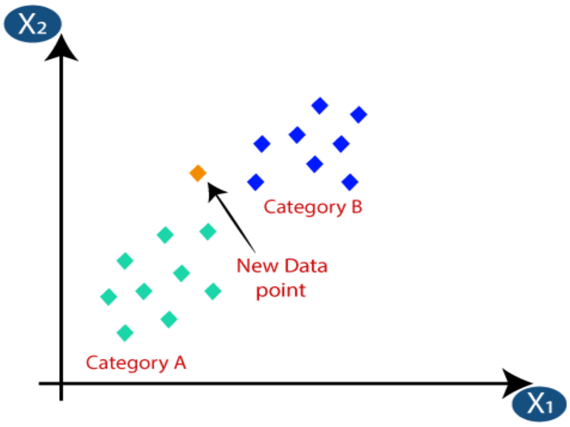
Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

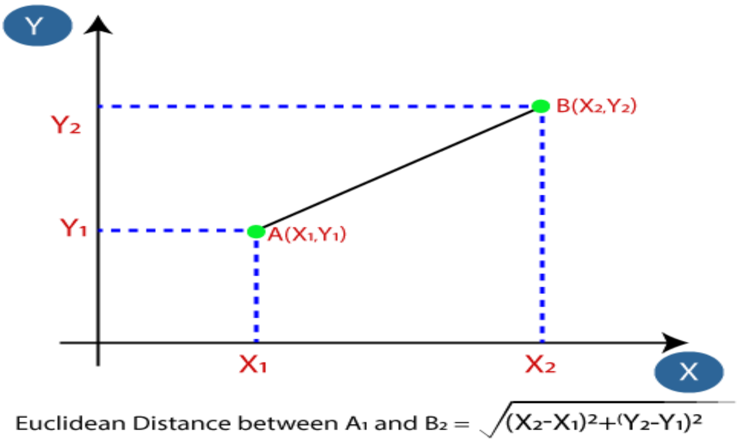
Step-6: Our model is ready.

Let's say we need to classify a new data point in order to use it. Consider the photo below:

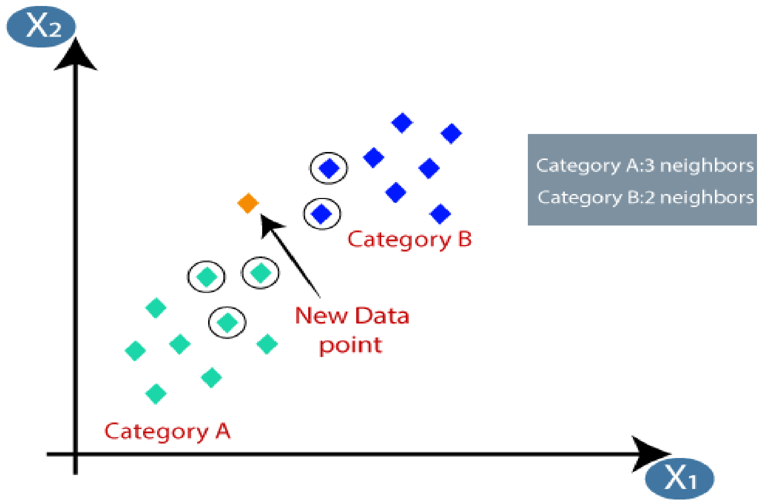


First, we'll decide on the number of neighbours; we'll go with k=5.

The Euclidean distance between the data points will then be determined. The distance between two points, which we have already examined in geometry, is known as the Euclidean distance. It is calculable as follows:


Y
Y_2
$B(X_2,Y_2)$
Y_1
$A(X_1,Y_1)$
X_1
X_2
X
Euclidean Distance between A_1 and B_2 = $\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$

The closest neighbours were determined by calculating the Euclidean distance. There were three closest neighbours in category A and two closest neighbours in category B. Consider the photo below:



As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

How to select K Value in KNN Algorithm

The following are some things to keep in mind while choosing K's value in the K-NN algorithm:

The ideal value for "K" cannot be determined in a specific fashion, thus we must experiment with different values to find the one that works best. K is best represented by the number 5.

A relatively small number of K, such K=1 or K=2, might be noisy and cause outlier effects in the model.

2.4.3 DECISION TREE ALGORITHM

A supervised learning method called a decision tree can be used to solve classification and regression problems, but it is typically favoured for doing so. It is a tree-structured classifier, where internal nodes stand in for a dataset's features, branches for the decision-making process, and each leaf node for the classification result.

The Decision Node and Leaf Node are the two nodes of a decision tree. While Leaf nodes are the results of decisions and do not have any more branches, Decision nodes are used to create decisions and have numerous branches.

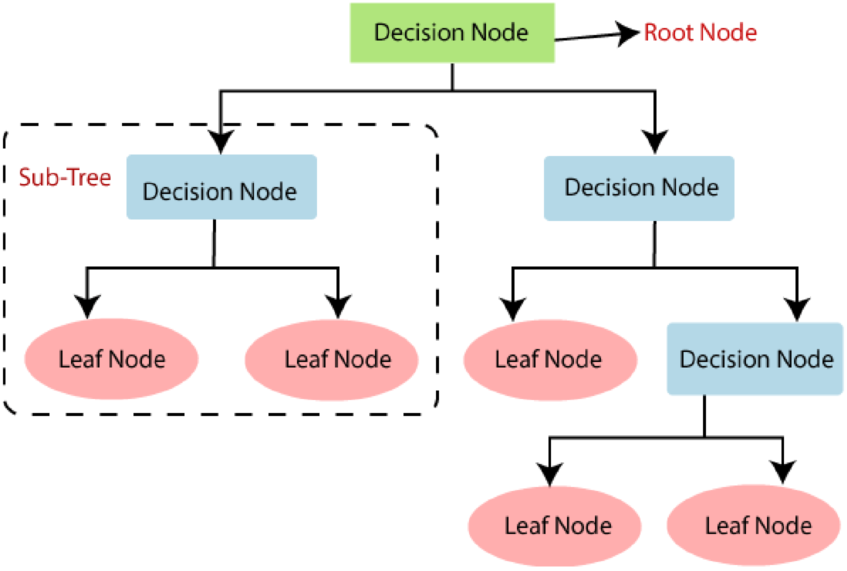
The decisions or the test are performed on the basis of features of the given dataset. It is a graphical depiction that shows all options for solving a dilemma or making a choice in light of certain parameters.

Because it begins with the root node and develops on additional branches to form a structure resembling a tree, it is known as a decision tree.

We employ the CART algorithm, or Classification and Regression Tree algorithm, to construct a tree.

Simply said, a decision tree poses a question and divides the tree into subtrees based on the response (Yes/No).

- Below diagram explains the general structure of a decision tree:



Reason for using Decision Trees

The most important thing to keep in mind when developing a machine learning model is to select the best algorithm for the dataset and problem at hand. The two rationales for employing the decision tree are as follows:

Decision trees are typically designed to resemble how people think when making decisions, making them simple to comprehend.

Because the decision tree displays a tree-like structure, the rationale behind it is simple to comprehend.

Decision Tree Terminologies

Root Node:The decision tree's root node is where it all begins. The full dataset is represented, which is then split into two or more homogeneous sets.

Leaf Node: Leaf nodes are the ultimate output nodes, after which the tree cannot be further divided.

Splitting: The division of the decision node/root node into sub-nodes in accordance with the specified conditions is known as splitting.

A branch or subtree is a tree created by slicing another tree.

Pruning is the procedure of removing the tree's undesirable branches.

Parent/Child node: The root node of the tree is referred to as the parent node, while the other nodes are referred to as the child nodes.

Working of Decision Tree Algorithm

In a decision tree, the algorithm begins at the root node and works its way up to forecast the class of the given dataset. This algorithm follows the branch and jumps to the following node by comparing the values of the root attribute with those of the record (real dataset) attribute.

The algorithm verifies the attribute value with the other sub-nodes once again for the following node before continuing. It keeps doing this until it reaches the tree's leaf node. The following algorithm can help you comprehend the entire procedure:

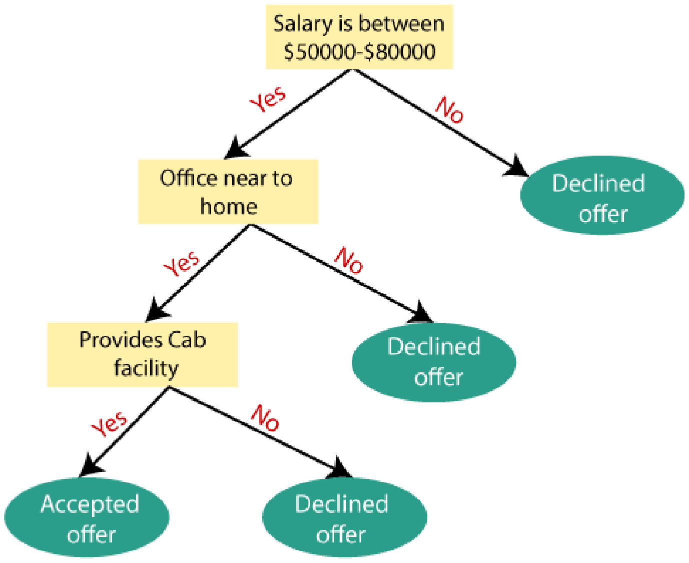
Step 1: According to S, start the tree at the root node, which has the entire dataset.

Step 2: Utilize the Attribute Selection Measure to identify the dataset's top attribute (ASM).

Step 3: Subset the S to include potential values for the best qualities.

Create the decision tree node that has the best attribute in step four.

Use the selections of the dataset generated in step 3 to iteratively develop new decision trees in step 5. Continue along this path until you reach a point when you can no longer categorise the nodes and you refer to the last node as a leaf node.



2.4.4 RANDOM FOREST

It is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random Forest classifier:

o There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.

o The predictions from each tree must have very low correlations.

Random Forest work in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase. The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

2.5 USED PERFORMANCE MEASURES

2.5.1 CONFUSION MATRIX IN MACHINE LEARNING

The performance of the classification models for a certain set of test data is evaluated using a matrix called the confusion matrix. Only after the true values of the test data are known can it be determined. Although the matrix itself is simple to understand, some of the terminology used in connection with it might be. It is also referred to as an error matrix since it displays the errors in the model performance as a matrix. The following list of Confusion matrix features includes:

The matrix is a 2*2 table for the classifiers' two prediction classes, a 3*3 table for the next three classes, and so on.

The matrix has two dimensions: actual values and expected values, as well as the total number of predictions.

Actual values are the real values for the provided data, whereas projected values are the values that the model predicts.

It looks like the below table:

n=total predictions	Actual: No	Actual: Yes
Predicted: No	True Negative	False Positive
Predicted: Yes	False Negative	True Positive

The above table has the following cases:

True Negative: The model predicted No, and the actual or true value likewise indicated No.

True Positive: Both the actual number and the model's prediction were accurate.

False Negative: This error is sometimes referred to as a Type-II error since the model predicted no, but the actual value was yes.

False Positive: Although the model expected Yes, the actual result was No. Another name for it is a Type-I error.

Need for Confusion Matrix in Machine learning

It assesses how well classification models perform when they make predictions based on test data and indicates how effective our classification model is.

It not only identifies the classification error but also the specific sort of error, such as type-I or type-II error.

We may compute the various model parameters, such as accuracy, precision, etc., using the confusion matrix.

Calculations using Confusion Matrix:

Using this matrix, we may calculate the model's accuracy as well as other things. The following computations are provided:

Classification Accuracy: This is a crucial factor in figuring out how accurate a problem's classification is. It specifies how frequently the model predicts the right result. The number of accurate predictions made by the classifier divided by the total number of predictions made by the classifiers can be used to compute it. The following is the formula:

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

Misclassification rate: Misclassification rate also known as error rate, expresses how frequently the model makes incorrect predictions. The ratio of wrong guesses to all of the classifier's predictions can be used to compute error rate. The following is the formula:

Error rate=$\frac{FN+FP}{TP+TN+FP+FN}$

Precision: Precision can be characterised as the proportion of the model's outputs that were accurate, or as the proportion of the model's correctly anticipated positive classes that really occurred. Using the formula below, it can be calculated:

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

Recall: It is defined as the out of total positive classes, how our model predicted correctly. The recall must be as high as possible.

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

F-measure: If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the precision. It can be calculated using the below formula:

F-measure=$\frac{2*recall*precision}{recall+precision}$

CHAPTER 3

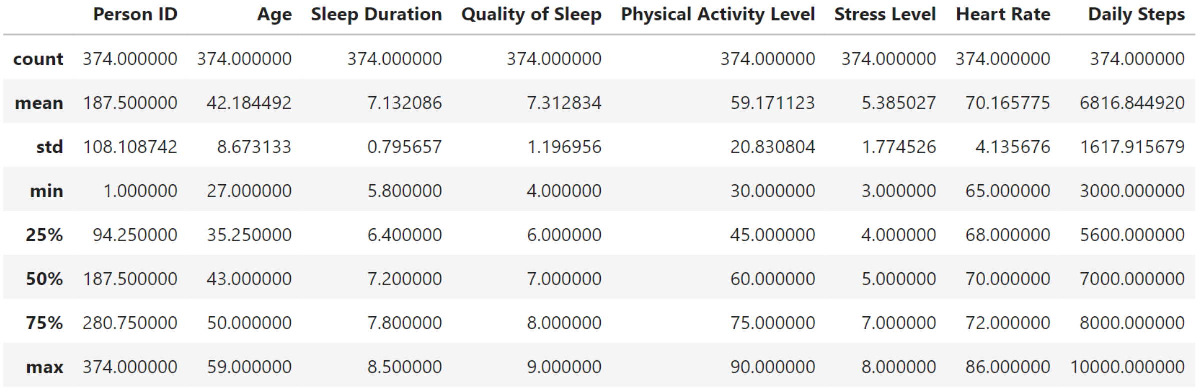
ANALYSIS OF THE DATA

3.1 INTRODUCTION

This chapter includes various data analyses. The performance of four distinct categorization models is evaluated. In EDA, descriptive statistics of the data are given as tables, and histograms and bar graphs are used to show the data graphically. Additionally, a heatmap is used to show the relationship between the attributes. Using the classification report and confusion matrix, the performance of the classification model is evaluated.

3.2 EXPLORATORY DATA ANALYSIS

3.2.1 DESCRIPTIVE STATISTICS



	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Heart Rate	Daily Steps
count	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000	374.000000
mean	187.500000	42.184492	7.132086	7.312834	59.171123	5.385027	70.165775	6816.844920
std	108.108742	8.673133	0.795657	1.196956	20.830804	1.774526	4.135676	1617.915679
min	1.000000	27.000000	5.800000	4.000000	30.000000	3.000000	65.000000	3000.000000
25%	94.250000	35.250000	6.400000	6.000000	45.000000	4.000000	68.000000	5600.000000
50%	187.500000	43.000000	7.200000	7.000000	60.000000	5.000000	70.000000	7000.000000
75%	280.750000	50.000000	7.800000	8.000000	75.000000	7.000000	72.000000	8000.000000
max	374.000000	59.000000	8.500000	9.000000	90.000000	8.000000	86.000000	10000.000000

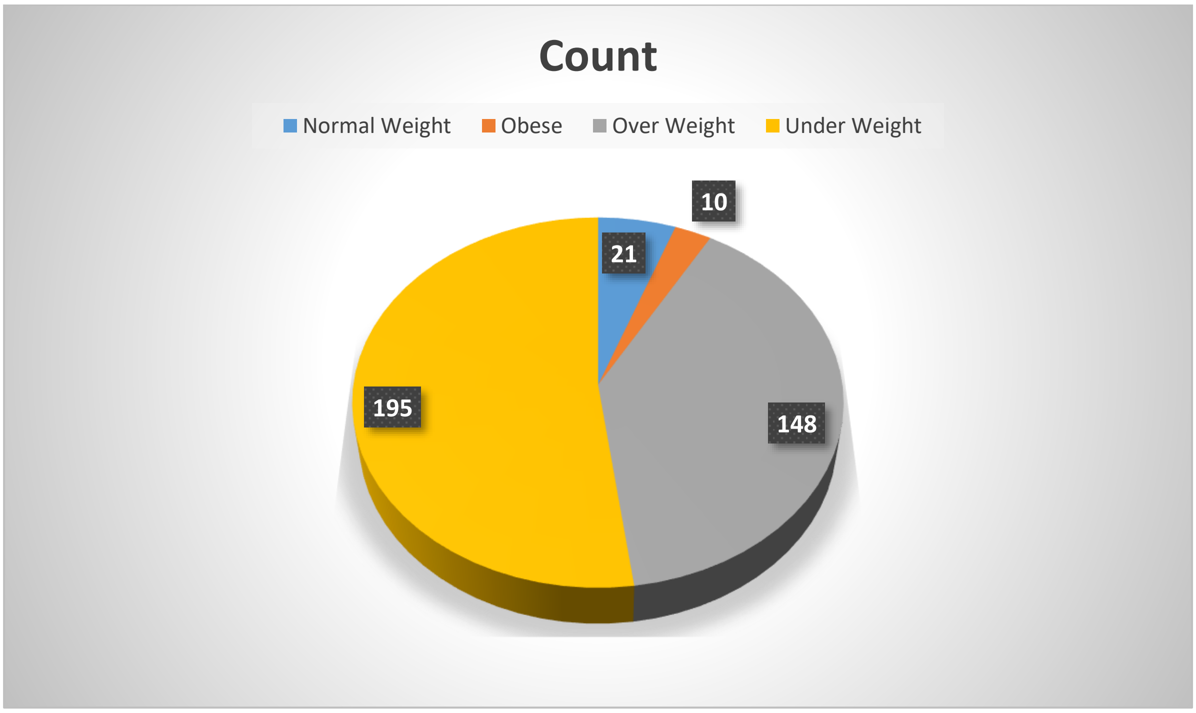
The provided data represents various attributes of 374 individuals, including their age, sleep duration, quality of sleep, physical activity level, stress level, heart rate, and daily steps.

On average, the individuals in this dataset are around 42 years old and report a mean sleep duration of approximately 7.13 hours per night, with an average sleep quality rating of 7.31 out of 10. The average physical activity level is approximately 59.17, while the stress level averages at 5.39 on a scale from 1 to 10. In terms of health indicators, the dataset reveals an average heart rate of approximately 70.17 beats per minute, and individuals take an average of 6817 steps daily. Additionally, the dataset provides some insights into the spread of these attributes. For instance, the age of individuals ranges from 27 to 59 years, while sleep duration spans from 5.8 to 8.5 hours. The stress level varies from 3 to 8, and daily steps range from 3000 to 10,000.

3.2.2 GRAPHICAL REPRESENTATION OF DATA

PIE CHART

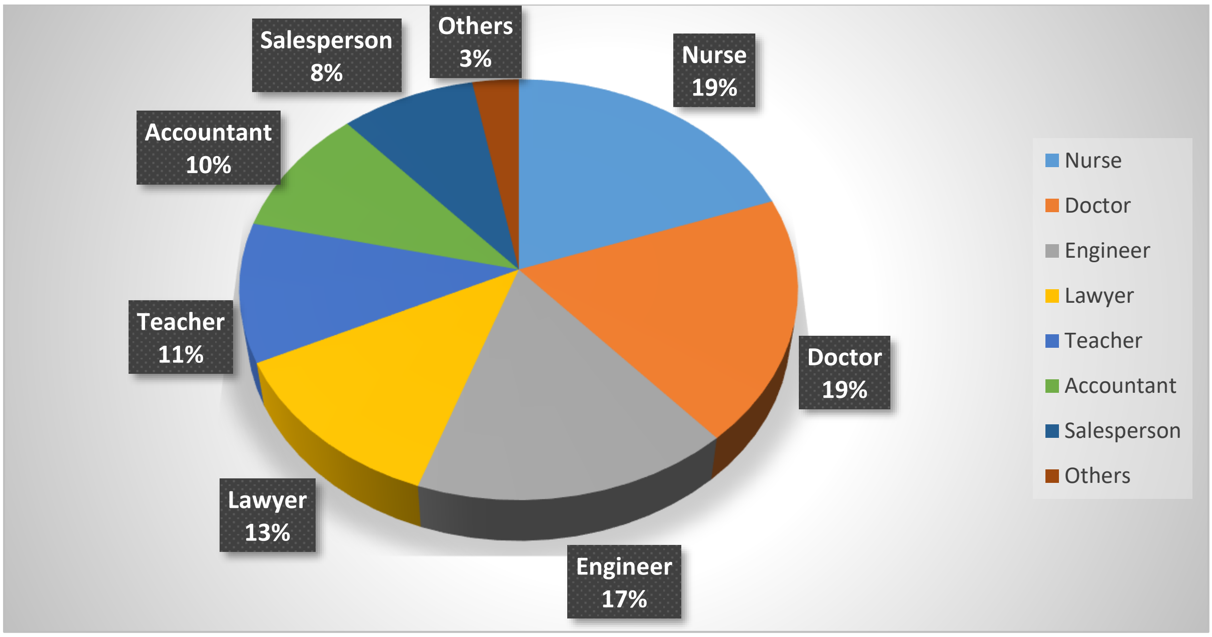
BMI CATEGORY



INFERENCE

The pie chart depicts the weight status distribution of a population based on four categories: "Under Weight," "Overweight," "Normal," and "Obese." The majority of individuals fall into the "Under Weight" and "Overweight" categories, with 195 and 148 individuals, respectively, indicating that a significant portion of the population is within a healthy or slightly overweight range. A smaller proportion of 21 individuals is categorized as "Normal," suggesting some variation in classification criteria. The "Obese" category has the fewest individuals, with only 10, indicating that a relatively small portion of the population falls into the obese category. However, the interpretation should consider the specific criteria used for these weight classifications.

COUNT OF OCCUPATION



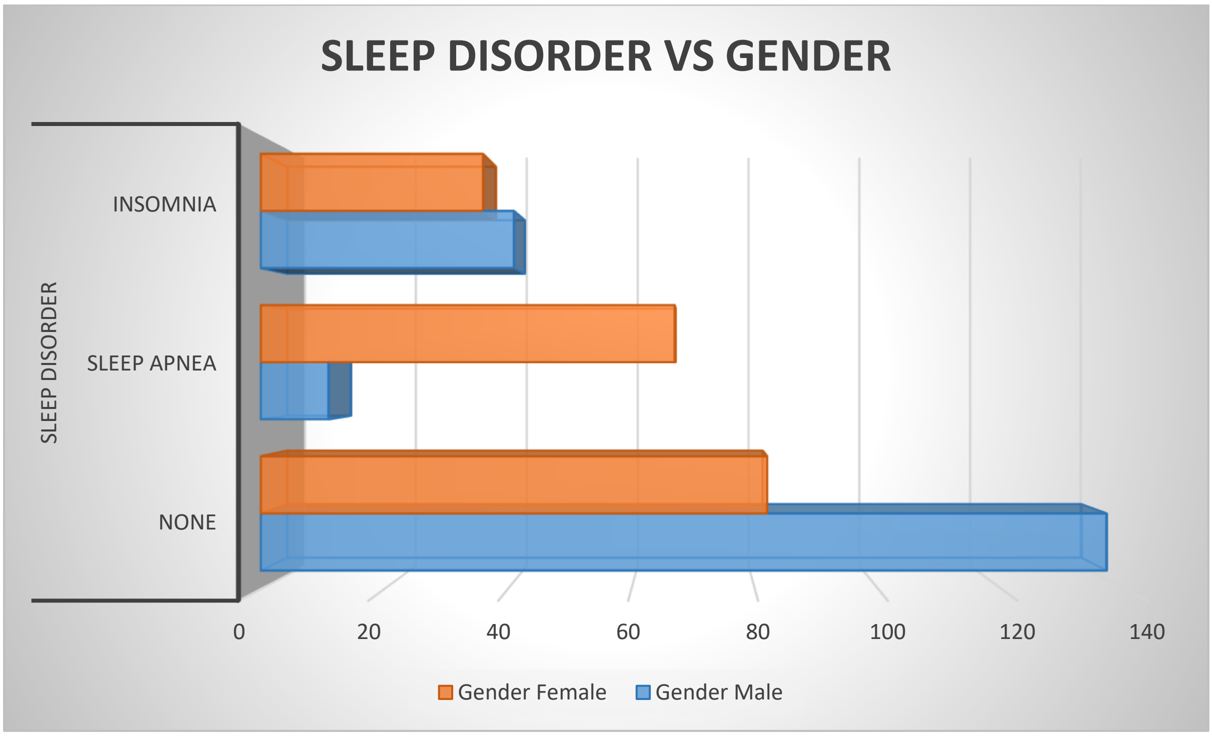
INFERENCE

The provided Pie chart offers a clear visual representation of the distribution of individuals across ten different professions. At the top of the chart, the profession with the highest representation is "Nurse," with 73 individuals, followed closely by "Doctor" with 71 individuals. "Engineer" ranks third with 63 individuals, and "Lawyer" is notable with 47 individuals. The teaching profession is also well-represented, with 40 individuals categorized as "Teacher." "Accountant" and "Salesperson" follow suit with 37 and 32 individuals, respectively.

On the other hand, "Scientist" and "Software Engineer" are less common professions, each consisting of only 4 individuals. Meanwhile, "Sales Representative" is even less prevalent, with only 2 individuals. At the bottom of the chart is "Manager," the least common profession in this dataset, with just 1 individual.

BAR CHART

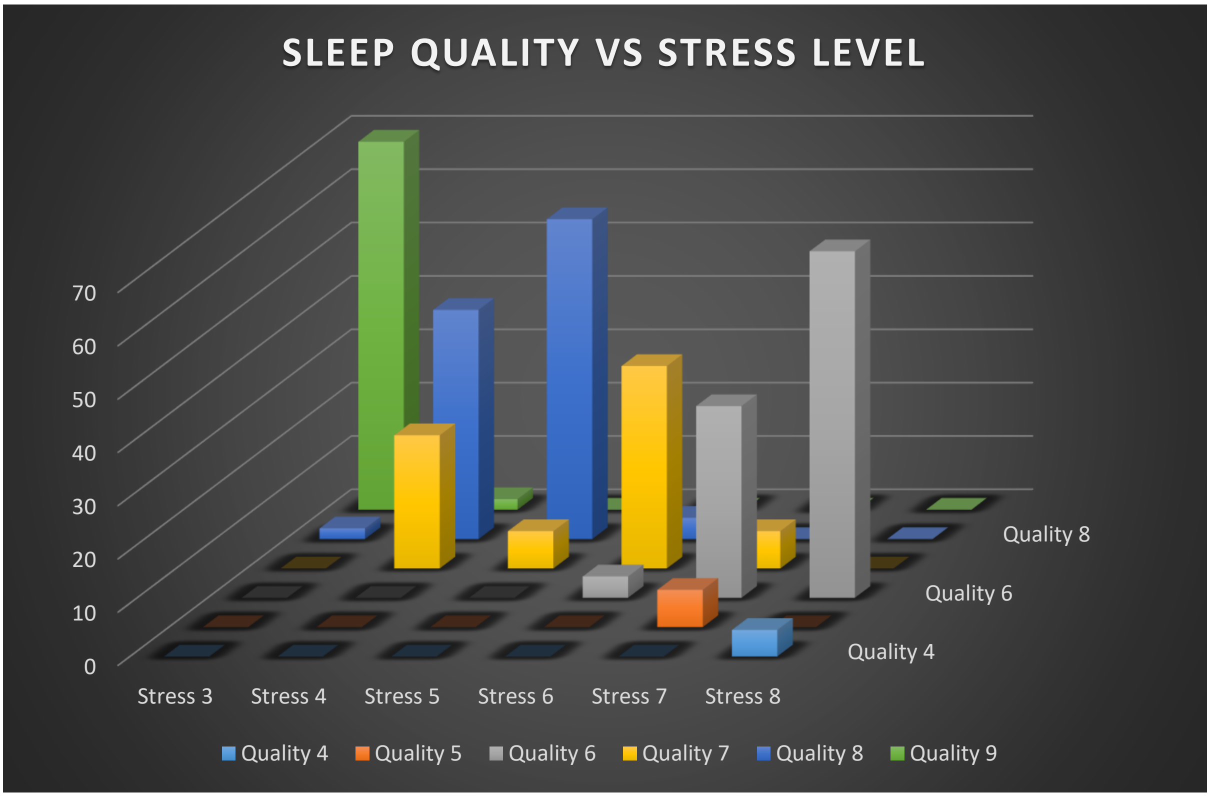
BAR CHART SLEEP DISORDER VS GENDER



INFERENCE

The bar chart illustrates the distribution of sleep disorders among males and females, with a focus on three distinct categories: "None," "Sleep Apnea," and "Insomnia.". In the "None" category, both males and females have a substantial number of cases, with 137 for males and 82 for females. This suggests that a significant portion of both genders in the dataset does not have a diagnosed sleep disorder.Moving on to "Sleep Apnea,". it's clear that this disorder is more prevalent among males, as there are 11 cases in males compared to 67 in females. This sizable gender gap indicates that sleep apnea is notably more common in females within this dataset.In contrast, when examining "Insomnia," there is a higher incidence among males, with 41 cases, in comparison to 36 cases in females. This implies that insomnia is slightly more prevalent among males in this dataset.

SLEEP QUALITY VS STRESS LEVEL

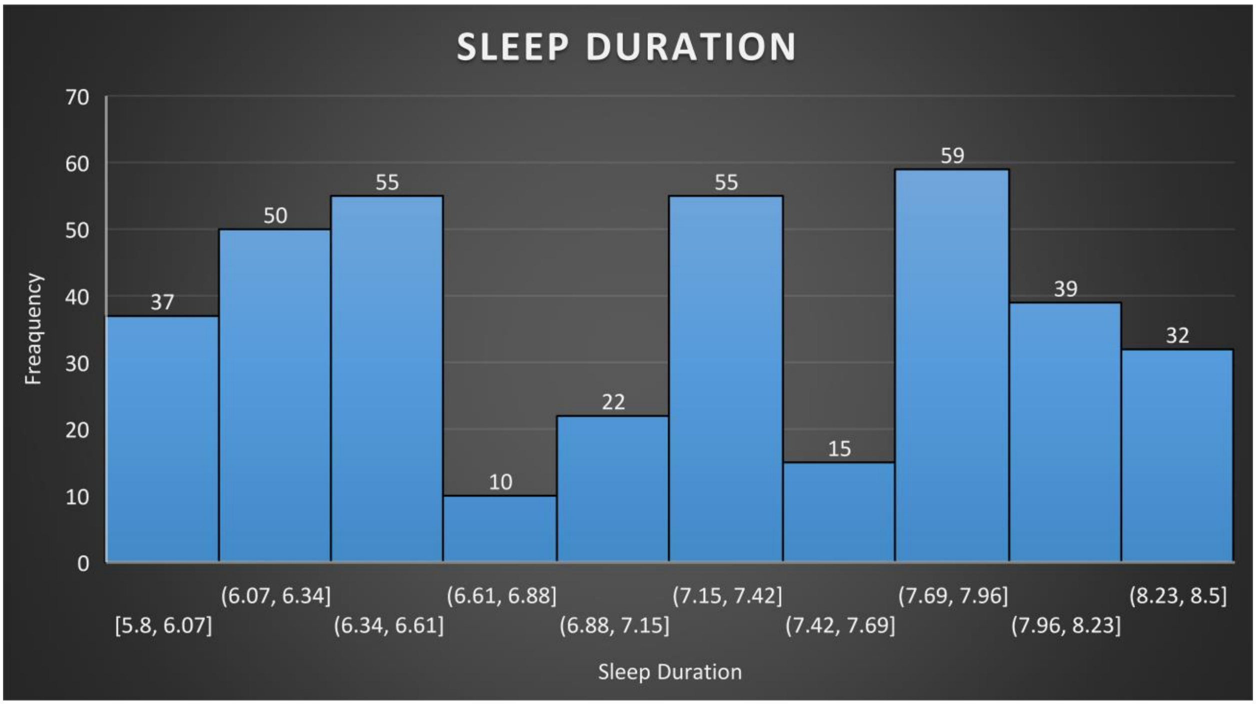


INFERENCE

This bar chart provides insights into the relationship between sleep quality and stress level, highlighting how different levels of sleep quality are associated with different levels of stress. It suggests that lower sleep quality is more often associated with higher stress levels, while better sleep quality is more often associated with lower stress levels, with variations in between.This indicates that as stress levels increase, the reported quality of sleep tends to decrease significantly. High stress is associated with lower sleep quality.

HISTOGRAM

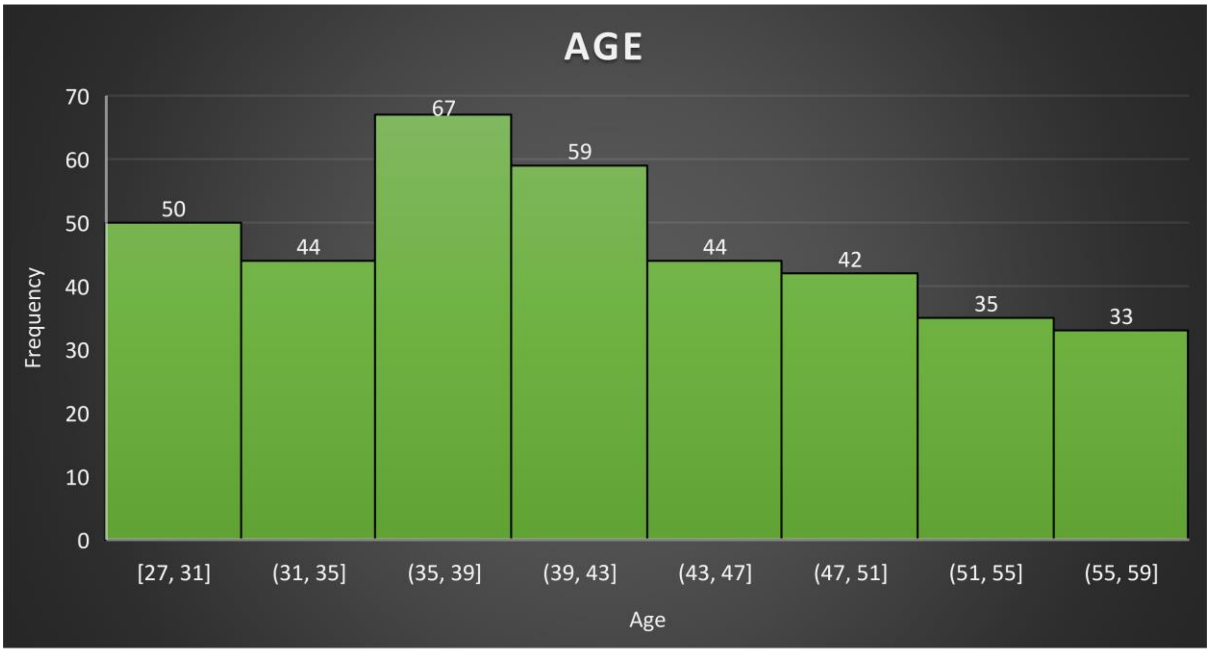
DISTRIBUTION OF SLEEP DURATION



INFERENCE

The histogram represents the distribution of sleep duration values within the dataset. The x-axis shows the range of sleep duration values, while the y-axis indicates the frequency (count) of each sleep duration range.The first peak, centered around 6 hours, suggests that a significant number of individuals in the dataset tend to have sleep durations clustered around this value. The second, larger peak is centered around 7.15 to 7.42 hours, indicating that a substantial portion of the dataset also experiences sleep durations within this range. The largest peak is centered around 7.69 to 7.96 hours.Further analysis would be needed to explore the factors contributing to these observed sleep duration patterns.

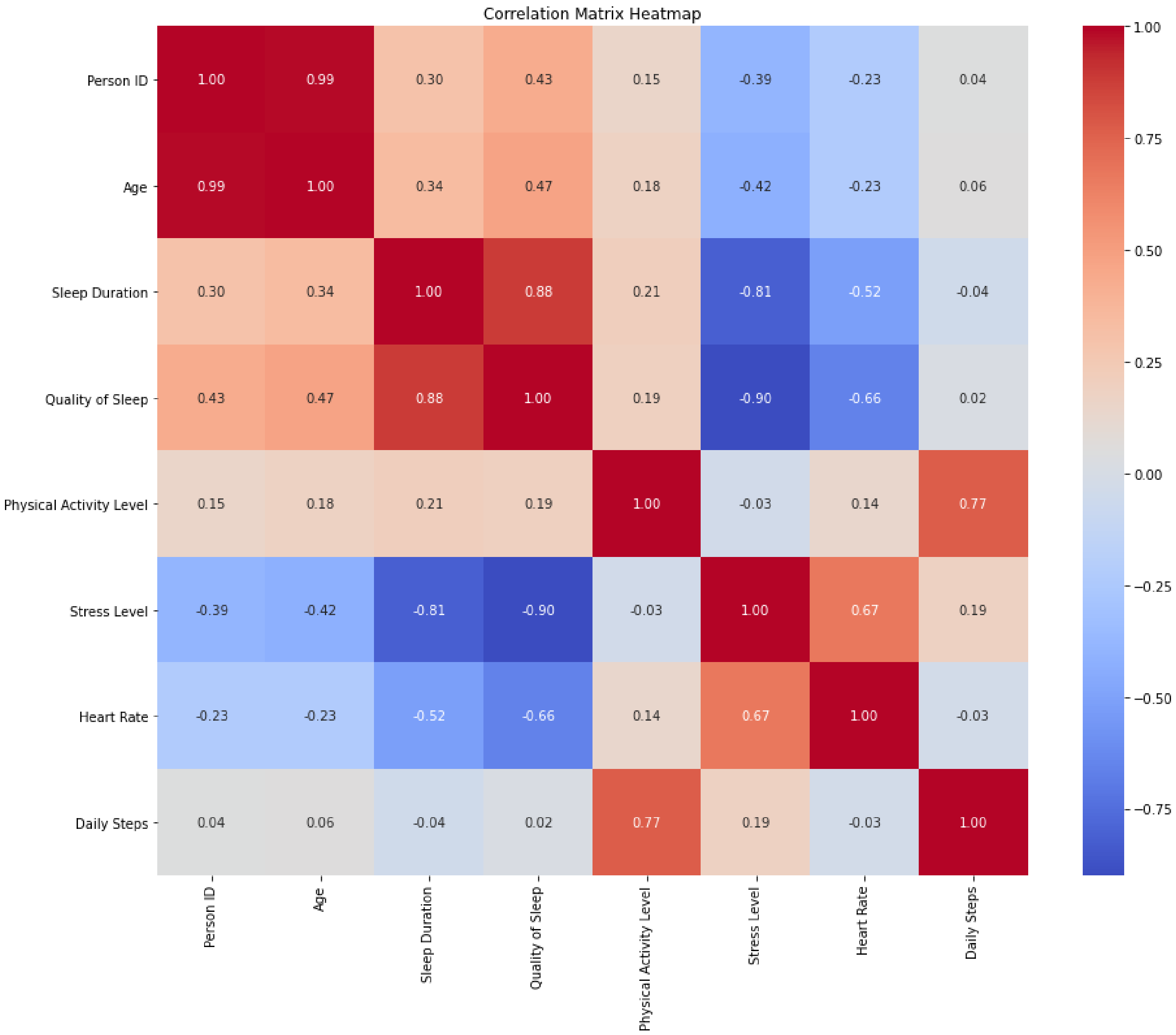
DISTRIBUTION OF AGE



INFERENCE

The histogram indicates a prominent peak in the 35-39 age group, implying that a significant portion of the dataset falls within this age range. This peak suggests that the sample population is primarily composed of individuals in their late 30s, potentially reflecting a specific demographic or target group for analysis. The histogram still maintains a generally normal distribution, with fewer individuals in both younger and older age groups.

CORRELATION HEAT MAP



	Person ID	Age	Sleep Duration	Quality of Sleep	Physical Activity Level	Stress Level	Heart Rate	Daily Steps
Person ID	1.00	0.99	0.30	0.43	0.15	-0.39	-0.23	0.04
Age	0.99	1.00	0.34	0.47	0.18	-0.42	-0.23	0.06
Sleep Duration	0.30	0.34	1.00	0.88	0.21	-0.81	-0.52	-0.04
Quality of Sleep	0.43	0.47	0.88	1.00	0.19	-0.90	-0.66	0.02
Physical Activity Level	0.15	0.18	0.21	0.19	1.00	-0.03	0.14	0.77
Stress Level	-0.39	-0.42	-0.81	-0.90	-0.03	1.00	0.67	0.19
Heart Rate	-0.23	-0.23	-0.52	-0.66	0.14	0.67	1.00	-0.03
Daily Steps	0.04	0.06	-0.04	0.02	0.77	0.19	-0.03	1.00

INFERENCE

The provided correlation matrix reveals the relationships between various variables in your dataset. Each cell in the matrix represents the correlation coefficient, which measures the strength and direction of the linear relationship between two variables. Here are some key insights and inferences based on the correlations:

Quality of Sleep vs. Sleep Duration (0.88): Quality of Sleep and Sleep Duration exhibit a strong positive correlation (0.88). This suggests that individuals who sleep longer tend to report higher sleep quality, indicating a positive relationship between these two factors.

Stress Level vs. Quality of Sleep (-0.90): Stress Level and Quality of Sleep have a strong negative correlation (-0.90). This indicates that as stress levels increase, the reported quality of sleep tends to decrease significantly. High stress is associated with lower sleep quality.

Stress Level vs. Sleep Duration (-0.81): Stress Level and Sleep Duration exhibit a strong negative correlation (-0.81). This implies that as stress levels increase, sleep duration tends to decrease. High stress is associated with shorter sleep.

The correlation matrix provides valuable insights into how various factors within your dataset are related. It highlights strong positive relationships between sleep duration and sleep quality, as well as negative relationships between stress level and sleep quality and stress level and sleep duration. These findings can be used to guide further analysis and potentially identify areas for intervention or improvement in factors affecting sleep and overall well-being.

3.3 SUPPORT VECTOR MACHINES

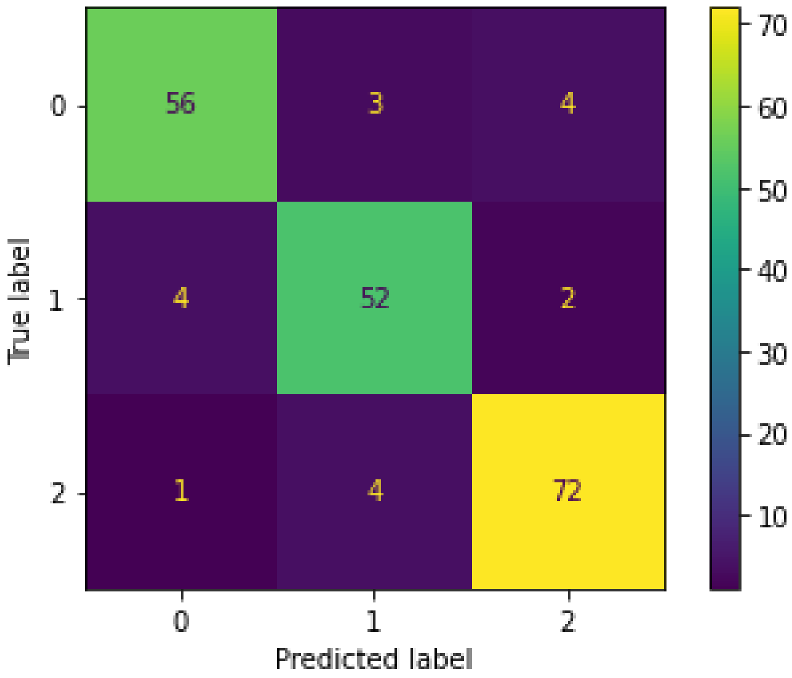
CLASSIFICATION REPORT

```
              precision    recall  f1-score   support

           0       0.92      0.89      0.90        63
           1       0.88      0.90      0.89        58
           2       0.92      0.94      0.93        77

    accuracy                           0.91       198
   macro avg       0.91      0.91      0.91       198
weighted avg       0.91      0.91      0.91       198
```

CONFUSION MATRIX



INFERENCE

The confusion matrix shows 180 correct predictions and 18 incorrect ones.

Class 0:

True Positives = 56

False Positives = 7

False Negatives = 5

Class 1:

True Positives = 52

False Positives = 6

False Negatives = 6

Class 2:

True Positives = 72

False Positives = 5

False Negatives = 5

Accuracy = 91%

3.4 K-NN ALGORITHM

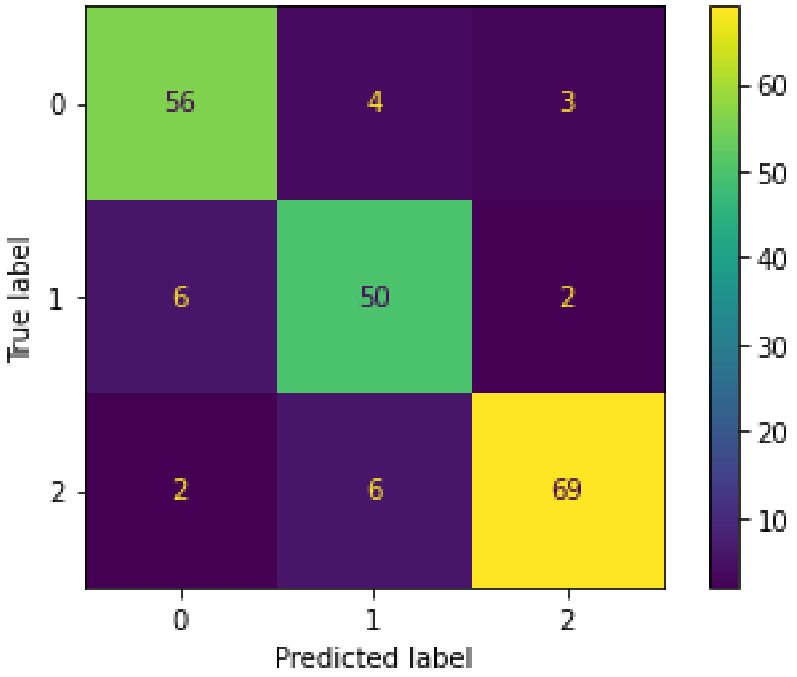
CLASSIFICATION REPORT

```
              precision    recall  f1-score   support

           0       0.88      0.89      0.88        63
           1       0.83      0.86      0.85        58
           2       0.93      0.90      0.91        77

    accuracy                           0.88       198
   macro avg       0.88      0.88      0.88       198
weighted avg       0.89      0.88      0.88       198
```

CONFUSION MATRIX



INFERENCE

The confusion matrix shows 175 correct predictions and 23 incorrect ones.

Class 0:

True Positives = 56

False Positives = 7

False Negatives = 8

Class 1:

True Positives = 50

False Positives = 8

False Negatives = 10

Class 2:

True Positives = 69

False Positives = 5

False Negatives = 8

Accuracy = 88%

3.5 DECISION TREE ALGORITHM

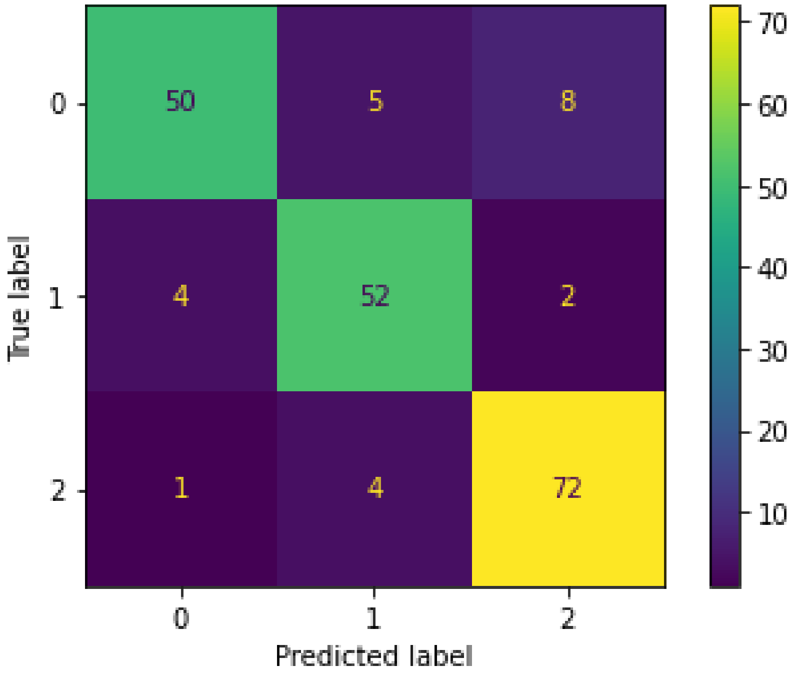
CLASSIFICATION REPORT

```
              precision    recall  f1-score   support

           0       0.91      0.79      0.85        63
           1       0.85      0.90      0.87        58
           2       0.88      0.94      0.91        77

    accuracy                           0.88       198
   macro avg       0.88      0.88      0.88       198
weighted avg       0.88      0.88      0.88       198
```

CONFUSION MATRIX



INFERENCE

The confusion matrix shows 174 correct predictions and 24 incorrect ones.

Class 0:

True Positives = 50

False Positives = 13

False Negatives =5

Class 1:

True Positives = 52

False Positives = 9

False Negatives = 6

Class 2:

True Positives = 72

False Positives = 10

False Negatives = 5

Accuracy = 88%

3.6 RANDOM FOREST

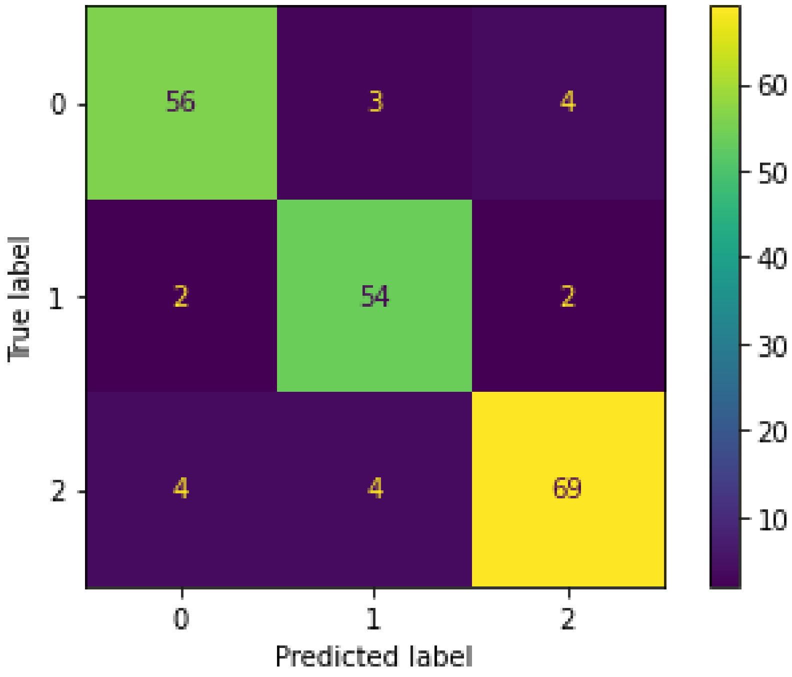
CLASSIFICATION REPORT

```
              precision    recall  f1-score   support

           0       0.90      0.89      0.90        63
           1       0.89      0.93      0.91        58
           2       0.92      0.90      0.91        77

    accuracy                           0.90       198
   macro avg       0.90      0.91      0.90       198
weighted avg       0.90      0.90      0.90       198
```

CONFUSION MATRIX



INFERENCE

The confusion matrix shows 179 correct predictions and 19 incorrect ones.

Class 0:

True Positives = 56

False Positives = 7

False Negatives =6

Class 1:

True Positives = 54

False Positives = 7

False Negatives = 4

Class 2:

True Positives = 69

False Positives = 6

False Negatives = 8

Accuracy = 90%

CHAPTER 4

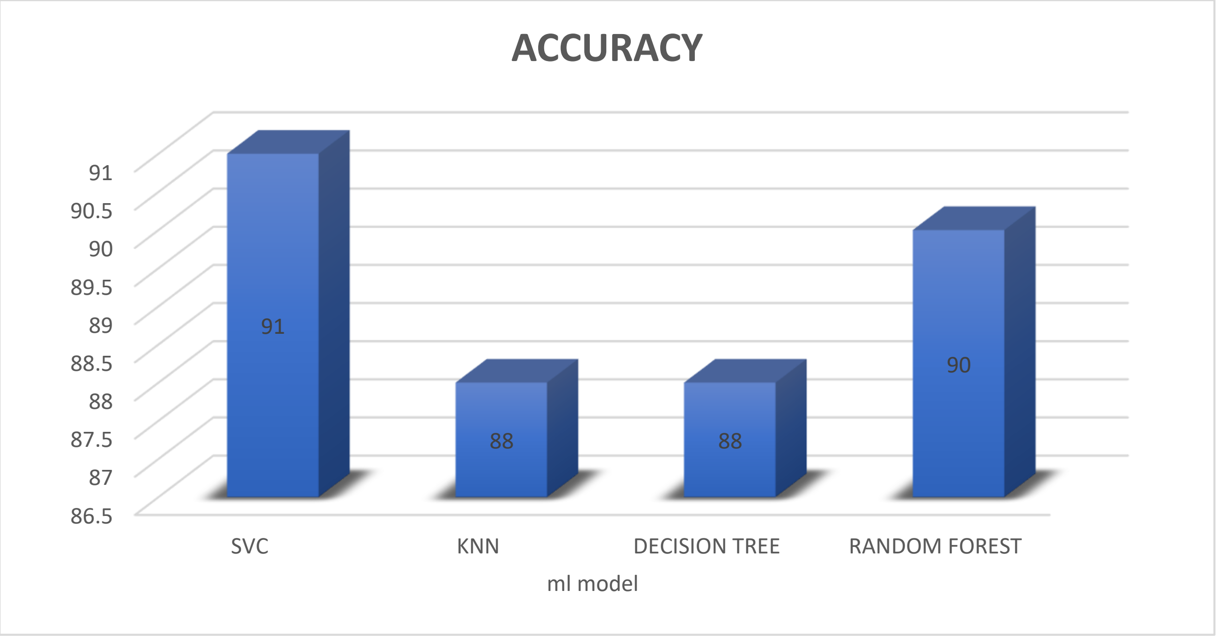
COMPARISON OF MODELS AND PREDICTION

4.1 INTRODUCTION

The optimal classification model is chosen in this chapter by comparing metrics like accuracy, precision, recall, and f1-score. Then, using this model, thyroid predictions of certain randomly chosen samples are made

4.2 COMPARISON OF ACCURACY

The comparison of accuracies obtained from various algorithms is as shown in the figure

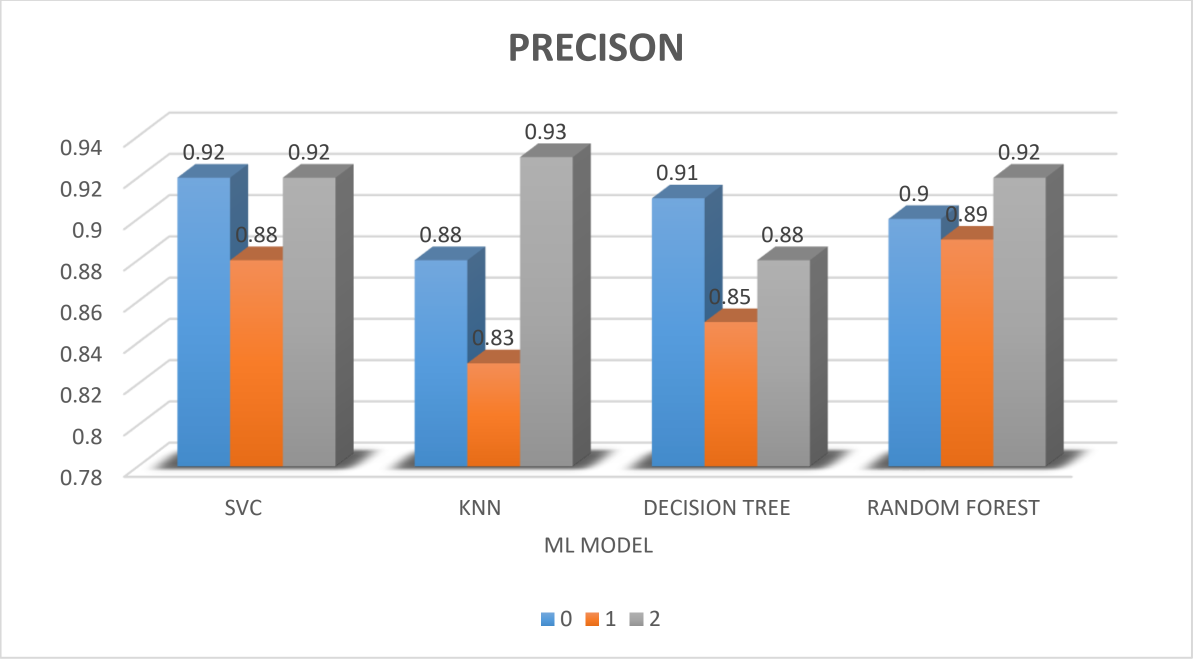


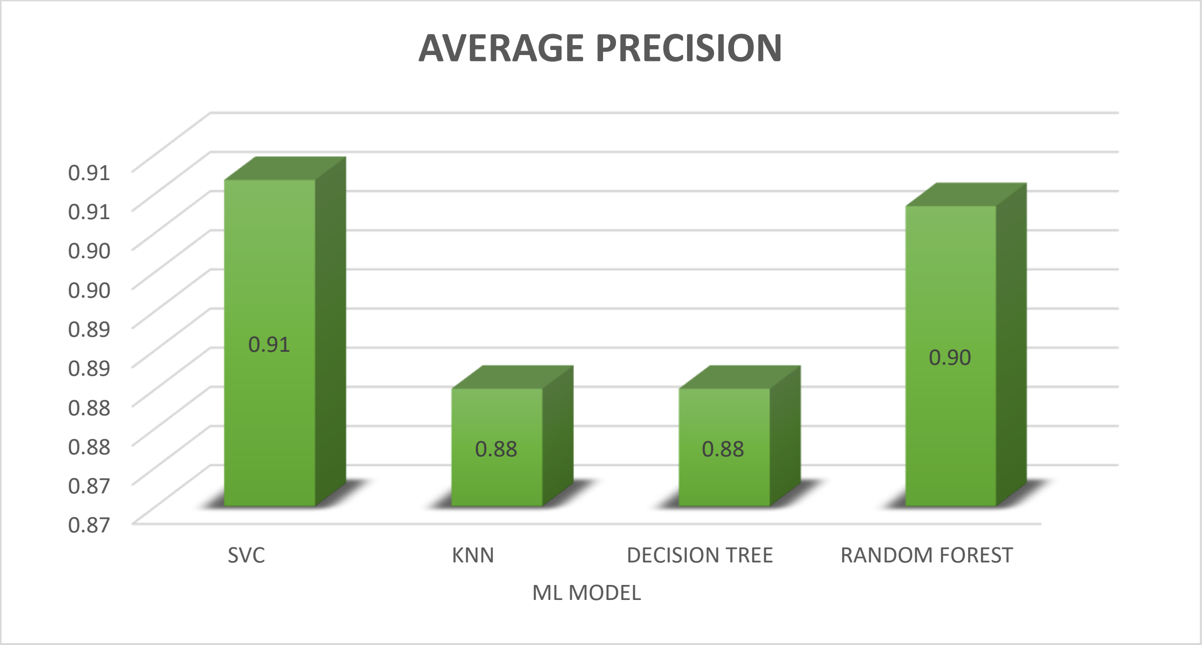
INFERENCE

• Out of all the algorithms chosen, Support Vector Classification performs best with an accuracy of 91.00 %

4.3 COMPARISON OF PRECISION

The comparison of precision obtained from various algorithms for each class is as shown in the figure.



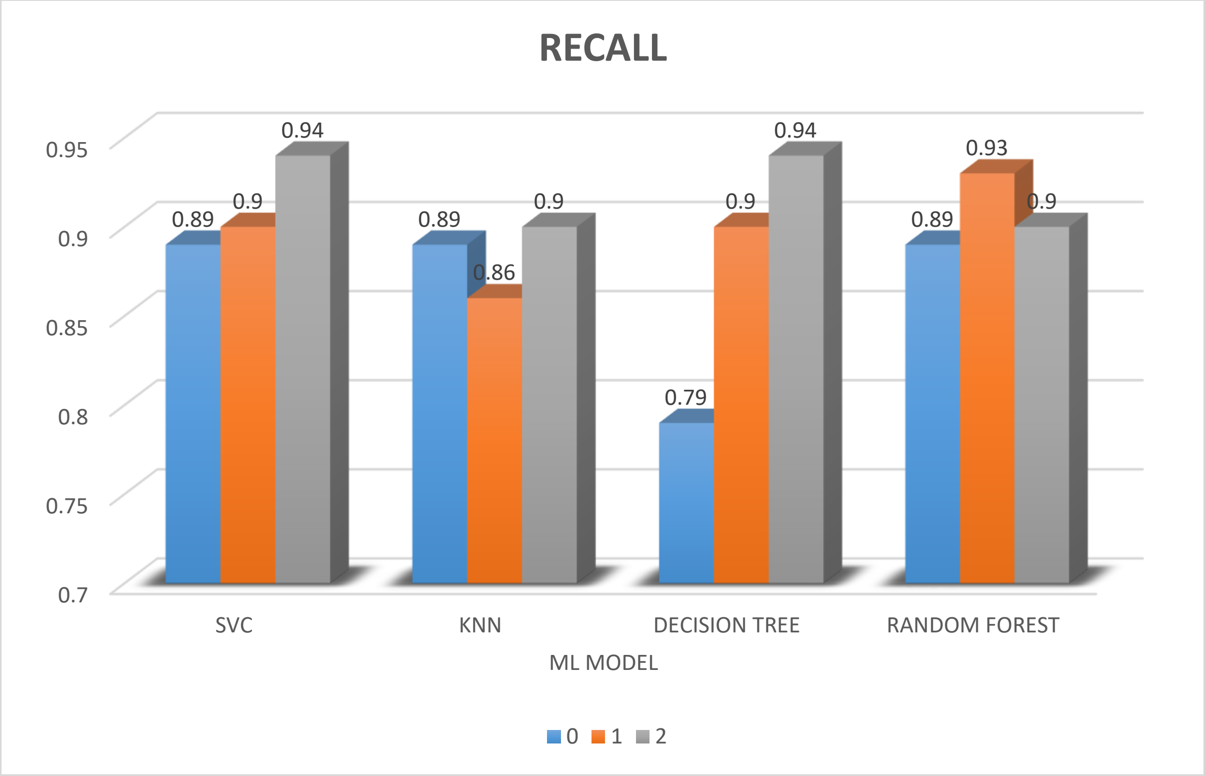


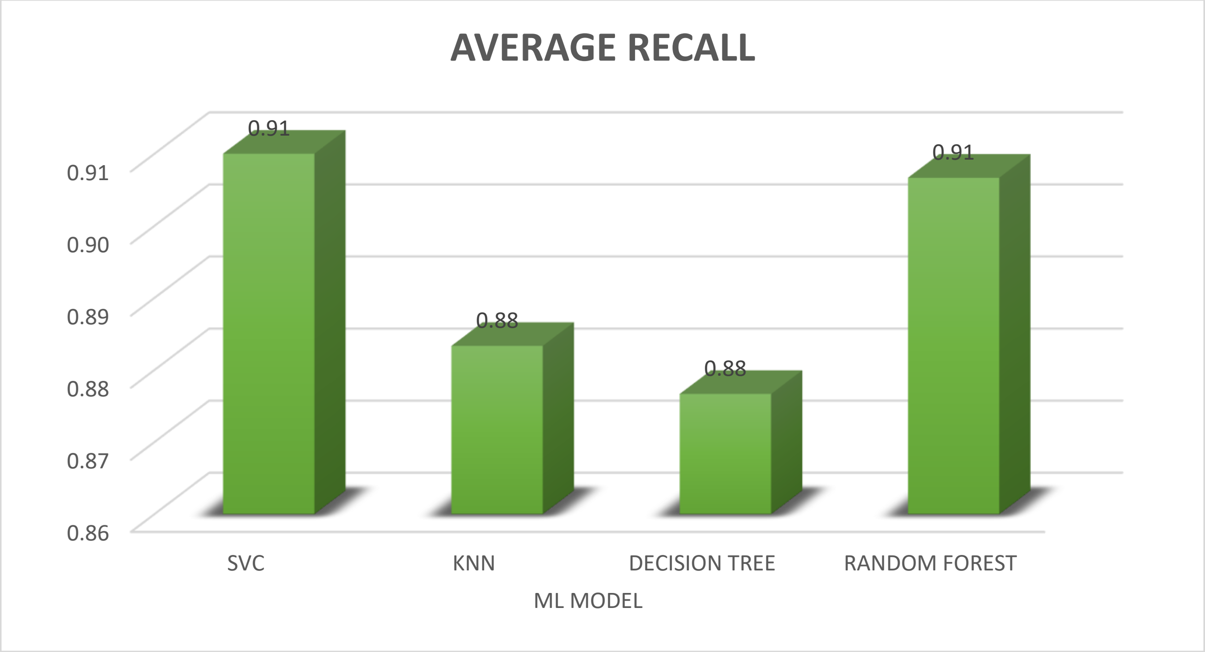
INFERENCE

•Out of all the algorithms chosen, Support Vector Classifier performs best with an average precision of 0.91

4.4 COMPARISON OF RECALL

The comparison of recall obtained from various algorithms for each class is as shown in the figure.



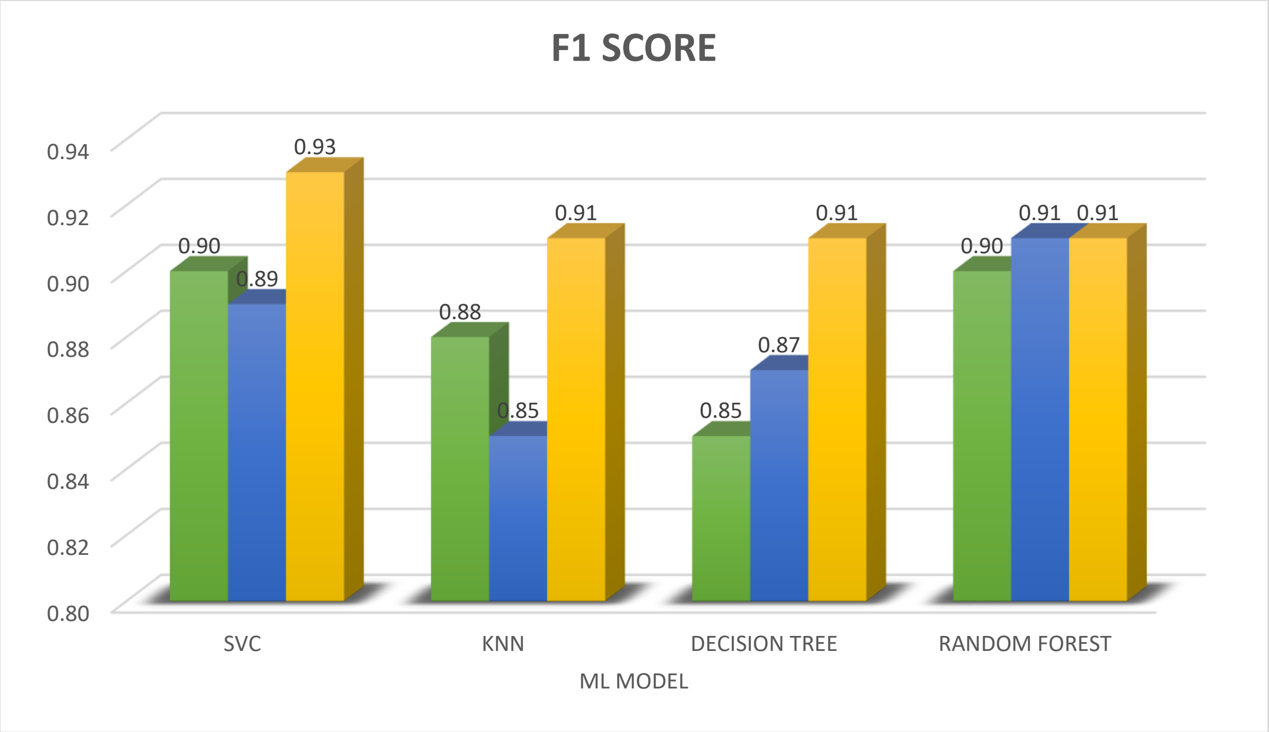


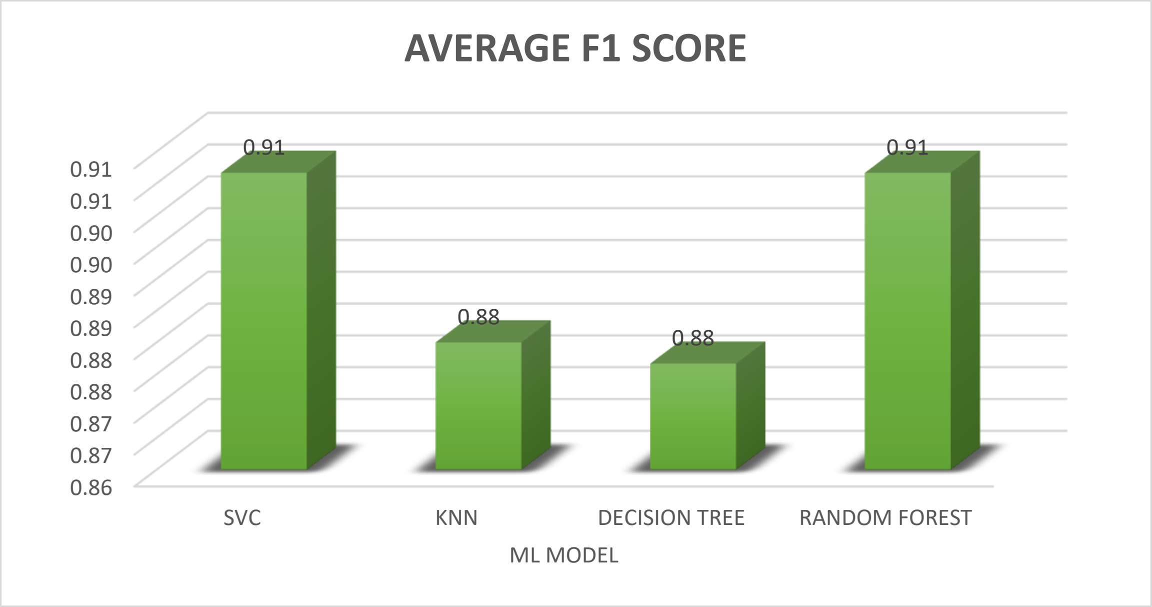
INFERENCE

• Out of all the algorithms chosen, Support Vector Classifier performs best with an average recall of 0.91

4.5 COMPARISON OF f 1-SCORE

If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the precision. The comparison of f1 score obtained from various algorithms for each class is as shown in the figure.





INFERENCE

Out of all the algorithms chosen Support Vector Classifier performs best with an average f1 score of 0.91

4.6 PREDICTION

Out of all the algorithms chosen Support Vector Classifier performs best. So predicted possibility of 12 samples using the classification Support Vector Classifier having accuracy 91.00%, average precision 0.91, average recall 0.91 and average f1 score 0.91 are given below.

```
model_loaded=pickle.load(open('C:/Users/CHROMIUM/Project/model_saved1','rb'))
import numpy as np
input_data=(1,27,9,6.1,6,42,6,3,77,4200,126,83)
input_numpy=np.asarray(input_data)
input_data_reshape=input_numpy.reshape(1,-1)
prediction=model_loaded.predict(input_data_reshape)
print(prediction)
if(prediction[0]==0):
  print('this person has insomnia')
elif(prediction[0]==1):
  print('this person is alright')
else:
  print('this person has sleep apnea')
```

OUTPUT

```
[1]
this person is alright
```

CHAPTER 5

CONCLUSIONS

Here in this study, we analyzed a dataset containing information on 374 individuals, unveiling that, on average, the subjects were approximately 42 years old, slept for around 7.13 hours per night, and reported a sleep quality rating of 7.31 out of 10. Furthermore, the data showcased intriguing relationships, such as a strong positive correlation between sleep duration and sleep quality, as well as a negative correlation between stress levels and both sleep quality and sleep duration. Our visualizations further elucidated the demographic composition, weight status distribution, and profession representation within the dataset, providing a holistic view of the study's context. These findings collectively contribute to a comprehensive understanding of sleep patterns and disorders while highlighting the potential of machine learning in healthcare applications.

we deal with the comparison of the four models using the classification algorithms in machine learning which has been used in analysing sleep disorder prediction. Here we have used

1. Support Vector Classifier algorithm
2. K- Nearest Neighbour algorithm
3. Random Forest algorithm
4. Decision Tree algorithm

We are comparing the performance of these models by accuracy, precision, recall and f1 ratio which has already been discussed in chapters. The accuracy, precision, recall and f1 ratio for the models have been tabled below:

Model	Accuracy	Precision	Recall	f1-score
SVM	0.91	0.91	0.91	0.91
K NEAREST NEIGHBOUR	0.88	0.88	0.88	0.88
DECISION TREE	0.88	0.88	0.88	0.88
RANDOM FOREST	0.90	0.90	0.91	0.91

The model using Support Vector Classifier algorithm has been selected out of all other model because it performs best with an accuracy 91%, average precision 0.91, average recall 0.91 and average f1 ratio 0.91

BIBLIOGRAPHY

1. Andreas C Muller and Sarah Gudio, 2016, Introduction to Machine Learning with Python, O‟Reilly Media,Inc
2. Shai Shalev-Shwartz and Shai Ben-David,2014, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press
3. Ethem Alpaydin,2014 , Introduction to Machine Learning, Third edition, Cambridge,Massacgusetts Lundon,England,The MIT Press.
4. Norman Matloff, Stastistical Regression and Classification From Linear Models to Machine Learning,Boca Raton London New York , CRC Press.
5. Peter Bruce and Andrew Bruce,2017, Practica Statistics for Data Scientists 50 Essential Concepts, O‟Reilly Media, Inc.
6. Gangavarapu Sailasya , Gorli L Aruna Kumari, 2021, Analyzing the Performance of Stroke Prediction using ML Classification Algorithms, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 6
7. Data - https://www.kaggle.com/datasets/ uom190346a/sleep-health-and-lifestyle-dataset
8. https://www.psychiatry.org/patients-families/sleep-disorders

APPENDIX

```
import pandas as pd
df =
pd.read_csv('C:/Users/CHROMIUM/Downloads/Sleep_health_and_lifestyle_dataset
.csv')
df.head()
df.tail()
df.isnull().sum()
df.describe()
df.info()
df[['SYSTOLIC','DIASTOLIC']]=df['Blood
Pressure'].str.split('/',expand=True)
df.drop(['Blood Pressure'],axis=1,inplace=True)
df['BMI Category'].value_counts()
df['Gender'].value_counts()
df['Quality of Sleep'].value_counts()
df['Stress Level'].value_counts()
df['BMI Category'] = df['BMI Category'].replace('Normal', 'Under Weight')
df1=df.copy()
df1['Sleep Disorder']=df['Sleep Disorder']
df1.head()
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(8, 6))
sns.countplot(x='Gender', data=df1, hue='Sleep Disorder')
plt.title('Gender vs. Sleep Disorder')
plt.show()
plt.figure(figsize=(12, 8))  # Set the figure size to 12x8 inches

sns.barplot(x='Quality of Sleep', y='Stress Level', data=df1)
plt.title('QUALITY OF SLEEP VS Stress Level')
plt.ylabel('Stress Level')
plt.xlabel('Quality of Sleep')
plt.show()

# Create a pie chart for BMI Category distribution
plt.figure(figsize=(12, 12))
df1['BMI Category'].value_counts().plot(kind='pie', autopct='%1.1f%%')
plt.title('BMI Category Distribution')
plt.ylabel('')
plt.show()
df1.corr()
# Calculate the correlation matrix
correlation_matrix = df1.corr()

# Create a heatmap for the correlation matrix
plt.figure(figsize=(15, 12))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix Heatmap')
plt.show()
# Create a histogram for Sleep Duration
plt.figure(figsize=(8, 6))
sns.histplot(data=df1, x='Sleep Duration')
plt.title('Histogram: Sleep Duration')
plt.show()

plt.figure(figsize=(8, 6))
sns.histplot(data=df1, color='blue', x='Age')
plt.title('Histogram: Age')
plt.show()
plt.figure(figsize=(8, 6))
sns.histplot(data=df1, color='green', x='Physical Activity Level')
plt.title('Histogram:  Physical Activity Level')
plt.show()
plt.figure(figsize=(8, 6))
sns.histplot(data=df1, color='black', x='Heart Rate')
plt.title('Histogram:  Heart Rate')
plt.show()
df1.columns
# Create a box plot for BMI Category vs. Stress Level
plt.figure(figsize=(8, 6))
sns.boxplot(data=df1, x='BMI Category', y='Stress Level', hue='Sleep
Disorder')
plt.title('Box Plot: BMI Category vs. Stress Level')
plt.show()
plt.figure(figsize=(12, 8))  # Set the figure size to 12x8 inches

sns.barplot(x='Sleep Duration', y='Quality of Sleep', data=df1)
plt.title('SLEEP DURATION VS QUALITY OF SLEEP')
plt.xlabel('Sleep Duration')
plt.ylabel('Quality of Sleep')
plt.show()
                                          #CONVERTING TO FLOAT
df['SYSTOLIC']=df['SYSTOLIC'].astype(float)
df['DIASTOLIC']=df['DIASTOLIC'].astype(float)
df1['SYSTOLIC']=df1['SYSTOLIC'].astype(float)
df1['DIASTOLIC']=df1['DIASTOLIC'].astype(float)
                                        #CHECK FOR MISSING VALUES AND CHECK
TYPES

df.isna().sum()
df.dtypes
df['Sleep Disorder'].value_counts()
import warnings
warnings.filterwarnings('ignore')
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
df1['Occupation']=le.fit_transform(df1['Occupation'])
df1['BMI Category']=le.fit_transform(df1['BMI Category'])
df['Sleep Disorder']=le.fit_transform(df['Sleep Disorder'])
df1['Gender']=le.fit_transform(df1['Gender'])
df1
df1['Sleep Disorder']=df['Sleep Disorder']
df1
df1['Sleep Disorder'].value_counts()
                                   #SEPERATE X AND Y

X=df1.iloc[:,1:-1]
y=df1.iloc[:,-1]
x
y
```

```
                                  #OVERSAMPLING THE DATA
from imblearn.over_sampling import SMOTE
os=SMOTE(random_state=1)
X,y=os.fit_resample(X,y) sns.barplot(x='Sleep Duration',y='Quality of
Sleep',data=df1).set(title='SLEEP DURATION VS QUALITY OF SLEEP')
sns.barplot(x='Stress Level',y='Quality of
Sleep',data=df1).set(title='Stress Level VS QUALITY OF SLEEP')
                                        #SCALING

from sklearn.preprocessing import MinMaxScaler
mm=MinMaxScaler()
X=mm.fit_transform(X)
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_sta
te=1)
                                  #ALGORITHM 1 : K-NEIGHBORS

from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report,ConfusionMatrixDisplay
knn=KNeighborsClassifier(n_neighbors=3)
knn.fit(X_train,y_train)
y_pred=knn.predict(X_test)
print(classification_report(y_test,y_pred))
print(ConfusionMatrixDisplay.from_predictions(y_test,y_pred))
                               #2 ALGORITHM 2: SVC
                           #BEFORE HYPERPARAMETER TUNING

from sklearn.svm import SVC
sv=SVC()
sv.fit(X_train,y_train)
y_pred1=sv.predict(X_test)
print(classification_report(y_test,y_pred1))
from sklearn.model_selection import GridSearchCV
sv=SVC()
params={'C':[0.1,1,10,100],'gamma':[1,0.1,0.01,0.001],'kernel':['rbf','poly
','sigmoid']}
clf=GridSearchCV(sv,params,cv=5,scoring='accuracy')
clf.fit(X_train,y_train)
clf.best_params_
                         #AFTER HYPERPARAMETER TUNING

sv=SVC(C=100,kernel='poly',gamma=1)
sv.fit(X_train,y_train)
y_pred1=sv.predict(X_test)
print(classification_report(y_test,y_pred1))
print(ConfusionMatrixDisplay.from_predictions(y_test,y_pred1))

                            #ALGORITHM 3 : RANDOM FOREST
                            # BEFORE HYPERPARAMETER TUNING

from math import sqrt
from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier()
rf=RandomForestClassifier()
params1={'n_estimators':[25,50,100,150],'max_features':['sqrt','auto'],'max
_depth':[2,4],'min_samples_split':[2,5],'min_samples_leaf':[1,2]}
clf=GridSearchCV(rf,params1,cv=5,scoring='accuracy')
```

```
clf.fit(X_train,y_train)
clf.best_params_
                                #AFTER HYPERPARAMETER TUNING

rf=RandomForestClassifier(max_depth= 10,
 max_features= 'sqrt',
 min_samples_leaf= 1,
 min_samples_split= 2,
 n_estimators= 150)
rf.fit(X_train,y_train)
y_pred3=rf.predict(X_test)
print(classification_report(y_test,y_pred3))

print(ConfusionMatrixDisplay.from_predictions(y_test,y_pred3))

                                  # ALGORITH 4: DECISION TREE

from sklearn.tree import DecisionTreeClassifier
clf=DecisionTreeClassifier()
clf.fit(X_train,y_train)
y_pred4=clf.predict(X_test)
print(classification_report(y_test,y_pred4))


print(ConfusionMatrixDisplay.from_predictions(y_test,y_pred4))

import pickle
pickle.dump(sv,open('C:/Users/CHROMIUM/Downloads/model_saved1','wb'))

model_loaded=pickle.load(open('C:/Users/CHROMIUM/Downloads/model_saved1','r
b'))
import numpy as np
input_data=(1,27,9,6.1,6,42,6,3,77,4200,126,83)
input_numpy=np.asarray(input_data)
input_data_reshape=input_numpy.reshape(1,-1)
prediction=model_loaded.predict(input_data_reshape)
print(prediction)
if(prediction[0]==0):
  print('this person has insomnia')
elif(prediction[0]==1):
  print('this person is alright')
else:
  print('this person has sleep apnea')
```