



MAR ATHANASIUS COLLEGE OF ENGINEERING, KOTHAMANGALAM

(Affiliated to APJ Abdul Kalam Technological University, Thiruvananthapuram)

Initial Project Report on

OBESITY DETECTION USING MACHINE LEARNING

In partial fulfillment of the requirement for the award of the degree in

MASTER OF COMPUTER APPLICATIONS

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY



Submitted by

MERYL JOHN
Reg No: MAC23MCA-2038

Under the guidance of

Prof. NISHA MARKOSE

ABSTRACT

Obesity is a medical condition characterized by an excessive accumulation of body fat, which can have negative effects on health. It is typically measured using the body mass index (BMI), where a BMI of 30 or above is classified as obese. Obesity increases the risk of various health problems, including heart disease, diabetes, high blood pressure, certain cancers, and joint issues. The “Obesity Detection Using Machine Learning” project is focused on detecting and diagnosing obesity.

From the three papers, we get to know that different approaches are used for the detection of obesity. The main theme of the first paper is the application of machine learning algorithms to predict obesity risk. The second paper deals with the application of machine learning techniques to predict overweight and obesity. Third paper aims at the prediction of obesity levels using a trained neural network approach optimized by Bayesian techniques.

The system is the comparative study of Logistic Regression and Random Forest. The models will classify obesity under seven classes which are Underweight, Normal Weight, Overweight Level I, Overweight Level II, Obesity Type I, Obesity Type II and Obesity Type III. By comparing these algorithms, the project aims to determine which model offers the highest accuracy and reliability in detecting obesity.

The dataset is taken from Kaggle repository. The dataset contains 2111 sample observations and has 17 columns including 1 identifier, 1 class variable and 16 features. The dataset contains Numeric and Categorical values.

Dataset: <https://www.kaggle.com/code/mabdullahabrar/multi-class-obesity-risk-prediction>

References

- Ferdowsy, Faria, Kazi Samsul Alam Rahi, Md Ismail Jabiullah, and Md Tarek Habib. "A machine learning approach for obesity risk prediction." *Current Research in Behavioral Sciences* 2 (2021): 100053.
- Rodríguez, Elias, Elen Rodríguez, Luiz Nascimento, Aneirson Francisco da Silva, and Fernando Augusto Silva Marins. "Machine learning Techniques to Predict Overweight or Obesity." In *IDDM*, pp. 190-204. 2021.
- Yagin, F. H., Güllü, M., Gormez, Y., Castañeda-Babarro, A., Colak, C., Greco, G., ... & Cataldi, S. (2023). Estimation of obesity levels with a trained neural network approach optimized by the Bayesian technique. *Applied Sciences*, 13(6), 3875

Faculty Guide

Prof. Nisha Markose
Associate Professor

Submitted By:

Meryl John
Reg No.: MAC23MCA-2038

Project Coordinator

Prof. Sonia Abraham
Associate Professor

INTRODUCTION

Obesity, a complex and multifaceted health condition, poses significant challenges to public health worldwide. Characterized by excessive body fat accumulation, obesity increases the risk of various chronic diseases such as diabetes, cardiovascular conditions, and certain cancers. Traditional methods of detecting and predicting obesity often rely on body mass index (BMI) calculations and clinical assessments, which, while useful, can be limited by their static nature and reliance on physical examinations. Machine learning, a subset of artificial intelligence, offers promising advancements in the field of obesity detection and prediction. By leveraging vast amounts of data and sophisticated algorithms, machine learning models can identify patterns and correlations that might not be evident through conventional methods.

The “Obesity Detection Using Machine Learning” project is focused on detecting obesity. The performance of two machine learning algorithms such as Logistic Regression and Random Forest are compared to classify obesity into Underweight, Normal Weight, Overweight Level I, Overweight Level II, Obesity Type I, Obesity Type II and Obesity Type III.

The dataset is taken from the Kaggle repository. The dataset contains 2111 sample observations and has 17 columns including 1 identifier, 1 class variable and 16 features. The dataset contains Numeric and Categorical values.

To address these challenges, an automated system using machine learning can assist medical professionals by providing more accurate predictions of obesity risk. This system leverages vast datasets containing clinical measurements, patient history, and lifestyle factors to train machine learning models. These models can identify patterns and correlations that might be missed by human experts.

LITERATURE REVIEW

Paper 1: Ferdowsy, Faria, Kazi Samsul Alam Rahi, Md Ismail Jabiullah, and Md Tarek Habib."A machine learning approach for obesity risk prediction." *Current Research in Behavioral Sciences* 2 (2021): 100053.

This paper focuses on predicting obesity risk using machine learning techniques. The authors collected data from 1,100 participants in Bangladesh and evaluated various algorithms, including logistic regression, random forest, and gradient boosting, achieving a prediction accuracy of 97.09%.

The study emphasizes the importance of feature selection and data preprocessing to enhance prediction accuracy. The results demonstrate that machine learning can effectively predict obesity risk, providing valuable insights for healthcare professionals to develop targeted interventions.

Title of the paper	Ferdowsy, Faria, Kazi Samsul Alam Rahi, Md Ismail Jabiullah, and Md Tarek Habib."A machine learning approach for obesity risk prediction." <i>Current Research in Behavioral Sciences</i> 2 (2021): 100053.	
Area of work	Prediction of Obesity	
Dataset	Collected 1100 data based on 28 factors and then labelled the class of each record of the data set by consulting with some nutritionists and student counsellors in educational institutions.	
Methodology / Strategy	k-NN, Support Vector Machine, Logistic Regression, Naïve Bias, Random Forest, Decision Tree, ADA Boosting, MLP, Gradient Boosting are compared on the basis of Accuracy, Sensitivity, Specificity, Precision Recall and F1-score.	
Algorithm	k-NN, SVM, LR, Naïve Bias, RF, Decision Tree, ADA Boosting, MLP, Gradient Boosting.	
Result/Accuracy	LR - 97.09% RF - 72.30% MLP - 66.02% k-NN - 77.50% SVM - 66.02%	Decision tree - 70.30% ADA boosting - 70.03% Naive Bayes - 86.04% Gradient boosting - 64.08%

Paper 2: Rodríguez, Elias, Elen Rodríguez, Luiz Nascimento, Aneirson Francisco da Silva, and Fernando Augusto Silva Marins. "Machine learning Techniques to Predict Overweight or Obesity." In IDDM, pp. 190-204. 2021.

This study explores the application of eight different machine learning models—decision tree, support vector machines, k-nearest neighbors, Gaussian naive Bayes, multilayer perceptron, random forest, gradient boosting, and extreme gradient boosting to identify individuals with obesity or overweight.

The data for model training was collected through surveys, focusing on eating habits and physical activity. The random forest model showed the highest performance with an accuracy of 77.69%, precision of 78.53%, recall of 78.15%, and F1-score of 78.09%. The paper concludes that machine learning models can significantly aid in the early identification of obesity.

Title of the paper	Rodríguez, Elias, Elen Rodríguez, Luiz Nascimento, Aneirson Francisco da Silva, and Fernando Augusto Silva Marins. "Machine learning Techniques to Predict Overweight or Obesity." In IDDM, pp. 190-204. 2021.	
Area of work	Prediction of Overweight or Obesity	
Dataset	The investigation included data for the estimation of obesity levels, including the eating habits and physical activity statuses of 498 participants between the ages of 14 and 61 from Barranquilla, Colombia; Lima, Peru; and the City of Mexico, Mexico.	
Methodology / Strategy	Decision Tree, Support Vector Machines, K Nearest Neighbors, Gaussian Naive Bias, Multilayer Perceptron, Random Forest, Gradient Boosting, Extreme Gradient Boosting are compared on the basis of Accuracy, Precision, Recall and F1-score.	
Algorithm	Decision Tree, SVM, KNN, Gaussian Naive Bias, MLP, Random Forest, Gradient Boosting, Extreme Gradient Boosting.	
Result/Accuracy	RF - 77.69% GB - 73.43% Decision Tree - 72.62% Extreme GB - 70.06% Gaussian Naive Bayes - 46.24% K-NN - 67.69% MLP - 63.77% SVM - 59.45%	

Paper 3: Yagin, F. H., Güllü, M., Gormez, Y., Castañeda-Babarro, A., Colak, C., Greco, G., ... & Cataldi, S. (2023). Estimation of obesity levels with a trained neural network approach optimized by the Bayesian technique. *Applied Sciences*, 13(6), 3875.

This paper examines the effectiveness of various machine learning algorithms in predicting obesity by analyzing a dataset containing information on eating habits, physical activity, and other relevant factors.

The study compares multiple models, including decision trees, support vector machines, and random forests, and concludes that machine learning provides a robust approach for obesity prediction.

The paper highlights that the integration of machine learning tools in the medical field can enhance the accuracy of obesity prediction, thus helping in the timely intervention and management of obesity-related health issues.

Title of the paper	Yagin, F. H., Güllü, M., Gormez, Y., Castañeda-Babarro, A., Colak, C., Greco, G., ... & Cataldi, S. (2023). Estimation of obesity levels with a trained neural network approach optimized by the Bayesian technique. <i>Applied Sciences</i> , 13(6), 3875.	
Area of work	Obesity Prediction and Classification Model	
Dataset	The dataset was taken from the UCI Machine Learning Repository. It consists of 2111 records with 17 features related to eating habits and physical conditions collected from Mexico, Peru, and Colombia.	
Methodology / Strategy	Chi-Square, F Classify, Mutual Information Classification are compared on the basis of Accuracy, SD Accuracy, F1-Score, Sensitivity, Specificity.	
Algorithm	Neural Network Model and Bayesian Classification	
Result/Accuracy	Original Model	93.06%
	Chi-Square	89.04%
	F-Classify	90.32%
	Mutual Information Classification	86.52%

SUMMARY

The first paper discusses the application of machine learning techniques to predict obesity risk based on various factors. The study details the system architecture, research methodology, and experimental evaluation used in the process. Features were carefully selected and the data was pre-processed to improve the model's accuracy. Comparative analysis of different machine learning models was conducted, showing that the approach could effectively predict obesity risk, aiding healthcare professionals in making better decisions regarding obesity management.

The second paper aims to develop a predictive model using machine learning to identify individuals at risk of being overweight or obese. The dataset includes physical condition and eating habits data. Various machine learning algorithms were tested, including decision trees, SVM, k-nearest neighbours, and random forests. The best performing model, a random forest, achieved 78% accuracy. The study highlights the potential of machine learning models in public health to identify and manage obesity and overweight issues effectively.

The third paper focuses on predicting obesity levels using a neural network optimized with Bayesian techniques. The study identifies critical factors associated with obesity, such as physical activity and eating habits, using chi-square, F-Classify, and mutual information classification algorithms. The model's performance showed high accuracy, with the neural network achieving 93.06% accuracy when all features were used. The study concludes that physical activity and eating habits are crucial for predicting obesity levels, with Bayesian optimization enhancing the prediction accuracy.

PROJECT PROPOSAL

From the above three papers, we get to know that different approaches are used for the detection of obesity. The main theme of the first paper is the application of machine learning algorithms to predict obesity risk. The second paper deals with the application of machine learning techniques to predict overweight and obesity. Third paper aims at the prediction of obesity levels using a trained neural network approach optimized by Bayesian techniques.

Accurate and timely detection of obesity is crucial for effective treatment and management. The proposed system is the comparative study of two algorithms Logistic Regression and Random Forest. These algorithms are more accurate than the other algorithms with approximate accuracy of 97% and 78% respectively. The models will classify obesity under seven classes which are Underweight, Normal Weight, Overweight Level I, Overweight Level II, Obesity Type I, Obesity Type II and Obesity Type III. Random Forest is known for its robustness and handling of complex datasets and Logistic Regression for its simplicity and interpretability.

Dataset is collected from Kaggle. The dataset is then pre-processed to normalize data, and select relevant features. The dataset is split into training and testing dataset. Two different machine learning models such as Logistic Regression (LR) and Random Forest (RF), are trained using the prepared training dataset. The models are evaluated on a separate test dataset using metrics such as accuracy, precision, recall, F1-score etc.

An automated system based on these machine learning models can significantly benefit both medical professionals and patients. Incorrect diagnosis may lead to wrong medication and further complexities. So, an automated system can be very helpful to assist medical experts and even make automated disease predictions without any human mistakes. Patients can also diagnose their condition without the assistance of a medical expert.

DATASET

The dataset is taken from the Kaggle repository. The dataset contains 2111 sample observations and has 17 columns including 1 identifier, 1 class variable and 16 features. The dataset contains Numeric, Boolean and Categorical values.

The identifier is the id. The features are gender, age, height, weight, family_history_with_overweight. The features related to eating habits were: frequent consumption of high-caloric food (FAVC), frequency of consumption of vegetables (FCVC), number of main meals (NCP), consumption of food between meals (CAEC), smoking habit (SMOKE), consumption of water daily (CH20), and consumption of alcohol (CALC). The features related to the physical condition were: calorie consumption monitoring (SCC), physical activity frequency (FAF), time using technological devices (TUE), and transportation used (MTRANS), Obesity Level (NOBeyesdad). The class labels include Underweight, Normal Weight, Overweight Level I, Overweight Level II, Obesity Type I, Obesity Type II, Obesity Type III.

Dataset : <https://www.kaggle.com/code/mabdullahabrar/multi-class-obesity-risk-prediction>

	Gender	Age	Height	Weight	Family History with Overweight	Frequent consumption of high caloric food	Frequency of consumption of vegetables	Number of main meals	Consumption of food between meals	Smoke	Consumption of water daily	Calories consumption monitoring
0	Female	21.0	162.0	64.0	yes	no	Sometimes	3	Sometimes	no	Between 1 and 2 L	no
1	Female	21.0	152.0	56.0	yes	no	Always	3	Sometimes	yes	More than 2 L	yes
2	Male	23.0	180.0	77.0	yes	no	Sometimes	3	Sometimes	no	Between 1 and 2 L	no
3	Male	27.0	180.0	87.0	no	no	Always	3	Sometimes	no	Between 1 and 2 L	no
4	Male	22.0	178.0	89.8	no	no	Sometimes	1	Sometimes	no	Between 1 and 2 L	no

Physical activity frequency	Time using technology devices	Consumption of alcohol	Transportation used	Obesity
I do not have	3–5 hours	no	Public Transportation	Normal Weight
4 or 5 days	0–2 hours	Sometimes	Public Transportation	Normal Weight
2 or 4 days	3–5 hours	Frequently	Public Transportation	Normal Weight
2 or 4 days	0–2 hours	Frequently	Walking	Overweight Level I
I do not have	0–2 hours	Sometimes	Public Transportation	Overweight Level II

EXPLORATORY ANALYSIS



Dataset has 2111 rows and 17 columns

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2111 entries, 0 to 2110
Data columns (total 17 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Gender                                     2111 non-null   object
1   Age                                        2111 non-null   float64
2   Height                                    2111 non-null   float64
3   Weight                                    2111 non-null   float64
4   family_history_with_overweight           2111 non-null   object
5   FAVC                                      2111 non-null   object
6   FCVC                                      2111 non-null   float64
7   NCP                                       2111 non-null   float64
8   CAEC                                      2111 non-null   object
9   SMOKE                                     2111 non-null   object
10  CH2O                                      2111 non-null   float64
11  SCC                                       2111 non-null   object
12  FAF                                       2111 non-null   float64
13  TUE                                       2111 non-null   float64
14  CALC                                      2111 non-null   object
15  MTRANS                                    2111 non-null   object
16  NObeyesdad                               2111 non-null   object
dtypes: float64(8), object(9)
memory usage: 280.5+ KB
```

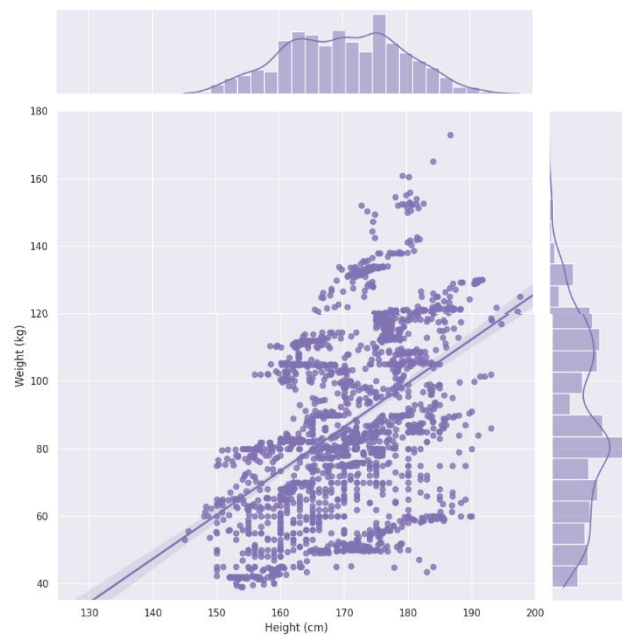
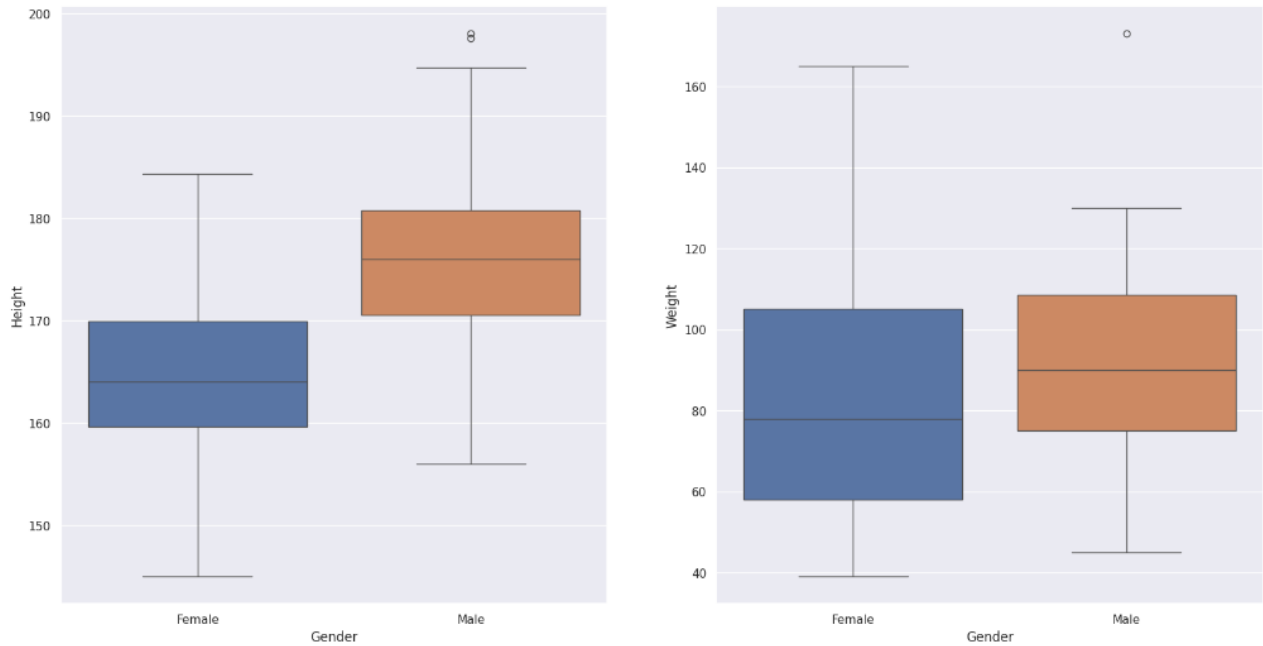
Dataset contains 8 numerical attributes and 9 categorical attributes including target class.

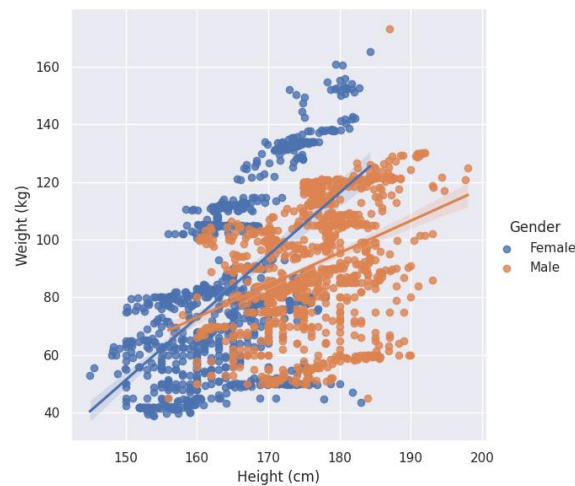
	Age	Height	Weight	FCVC	NCP	CH2O	FAF	TUE
count	2111.000000	2111.000000	2111.000000	2111.000000	2111.000000	2111.000000	2111.000000	2111.000000
mean	24.312600	1.701677	86.586058	2.419043	2.685628	2.008011	1.010298	0.657866
std	6.345968	0.093305	26.191172	0.533927	0.778039	0.612953	0.850592	0.608927
min	14.000000	1.450000	39.000000	1.000000	1.000000	1.000000	0.000000	0.000000
25%	19.947192	1.630000	65.473343	2.000000	2.658738	1.584812	0.124505	0.000000
50%	22.777890	1.700499	83.000000	2.385502	3.000000	2.000000	1.000000	0.625350
75%	26.000000	1.768464	107.430682	3.000000	3.000000	2.477420	1.666678	1.000000
max	61.000000	1.980000	173.000000	3.000000	4.000000	3.000000	3.000000	2.000000

The count, mean, standard deviation, minimum value, maximum value of the dataset attributes.

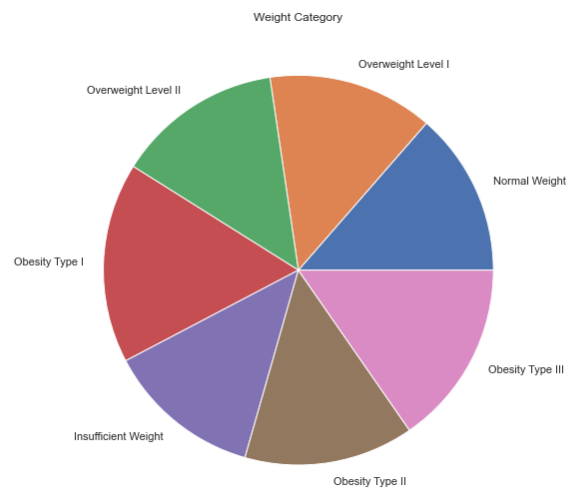
Age, Height and Weight

In terms of height, male and female are similarly distributed according to the box plot below. While males are generally taller than female, both male and female share a similar average. In weight, females has a much larger range of (as well as BMI) compared to male. This is further illustrated by the steeper line plot between weight and height of female than male.

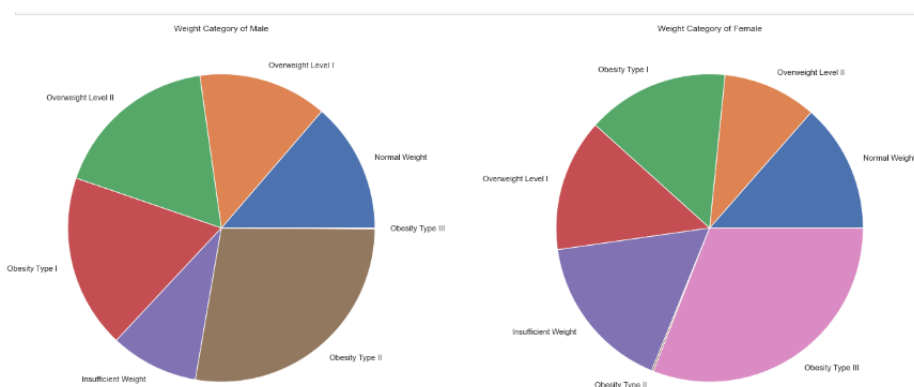




Obesity

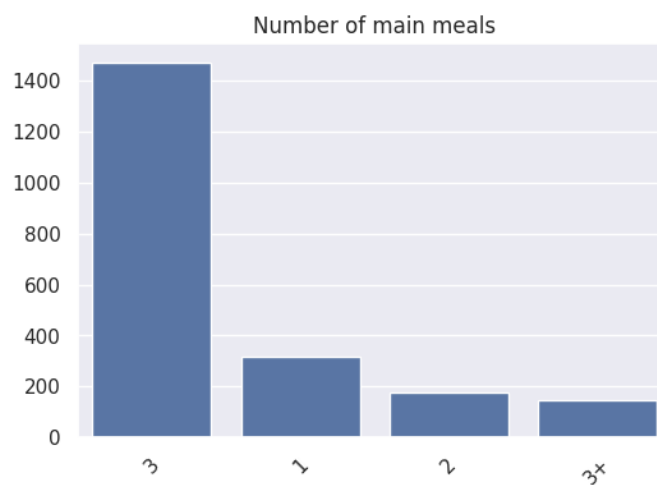
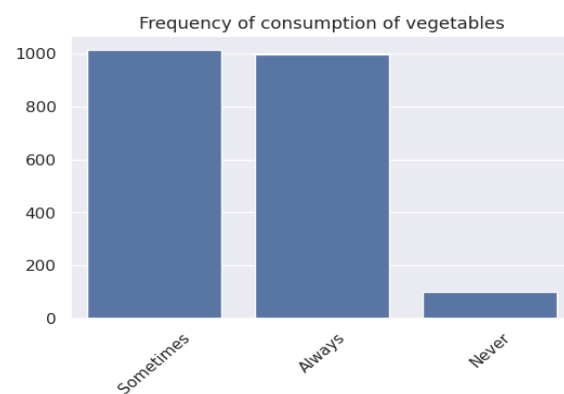
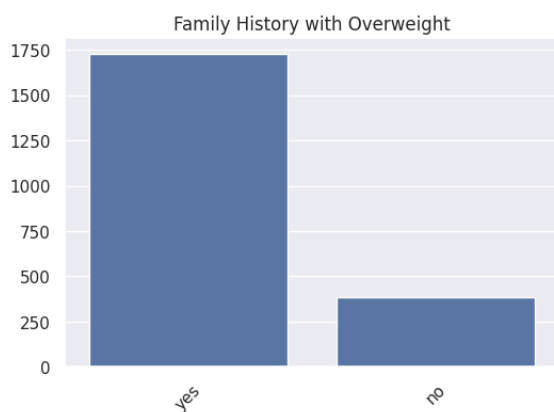


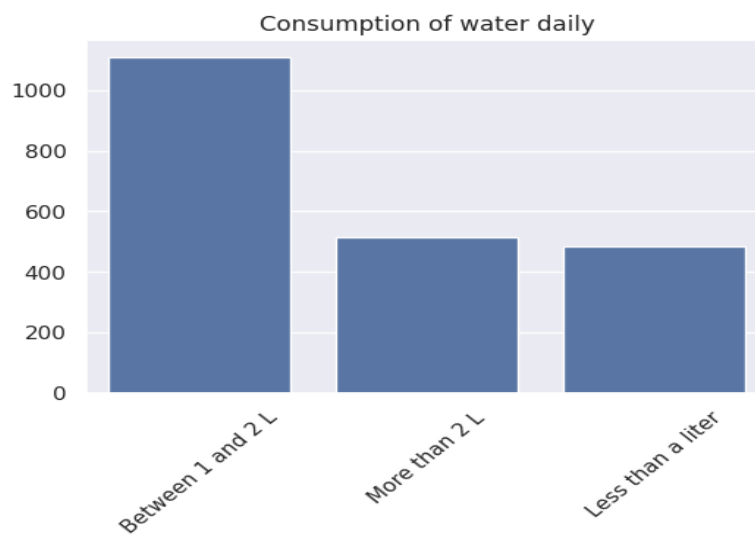
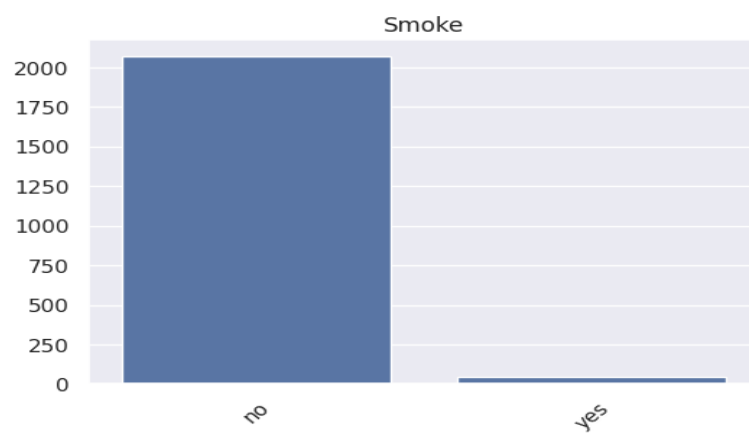
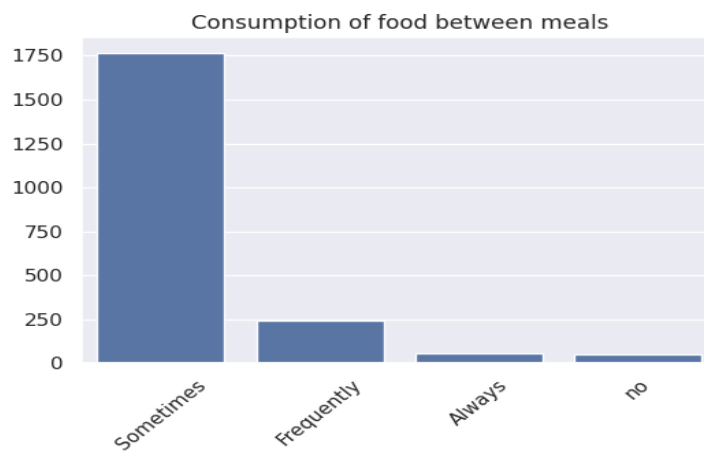
A bigger proportion of female with a higher BMI is reflected by the large slice of Obesity Type III in the pie chart below, while Obesity Type II is the most prevalent type of obesity in male. Interestingly, there is also a higher proportion of Insufficient Weight in female compared to male, this could be explained by a heavier societal pressure on women to go on diets.

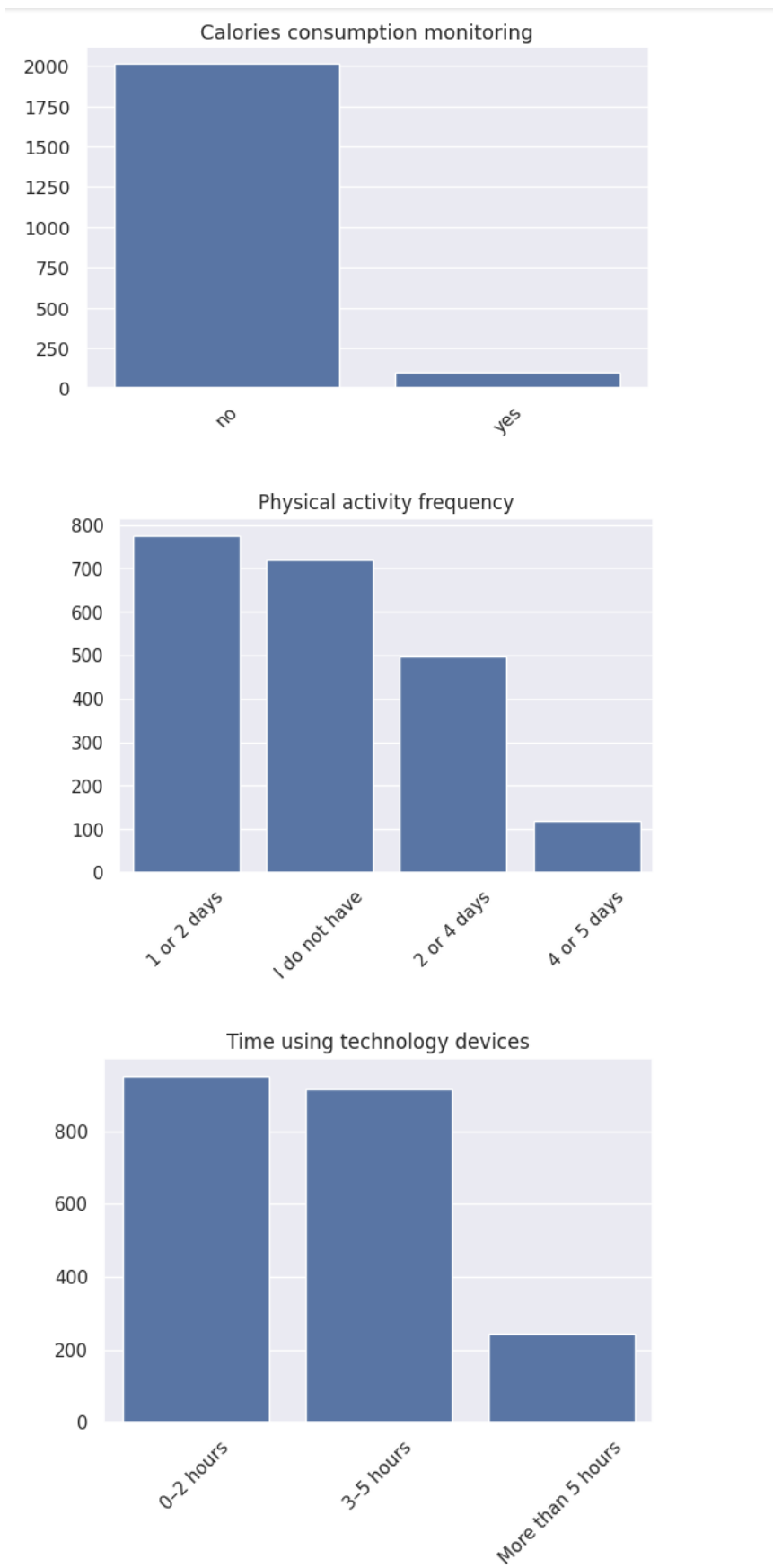


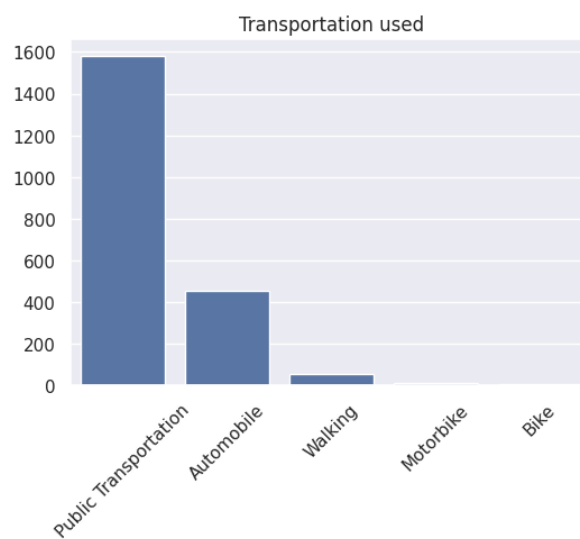
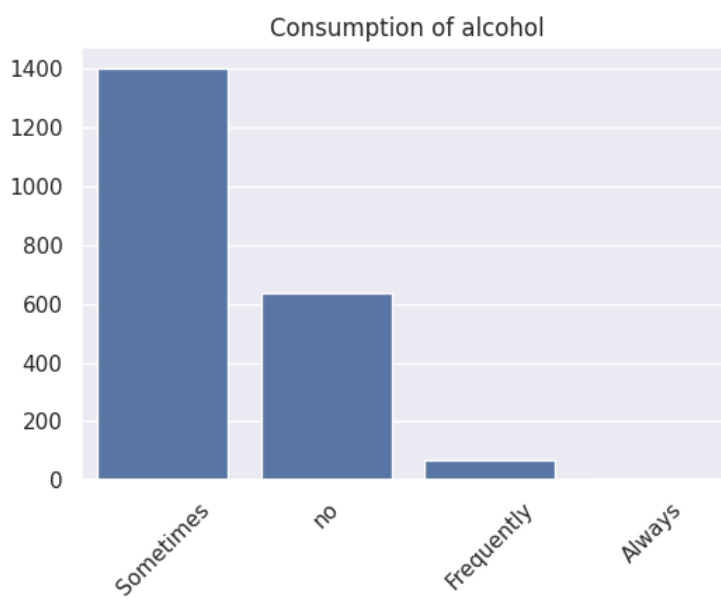
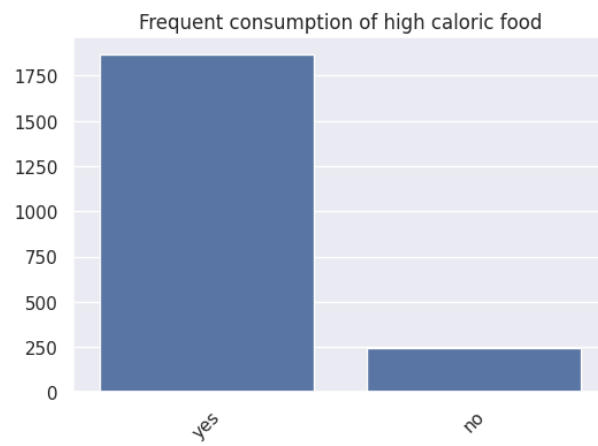
Eating and Exercise Habits

Family History with Overweight ['yes', 'no'] [1726, 385]
 Frequent consumption of high caloric food ['yes', 'no'] [1866, 245]
 Frequency of consumption of vegetables ['Sometimes', 'Always', 'Never'] [1013, 996, 102]
 Number of main meals ['3', '1', '2', '3+'] [1470, 316, 176, 149]
 Consumption of food between meals ['Sometimes', 'Frequently', 'Always', 'no'] [1765, 242, 53, 51]
 Smoke ['no', 'yes'] [2067, 44]
 Consumption of water daily ['Between 1 and 2 L', 'More than 2 L', 'Less than a liter'] [1110, 516, 485]
 Calories consumption monitoring ['no', 'yes'] [2015, 96]
 Physical activity frequency ['1 or 2 days', 'I do not have', '2 or 4 days', '4 or 5 days'] [776, 720, 496, 119]
 Time using technology devices ['0-2 hours', '3-5 hours', 'More than 5 hours'] [952, 915, 244]
 Consumption of alcohol ['Sometimes', 'no', 'Frequently', 'Always'] [1401, 639, 70, 1]
 Transportation used ['Public Transportation', 'Automobile', 'Walking', 'Motorbike', 'Bike'] [1580, 457, 56, 11, 7]









DATA PREPROCESSING

Missing Values

There are no missing values in the dataset.

```
print(df.isnull().sum())
```

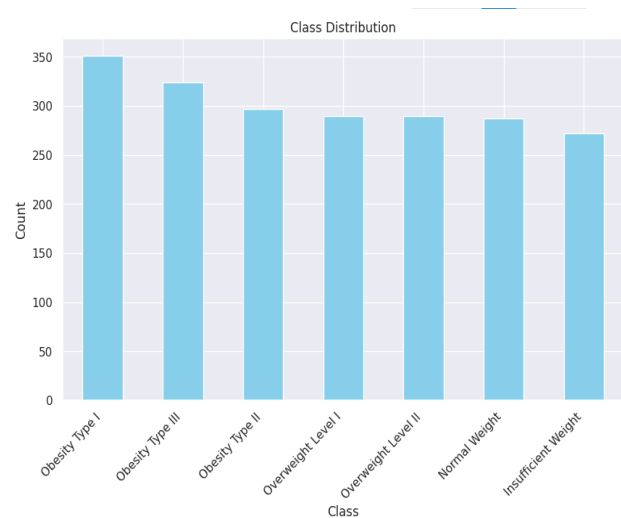
```
Gender      0
Age         0
Height      0
Weight      0
family_history_with_overweight  0
FAVC        0
FCVC        0
NCP         0
CAEC        0
SMOKE       0
CH2O        0
SCC         0
FAF         0
TUE         0
CALC        0
MTRANS      0
NObeyesdad  0
dtype: int64
```

Handling class imbalance

There is no imbalanced class.

```
df['Obesity'].value_counts()
```

```
Obesity
Obesity Type I    351
Obesity Type III  324
Obesity Type II   297
Overweight Level I 290
Overweight Level II 290
Normal Weight     287
Insufficient Weight 272
dtype: int64
```



WORKING OF ALGORITHMS

Algorithms used in ‘Obesity Detection Using Machine Learning’ are Logistic Regression and Random Forest.

Logistic Regression

Logistic regression comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression has several variants, including binary logistic regression, multinomial logistic regression, and ordinal logistic regression. Logistic regression is used for solving the classification problems.

Logistic Function (Sigmoid Function)

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- It maps any real value into another value within a range of 0 and 1.

Working

- Logistic regression works by creating a linear combination of input features, which involves multiplying each feature by a coefficient and summing the results.

Applies the multi-linear function to the input variables

$$z = \left(\sum_{i=1}^n w_i x_i \right) + b$$

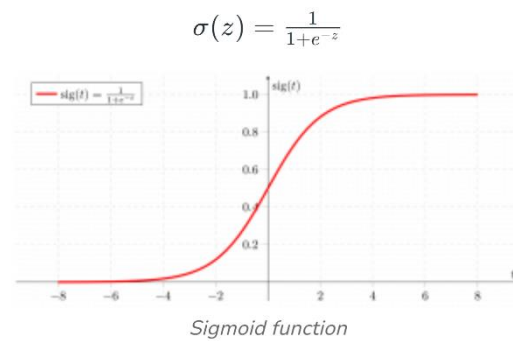
x_i is the i th observation of X

$w_i = [w_1, w_2, w_3, \dots, w_m]$ is the weights or Coefficient

b is the bias term also known as intercept.

- This value is then passed through the logistic (sigmoid) function, which transforms it into a probability value between 0 and 1.

Z will be input to the sigmoid function and find the probability between 0 and 1.



- Sigmoid function converts the continuous variable data into the probability i.e. between 0 and 1.
- For binary classification, this probability is compared to a threshold (usually 0.5) to decide the class label.

Multinomial logistic regression, also known as softmax regression, is used for multi-class classification tasks, where there are more than two possible outcomes for the output variable.

In this case, the softmax function is used in place of the sigmoid function.

Softmax function for K classes will be:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

σ = softmax

\vec{z} = input vector

e^{z_i} = standard exponential function for input vector

K = number of classes in the multi-class classifier

e^{z_j} = standard exponential function for output vector

e^{z_j} = standard exponential function for output vector

- The calculated probabilities will be in the range of 0 to 1.
- The sum of all the probabilities is equals to 1.
- The class with highest probability will be the final output.

Pseudocode

1. Input:

- Training data with N samples and M features.
- Labels (0 or 1) for each sample.
- Learning rate (how big of a step to take when updating weights).
- Number of iterations (how many times to update weights).

2. Initialize:

- Set weights and bias to 0 (or small random numbers).

3. For a set number of iterations:

a. For each sample in the training data:

i. Calculate the weighted sum (z):

$$z = (\text{weight_1} * \text{feature_1}) + (\text{weight_2} * \text{feature_2}) + \dots + (\text{weight_M} * \text{feature_M}) + \text{bias}$$

ii. Apply the sigmoid function to get a probability:

$$\text{probability} = 1 / (1 + \exp(-z))$$

iii. Calculate the error:

$$\text{error} = \text{probability} - \text{actual_label}$$

iv. Update each weight:

$$\text{weight_j} = \text{weight_j} - (\text{learning rate} * \text{error} * \text{feature_j})$$

v. Update the bias:

$$\text{bias} = \text{bias} - (\text{learning rate} * \text{error})$$

4. After finishing all iterations, use the weights and bias to make predictions:

a. For a new sample:

i. Calculate the weighted sum (z) using the final weights and bias.

ii. Apply the sigmoid function to get a probability.

iii. If the probability is 0.5 or higher, output class 1. Otherwise, output class 0.

Random Forest

Random Forest is a supervised learning technique used for both Classification and Regression problems. 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. Instead of relying on one decision tree, the random forest takes the output from each tree and based on the majority votes of outputs, and it takes the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Working

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.

Pseudocode

1. Input:

- Training dataset with N samples and M features.
- Number of trees to create: T.
- Number of features to randomly select at each split: F.

2. Initialize an empty list called 'forest' to store the trees.

3. For each tree (from 1 to T):

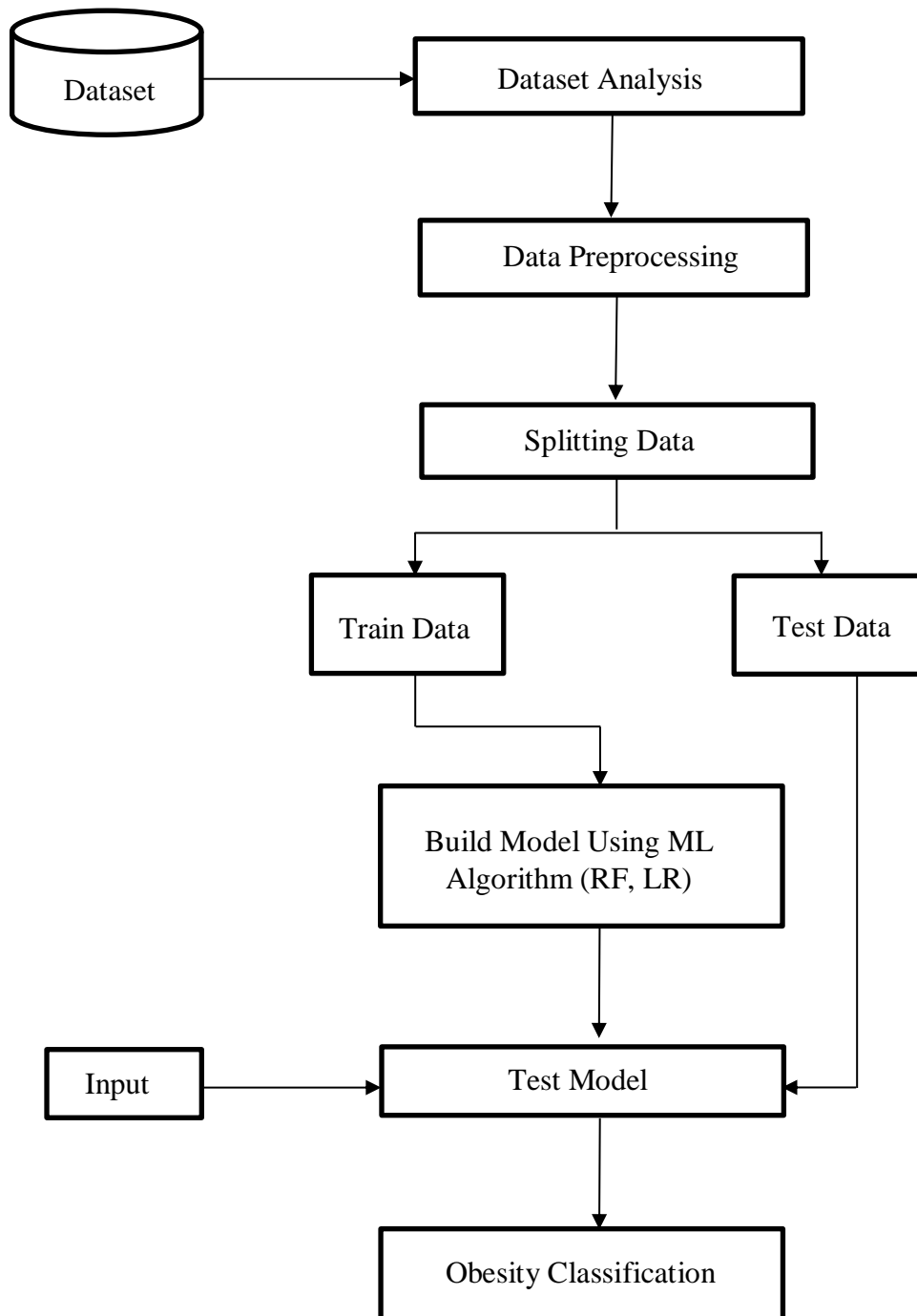
- a. Create a random sample of the training data (some samples may repeat).
- b. Build a decision tree:
 - i. At each decision point in the tree:
 - Randomly pick F features.
 - Find the best feature and split point among the F features to split the node.
 - Split the node into two child nodes based on the selected split point.
 - ii. Keep splitting until the tree is fully grown or meets some stopping condition (like a max depth or minimum samples).
- c. Add this decision tree to the 'forest'.

4. To make a prediction for a new sample x:

- a. Let each tree in the forest make a prediction.
- b. The final prediction is the one that most trees agree on (majority vote).

PACKAGES AND FUNCTIONS

- Pandas
 - read_csv()
 - head()
 - shape()
 - info()
 - drop()
 - value_counts()
 - isnull()
 - describe()
 - tail()
- Matplotlib
 - figure()
 - title()
 - show()
 - xlabel()
 - ylabel()
- Seaborn
 - boxplot()
 - countplot()
 - heatmap()
 - scatterplot()
- Numpy

PROJECT PIPELINE

TIMELINE

- Submission of project synopsis with Journal Papers - 22.07.2024
- project proposal approval - 26.07.2024
- Presenting project proposal before the
Approval Committee - 29.07.2024 & 30.07.2024
- Initial report submission - 12.08.2024
- Analysis and design report submission - 16.08.2024
- Verification of the report and PPT by the guide - 19.08.2024
- First project presentation - 21.08.2024 & 23.08.2024
- Sprint Release I - 30.08.2024
- Sprint Release II - 26.09.2024
- Interim project presentation - 07.10.2024 & 0.10.2024
- Sprint Release III - 18.10.2024
- Project execution, submission of project report and PPT before the guide - 24.10.2024
- Submission of the project report to the guide - 28.10.2024
- Final project presentation - 28.10.2024 & 29.10.2024
- Submission of project report after corrections - 01.11.2024

SYSTEM DESIGN

Model Building

The model planning phase for obesity detection focuses on developing a structured approach for detecting obesity levels using machine learning techniques. It involves selecting algorithms, defining the overall scope, and outlining strategies to accurately classify obesity based on demographic features, eating habits and physical condition features.

1. Objective:

Build an obesity classification system using machine learning techniques to predict and categorize various obesity levels. The goal is to accurately classify obesity such as underweight, normal weight, overweight level I, overweight level II, obesity type I, obesity type II and obesity type III using demographic and eating habit features like weight, height, family history with overweight and other relevant biomarkers.

2. Approach:

Use a comparative study of two machine learning algorithms: Random Forest (RF) and Logistic Regression (LR). These models will be evaluated for accuracy, precision, recall, and robustness in detecting obesity using a pre-processed dataset.

3. Data Preparation:

Import and preprocess the dataset that includes data with physical features and other relevant indicators. Split the dataset into training and test sets for model evaluation.

4. Exploratory Data Analysis (EDA):

Perform EDA to understand the distribution of obesity across the dataset, analyze feature distributions and relationships between physical indicators and different obesity levels to understand patterns in the data that helps in classification.

5. Model Building:

Implement the machine learning algorithms: Random Forest (RF) and Logistic Regression (LR). Train the models using the dataset features and evaluate their performance on the test set using evaluation metrics such as accuracy, confusion matrix and classification report to assess the models' predictive abilities.

6. Model Comparison:

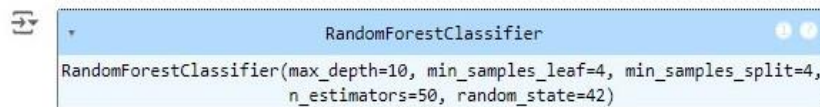
Compare the performance of the two algorithms based on accuracy and speed. Choose the best-performing model based on its ability to accurately detect different obesity levels in unseen data.

Model Training

The dataset was divided into two parts. X representing the input features, and y representing the target variable. The training set consists of 80% of the data and is used to train the model.

Random Forest

```
[ ] # Build a Random Forest Classifier model with reduced complexity
rf_classifier = RandomForestClassifier(
    n_estimators=50,          # Reduce the number of trees
    max_depth=10,            # Limit the maximum depth of the trees
    max_features='sqrt',     # Limit the number of features used to split each node
    min_samples_split=4,     # Increase the minimum samples required to split a node
    min_samples_leaf=4,      # Increase the minimum samples required to be at a leaf node
    random_state=42
)
rf_classifier.fit(X_train, y_train)
```



```
RandomForestClassifier(max_depth=10, min_samples_leaf=4, min_samples_split=4,
n_estimators=50, random_state=42)
```

A Random Forest Classifier, which is an ensemble learning method based on multiple decision trees is used to train the model. The model consists of 50 decision trees. The maximum depth of each tree is limited to 10 levels. The number of features considered for splitting at each node is the square root of the total number of features.



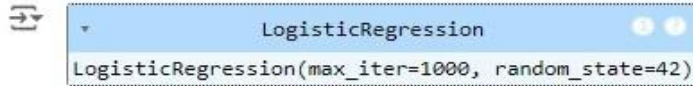
```
[45] print(f"\nRandom Forest Training Accuracy: {train_accuracy_rf*100:.2f}%")

Random Forest Training Accuracy: 98.05%
```

The model achieved an accuracy of 98.05% on the training data.

Logistic Regression

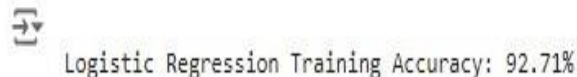
```
[ ] # Logistic Regression Model
lr_classifier = LogisticRegression(max_iter=1000, random_state=42)
lr_classifier.fit(X_train, y_train)
```



```
LogisticRegression(max_iter=1000, random_state=42)
```

Logistic Regression is used to train the model. The model was set to handle multiple classes using the multinomial logistic regression approach. The parameter controls the regularization strength and uses Softmax approaches to classify between multiple conditions.

```
[44] print(f"\nLogistic Regression Training Accuracy: {train_accuracy_lr*100:.2f}%")
```



```
Logistic Regression Training Accuracy: 92.71%
```

The model achieved an accuracy of 92.71% on the training data.

Model Testing

```
[35] # Print accuracy of model using Random Forest
print(f"\nRandom Forest Training Accuracy: {train_accuracy_rf*100:.2f}%")
print(f"\nRandom Forest Testing Accuracy: {test_accuracy_rf*100:.2f}%")
print("\nRandom Forest Classification Report:\n", class_report_rf)
```



```
Random Forest Training Accuracy: 98.05%
Random Forest Testing Accuracy: 92.91%

Random Forest Classification Report:
              precision    recall  f1-score   support

Insufficient Weight      0.95      0.98      0.96         56
Normal Weight            0.86      0.79      0.82         62
Obesity Type I           0.96      0.95      0.95         78
Obesity Type II          0.97      0.97      0.97         58
Obesity Type III         1.00      1.00      1.00         63
Overweight Level I       0.83      0.88      0.85         56
Overweight Level II      0.92      0.94      0.93         50

               accuracy
macro avg       0.93      0.93      0.93        423
weighted avg    0.93      0.93      0.93        423
```

Random forest obtained a testing accuracy of 92.91%.

```

✓ [36] # Print accuracy of model using Logistic Regression
0s print(f"\nLogistic Regression Training Accuracy: {train_accuracy_lr*100:.2f}%")
    print(f"Logistic Regression Testing Accuracy: {test_accuracy_lr*100:.2f}%")
    print("\nLogistic Regression Classification Report:\n", class_report_lr)

```

Logistic Regression Training Accuracy: 92.71%
 Logistic Regression Testing Accuracy: 91.96%

Logistic Regression Classification Report:

	precision	recall	f1-score	support
Insufficient Weight	0.87	0.95	0.91	56
Normal Weight	0.92	0.76	0.83	62
Obesity Type I	0.95	0.99	0.97	78
Obesity Type II	1.00	1.00	1.00	58
Obesity Type III	1.00	1.00	1.00	63
Overweight Level I	0.80	0.91	0.85	56
Overweight Level II	0.89	0.80	0.84	50
accuracy			0.92	423
macro avg	0.92	0.91	0.91	423
weighted avg	0.92	0.92	0.92	423

Logistic Regression obtained an accuracy of 91.96%.