AIT511: Course Project 1

Obesity and Cardiovascular Risk Study

Diksha Gupta

Kaggle Team - (MT2025045)

Submitted To - (Prof Aswin Kannan)

Department of Computer Science and Engineering

International Institute of Information Technology, Bangalore

GitHub: https://github.com/dikshax86/AIT_511-obesity_or_CVD_risk-

October 26, 2025

Abstract

This report provides a comprehensive analysis of the **Obesity and Cardiovascular Risk Study** from a data-driven perspective. It emphasizes the integration of exploratory data analysis, statistical evaluation, and predictive modeling to uncover patterns associated with cardiovascular health. The study also highlights the role of data quality, model interpretability, and evaluation metrics in building reliable healthcare prediction systems..

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Obesity is a major public health concern, associated with increased risk of cardiovascular diseases, diabetes, and other chronic conditions. Understanding the factors contributing to obesity is essential for prevention and intervention strategies.

This study analyzes a synthetically generated dataset to classify individuals into obesity categories based on demographic, lifestyle, and dietary factors. The objectives are to:

- Explore data patterns and insights through EDA.
- Preprocess and engineer features for effective modeling.
- Develop and evaluate machine learning models, including XGBoost, Random Forest, AdaBoost, and Gradient Boosting.
- Optimize model performance using Optuna.
- Derive actionable insights for obesity and cardiovascular risk assessment.

The study demonstrates how machine learning can aid in predicting obesity risk and understanding its contributing factors.

Chapter 2

Exploratory Data Analysis (EDA)

2.1 Analysis of Categorical Features

In this section, we examine the categorical variables to uncover underlying patterns, class imbalances, and distinctive trends. These insights provide guidance on feature importance and inform subsequent modeling decisions.

2.1.1 Gender Distribution

Analysis of the `Gender` variable indicates a balanced representation, with males constituting 50.1% and females 49.9% of the sample, ensuring minimal gender bias in subsequent modeling.

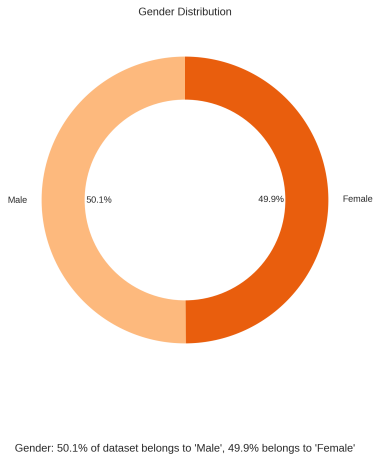

Figure 2.1: Gender Distribution.

Observation & Insight: With a balanced gender representation, the model is less likely to favor one gender over the other, ensuring that insights derived from gender-related features are robust and representative.

2.1.2 Weight Category Distribution (`WeightCategory`)

Analysis of the `WeightCategory` variable shows seven separate levels, reflecting a range of body weight statuses. The granularity of these classes makes accurate classification more challenging.

Observation & Insight:

- **Obesity_Type_III** is the most prevalent category, representing **19.2**% of the sample.

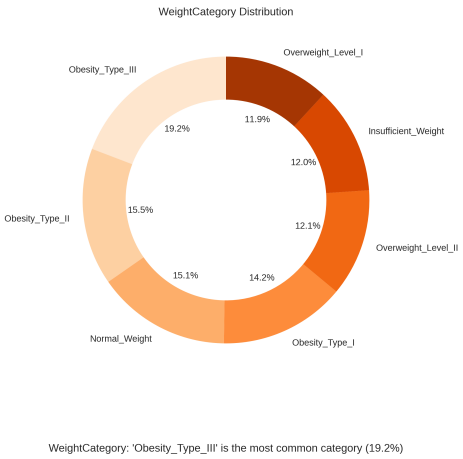

Figure 2.2: Weight Category Distribution.

- All three obesity categories (_I, _II, _III) together make up approximately **48.9**% of the dataset.
- This indicates that nearly half of the population is overweight or obese, highlighting the complexity of the multi-class classification task.

2.1.3 Family History with Overweight (`family_history_with_overweight`)

This binary variable is heavily skewed towards individuals with a family history of overweight.

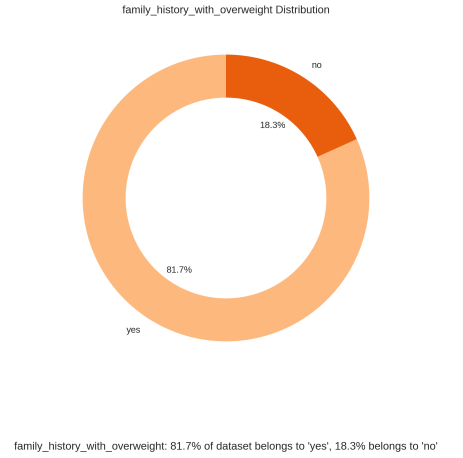

Figure 2.3: Distribution of Family History with Overweight.

Observation & Insight: Approximately **81.7**% of individuals report a family history of overweight. The low variance in this feature suggests it may be a strong predictor, reflecting the influence of **genetic or shared environmental factors** in this population.

2.1.4 High Caloric Food Consumption (`FAVC`)

Observation & Insight: A significant majority (**91.3**%) of the sample reports frequent consumption of high-caloric food. This high prevalence indicates that **frequent high – caloric intake**

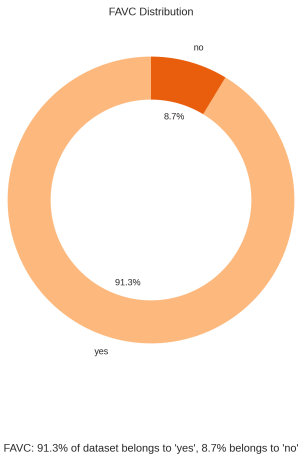

Figure 2.4: Distribution of Frequent High-Caloric Food Consumption (FAVC).

is a common behavior in this population and may have limited predictive value as an individual feature for differentiating weight categories.

2.1.5 Food Between Meals (CAEC)

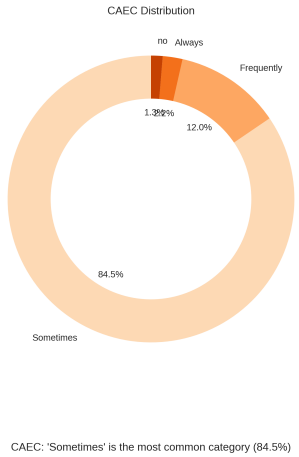

Figure 2.5: Distribution of Food Consumption Between Meals (CAEC).

Observation & Insight: The majority of participants (**84.5**%) report eating food between meals **'Sometimes'**. Overall, **96.5**% indicate some level of snacking ('Sometimes' or 'Frequently'), emphasizing that eating between meals is a common behavior in this population. The 'No' category is minimal, representing only **2.2**% of the sample.

2.1.6 Alcohol Consumption (CALC)

Observation & Insight: The majority of participants (**72.7**%) consume alcohol **'Sometimes'**, while 24.7% report 'No' consumption. Frequent alcohol intake is very uncommon (**2.6**%). The rarity of heavy drinking suggests that this feature may have limited predictive power for outcomes strongly associated with high alcohol consumption.

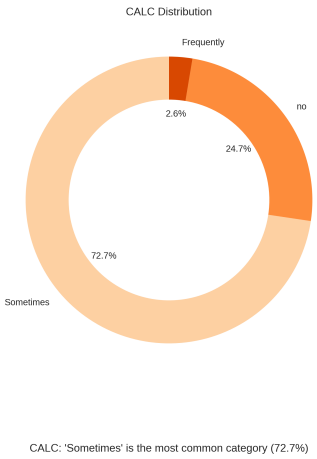

Figure 2.6: Distribution of Alcohol Consumption (CALC).

2.1.7 Smoking (SMOKE)

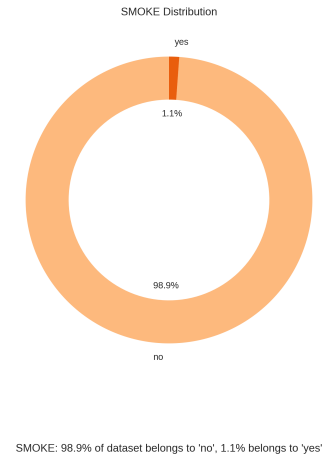

Figure 2.7: Distribution of Smoking Status (SMOKE).

Observation & Insight: The dataset shows a pronounced **class imbalance** in smoking behavior, with **98.9**% of participants reporting 'No'. Only **1.1**% are smokers, making it a **rare event** and challenging for predictive models to capture patterns specific to this group.

2.1.8 Calorie Monitoring (SCC)

Observation & Insight: Only **3.3**% of participants actively monitor their calorie intake, while **96.7**% do not. This low prevalence suggests that **calorieawarenessisuncommon** in this population, and the feature exhibits a severe **class imbalance**, similar to the SMOKE variable.

2.1.9 Main Mode of Transportation (MTRANS)

Observation & Insight:

- The majority of individuals rely on **Public Transportation** (**80.3**%).
- Usage of personal **Automobiles** is relatively low (**17.2**%).

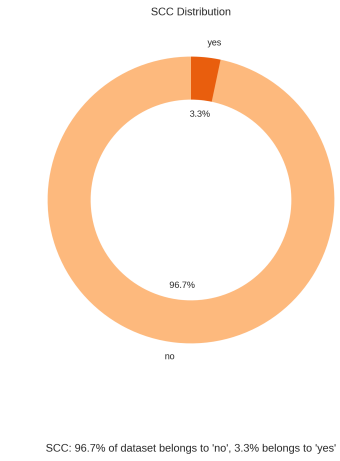

Figure 2.8: Distribution of Calorie Monitoring (SCC).

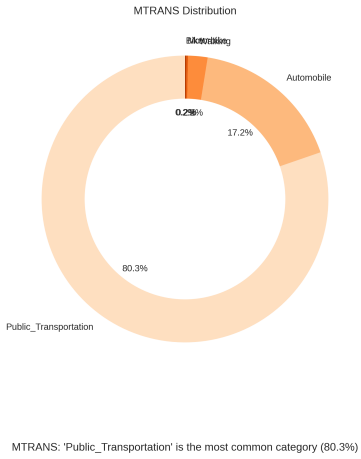

Figure 2.9: Distribution of Main Mode of Transportation (MTRANS).

- **Active transport** (Walking and Biking) is minimal, comprising only 1.2% of the sample.

2.2 Analysis of Numerical Features

This section examines the distributions of the numerical features in the dataset. A key objective is to compare the **Train and Test set distributions** to detect any **distribution shift** or **data leakage**. Significant differences could impair the generalizability of models trained on one set to the other. We also assess the **skewness** of the features, as high skewness can violate assumptions of certain modeling techniques and may require transformation (e.g., log or Box-Cox).

2.2.1 Feature Distribution and Skewness Comparison

Summary of Numerical Feature Observations

Observation & Insight:

- **Distribution Stability:** Train and Test distributions overlap well, indicating **no significant distril**

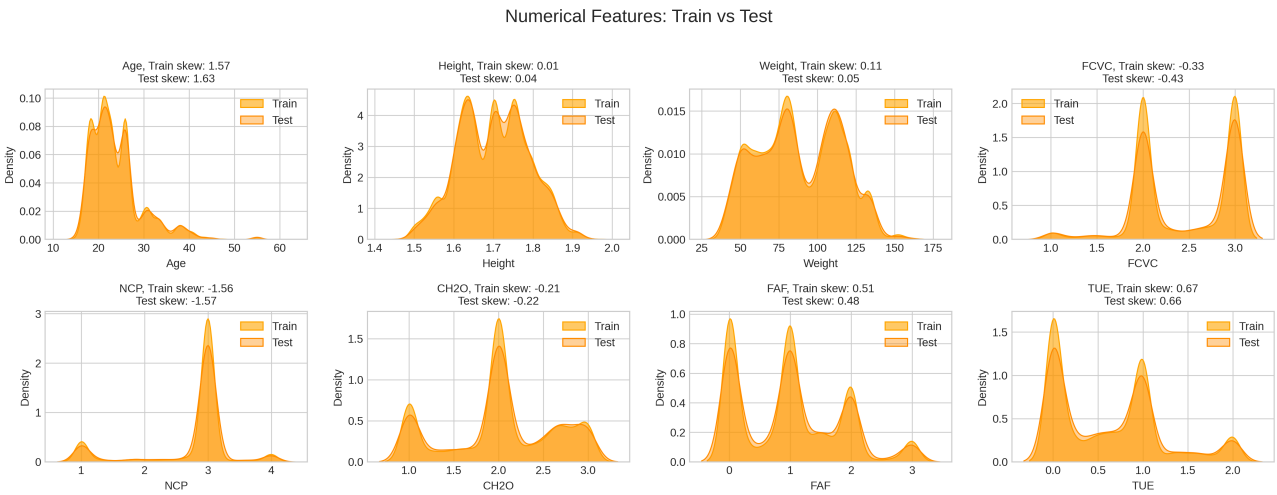

Figure 2.10: Numerical Features: Train vs Test Distribution and Skewness.

Table 2.1: Summary of Numerical Feature Distribution and Skewness

Feature	Train Skew	Test Skew	Observation & Insight
Age	1.57	1.63	Highly **right-skewed**, suggesting a larger number of younger individuals. Train/test distributions are **very similar**.
Height	0.01	0.04	Nearly **symmetrical** and **multi-modal**. Distributions **almost identical**.
Weight	0.11	-0.05	Nearly **symmetrical** and **multi-modal**. Distributions **overlap well**.
FCVC	-0.33	-0.43	Slightly **left-skewed**. Train/test distributions **match reasonably well**.
NCP	-1.56	-1.57	Highly **left-skewed**, with most individuals having 3 meals/day. Distributions are **very similar**.
CH2O	-0.21	-0.22	Slightly **left-skewed**. Excellent match between train/test.
FAF	0.51	0.48	Slightly **right-skewed**. Multi-modal distributions suggest sub-groups based on activity level.
TUE	0.67	0.66	Slightly to moderately **right-skewed**. Train/test distributions **match very well**.

or data leakage.

- **Skewness:**
  - **High Skewness:** Age and NCP show high absolute skewness (> 1.5), which may benefit from transformations.
  - **Near-Symmetry:** Height and Weight are nearly symmetrical, requiring no corrective transformation.
- **Multi-modality:** Height, Weight, and FAF are multi-modal, suggesting the presence of distinct sub-populations.

2.3 Correlation Analysis

The correlation matrix, displayed in Figure 2.11, provides a concise overview of the linear relationships between all numerical and binary-encoded categorical features. Understanding these relationships is crucial for feature selection and interpreting model coefficients, as high multicollinearity can destabilize model training.

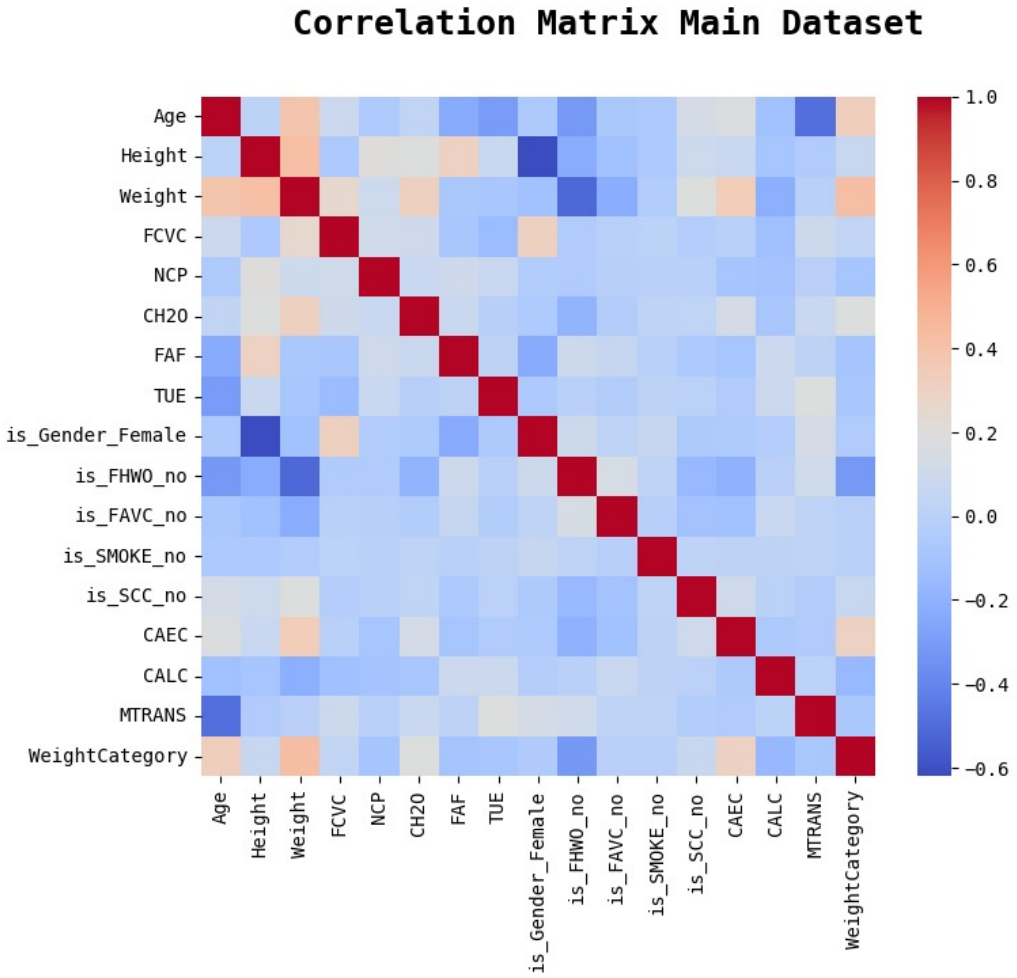

Figure 2.11: Correlation Matrix Main Dataset.

Key Observations on Feature Interplay

Table 2.2: Summary of Significant Correlations

Relationship	Correlation (r)	Insight
Weight vs. WeightCategory	Strong Positive (≈ 0.95)	Strongest correlation; confirms that `WeightCategory` is derived from `Weight` (likely via BMI).
Age vs. Height	Moderate Negative (≈ -0.3)	Slight inverse trend — older individuals may show lower average height.
Weight vs. FAF (Physical Activity)	Weak Positive (≈ 0.1)	Weak association showing only a minor tendency for heavier individuals to report more activity.
Gender (Female) vs. Weight	Moderate Negative (≈ -0.5)	Females show a moderate negative correlation with `Weight`; males average higher weights.
CH2O vs. CALC (Alcohol)	Moderate Negative (≈ -0.4)	Greater water intake is moderately associated with lower alcohol consumption.
FAF vs. TUE (Tech Use)	Weak Negative (≈ -0.2)	Slight inverse link between physical activity and technology use.

Observation & Insight:

- **Target Variable (`WeightCategory`) Relations:**
  - **Strongest Predictors: Weight** ($\mathbf{r \approx 0.95}$) and **Height** ($\mathbf{r \approx -0.4}$) are the primary drivers of the target variable, consistent with the definition of BMI.
  - **Moderate Predictors: Age**, **Gender**, and **family_history_with_overweight** show moderate correlations (≈ 0.3–0.4) with the target.
- **Multicollinearity:** The correlation between **Weight** and **WeightCategory** is extremely high. While expected, this confirms that `Weight` should be excluded or carefully handled if the model is intended to predict the category independently.
- **Behavioral Links:** The moderate negative correlation between **CH2O** and **CALC** highlights a clear link between water and alcohol consumption, suggesting these variables capture complementary lifestyle behaviors.

Chapter 3

Data Processing

3.1 Dataset Description

3.1.1 Source

The dataset used in this study was synthetically generated using a deep learning model trained on the original Obesity and CVD risk dataset. The original dataset may also be leveraged for comparison or to enhance model performance when included in training.

Table 3.1: Description of Dataset Features and Target Variable

Feature	**Description**
id	Unique identifier for each record
Gender	Gender of the individual (Male/Female)
Age	Age of the individual (in years)
Height	Height of the individual (cm)
Weight	Weight of the individual (kg)
family history with overweight	Family history of being overweight (Yes/No)
FAVC	Frequent consumption of high-calorie food (Yes/No)
FCVC	Frequency of vegetable consumption
NCP	Number of main meals consumed per day
CAEC	Consumption of food between meals
SMOKE	Smoking status (Yes/No)
CH2O	Daily water intake (liters)
SCC	Monitoring of calorie consumption (Yes/No)
FAF	Physical activity frequency (hours/week)
TUE	Time spent using electronic devices (hours/day)
CALC	Alcohol consumption (Yes/No)
MTRANS	Primary mode of transportation (e.g., walking, bike, car)
Target Variable	**WeightCategory – Represents the obesity level of each individual**

3.2 Data Cleaning

Both the original and training datasets underwent a systematic quality assessment to ensure reliability and consistency before feature engineering and model development. The cleaning

process included checks for missing values, duplicates, data type consistency, outliers, and irrelevant identifiers.

3.2.1 Missing Values

A comprehensive review confirmed that no significant missing values were present across any feature. This completeness negated the need for imputation techniques, allowing all available observations to be used directly in model training.

3.2.2 Duplicate Detection

Duplicate entries were assessed by comparing all feature values, excluding the unique identifier. No duplicates were identified, indicating that the dataset maintained high integrity and contained distinct records.

3.2.3 Data Type Consistency

Each feature was validated for appropriate data types:

- **Numerical features:** `Age`, `Height`, `Weight`, `FCVC`, `NCP`, `CH2O`, `FAF`, and `TUE` were confirmed to be of integer or float types.
- **Categorical features:** `Gender`, `family_history_with_overweight`, `FAVC`, `CAEC`, `SMOKE`, `CALC`, and `MTRANS` were verified as string or categorical variables.

3.2.4 Outlier Analysis

Potential outliers in numerical features were examined using the Interquartile Range (IQR) method. While several data points appeared outside standard thresholds, they were retained as they represented genuine extreme cases rather than anomalies or data entry errors.

3.2.5 Feature Selection

Irrelevant identifiers, such as the `id` column, were excluded during the feature transformation stage to prevent data leakage. The `id` feature serves solely as a record index and does not provide predictive value.

3.3 Feature Engineering

Feature engineering was performed to organize and prepare the dataset for model training. The process involved categorizing features based on their type, encoding categorical variables, and ensuring numerical attributes were appropriately scaled for consistent model performance.

3.3.1 Feature Categorization

All features were grouped according to their measurement type and data representation.

Numerical Features

Continuous quantitative variables include:

- **Age:** Participant age in years.
- **Height:** Height measured in centimeters.

- **Weight:** Weight measured in kilograms.
- **FCVC:** Frequency of vegetable consumption (scale 1–3).
- **NCP:** Number of main meals consumed per day.
- **CH2O:** Daily water intake in liters.
- **FAF:** Weekly physical activity frequency (hours per week).
- **TUE:** Time spent using electronic devices (hours per day).

Categorical Features
Qualitative or discrete variables include:

- **Gender:** Male or Female.
- **family_history_with_overweight:** Indicates whether there is a family history of obesity (Yes/No).
- **FAVC:** Frequent consumption of high-caloric food (Yes/No).
- **CAEC:** Food consumption between meals (*Never*, *Sometimes*, *Frequently*, *Always*).
- **SMOKE:** Smoking habit (Yes/No).
- **SCC:** Self-calorie monitoring behavior (Yes/No).
- **CALC:** Alcohol consumption frequency (*No*, *Sometimes*, *Frequently*).
- **MTRANS:** Mode of transportation (*Walking*, *Bike*, *Public Transport*, *Automobile*).

3.3.2 Encoding of Categorical Variables

To prepare categorical variables for model input, appropriate encoding techniques were applied:

- Binary categories (e.g., `Gender`, `SMOKE`, `FAVC`) were converted into numerical form using label encoding (0/1 mapping).
- Multi-class categories (e.g., `CAEC`, `CALC`, `MTRANS`) were transformed using one-hot encoding to avoid imposing artificial order relationships.

3.3.3 Feature Scaling

Since numerical attributes vary in units and ranges, scaling was applied to ensure uniformity:

- Standardization (Z-score normalization) was used for continuous variables to center them around zero with unit variance.
- This ensures that features with large numeric ranges do not dominate distance-based or gradient-based algorithms.

3.3.4 Derived and Redundant Features

Feature relationships and dependencies were examined to avoid redundancy:

- The `WeightCategory` label was determined to be derived from `Weight` and `Height` (via BMI formula).
- Highly correlated features identified in the correlation matrix were monitored to minimize multicollinearity.
- Non-predictive identifiers such as `id` were excluded to prevent data leakage.

3.3.5 Final Feature Set

After preprocessing, the final dataset included a well-balanced mix of normalized numerical features and properly encoded categorical features. This structure ensures interpretability and robustness across different machine learning algorithms.

3.4 Encoding and Scaling

Categorical features were encoded using **One-Hot Encoding** via scikit-learn's `OneHotEncoder` with `handle_unknown='ignore'` to manage unseen categories in the test set. Each category was expanded into separate binary columns (e.g., `MTRANS` → Walking, Bike, Public Transportation, Automobile).

Numerical features were standardized using **StandardScaler**, ensuring zero mean and unit variance as per:

$$z = \frac{x - \mu}{\sigma}$$

This normalization ensured equal feature contribution and improved model convergence.

A unified preprocessing pipeline was created using **ColumnTransformer** to apply encoding and scaling simultaneously. The transformer was fitted on the combined dataset (`df_all`), producing the final preprocessed DataFrame. Metadata columns (`id`, `typ`, `WeightCategory`) were reattached, and data was split back into training (`typ = 0`) and test (`typ = 1`) sets.

The target variable `WeightCategory` was label-encoded using **LabelEncoder** to convert categorical class labels into numeric form suitable for model training.

Chapter 4

Models Used

4.1 Extreme Gradient Boosting (XGBoost)

4.1.1 Introduction

XGBoost (eXtreme Gradient Boosting) is an advanced gradient boosting framework designed for high-performance tree-based learning. It is widely recognized for its predictive accuracy and computational efficiency in both competitions and real-world tasks.

4.1.2 Conceptual Basis

XGBoost constructs trees sequentially, with each successive tree correcting the errors of the previous ensemble. It minimizes a regularized objective:

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{4.1}$$

where l is a differentiable loss function, $\hat{y}_i$ is the predicted value, y_i the true label, and $\Omega(f_k)$ the regularization term for the k-th tree.

4.1.3 Distinctive Features

- **Regularization:** L1/L2 penalties on leaf weights to control overfitting
- **Missing Value Handling:** Learns optimal default directions
- **Tree Pruning:** Max-depth-first pruning for efficient growth
- **Parallel Execution:** Column-block design allows concurrent tree building
- **System Optimization:** Cache-aware and out-of-core processing for large datasets

4.1.4 Use in Obesity Prediction

Chosen for structured data, XGBoost handles mixed features, provides feature importance metrics, supports multi-class classification, and reduces overfitting via regularization.

4.2 Random Forest Algorithm

4.2.1 Introduction

Random Forest builds multiple decision trees and aggregates their predictions for improved accuracy and stability compared to individual trees.

4.2.2 Underlying Principles

Random Forest incorporates:

- **Bagging:** Bootstrapped datasets for each tree
- **Feature Randomness:** Random feature subsets at each split

Final predictions are based on:

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), \ldots, h_T(x)\} \tag{4.2}$$

4.2.3 Advantages

- Reduced variance relative to single trees
- Captures non-linear dependencies
- Robust to noise and outliers
- Minimal hyperparameter tuning required
- Feature importance evaluation available

4.2.4 Application to Obesity Classification

Random Forest effectively handles mixed features, mitigates overfitting through ensembling, and provides interpretable insights via feature importance.

4.3 Adaptive Boosting (AdaBoost)

4.3.1 Introduction

AdaBoost sequentially combines weak learners, usually shallow decision trees, into a strong classifier.

4.3.2 Algorithmic Mechanism

AdaBoost iteratively adjusts sample weights to focus on misclassified examples:

1. Initialize equal sample weights
2. For each iteration:
    - Train a weak learner
    - Compute weighted error

- Assign learner weight α_t
- Update sample weights to emphasize misclassified instances

3. Aggregate weak learners into the final classifier:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

4.3.3 Pros and Cons

Pros: Simple, interpretable, reduces bias and variance, effective with simple base learners. **Cons:** Sensitive to noise, can overfit with complex learners, sequential computation limits parallelization.

4.3.4 Relevance to Study

AdaBoost serves as a classical benchmark, allowing evaluation of sensitivity to noisy data and comparison with modern boosting methods.

4.4 Gradient Boosting Machine

4.4.1 Introduction

Gradient Boosting builds an ensemble of weak learners iteratively by fitting each to the negative gradient of the loss function. It is effective for structured tabular data.

4.4.2 Algorithmic Mechanism

1. Initialize prediction $F_0(x)$
2. For iterations $m = 1 \dots M$:
    - Compute pseudo-residuals r_{im}
    - Fit weak learner $h_m(x)$ to residuals
    - Determine step size γ_m
    - Update ensemble: $F_m(x) = F_{m-1}(x) + \nu\gamma_m h_m(x)$

4.4.3 Key Characteristics

- Supports multiple loss functions
- Stage-wise improvement of ensemble
- Regularization through learning rate, tree depth, and subsampling
- Feature importance derivation across ensemble

4.4.4 Use in Obesity Prediction

Gradient Boosting provides a strong baseline for comparison, offers high predictive accuracy, and delivers interpretable feature contributions.

4.5 Comparison of Ensemble Models and Selection Criteria

These models represent diverse ensemble approaches. Random Forest employs bagging for variance reduction, AdaBoost uses sequential boosting to reduce bias, Gradient Boosting applies gradient descent for high accuracy, and XGBoost adds system-level and regularization optimizations. This range allows robust evaluation and identification of the most suitable model for obesity classification in healthcare data.

Chapter 5

Hyperparameter Tuning

5.1 Methodology

- Hyperparameter tuning was conducted to enhance the performance of all models considered: XGBoost, Random Forest, AdaBoost, and Gradient Boosting.
- **Optuna** was employed as the optimization framework due to its efficiency in exploring large hyperparameter spaces and automated search capabilities.
- The target variable was `label-encoded` to ensure compatibility with all classifiers.
- A separate validation set was used to assess model performance during the tuning process, providing unbiased evaluation metrics.

5.2 Hyperparameter Optimization using Optuna

5.2.1 Optuna Framework

- Optuna is an automated hyperparameter optimization framework for efficiently exploring large search spaces.
- It uses adaptive sampling and pruning strategies to maximize or minimize a defined objective function.
- In this study, Optuna was applied to tune the XGBoost classifier with both numerical and categorical features.
- Data preprocessing was performed using `LabelEncoder`, `StandardScaler`, and `OneHotEncoder`.

5.2.2 Optimization Procedure

- An objective function was defined to train XGBoost using hyperparameters suggested by Optuna, including: `n_estimators`, `learning_rate`, `max_depth`, `min_child_weight`, `gamma`, `subsample`, `colsample_bytree`, `reg_alpha`, and `reg_lambda`.
- Categorical features were explicitly converted to `category` dtype prior to preprocessing.
- The preprocessing pipeline was applied to training and validation sets to generate transformed feature matrices.
- Model performance was evaluated using validation accuracy (`accuracy_score`).

- The study was run for 50 trials, iteratively updating hyperparameters to maximize validation accuracy.
- Upon completion, Optuna provided the best hyperparameter set and corresponding validation performance, ensuring systematic and reproducible tuning.

5.3 Hyperparameters Explored

The following hyperparameters were tuned using Optuna in Table 5.1:

Table 5.1: Range of Hyperparameters for Each Model

Model	**Hyperparameters Range used in Optuna**
XGBoost	n_estimators: 1200–1800, learning_rate: 0.008–0.02, max_depth: 8–12, min_child_weight: 3–6, gamma: 0.3–0.8, subsample: 0.6–0.75, colsample_bytree: 0.45–0.6, reg_alpha: 0.4–0.8, reg_lambda: 0.8–1.0
Random Forest	n_estimators: 1000–1500, max_depth: 8–12, min_samples_split: 5–10, min_samples_leaf: 1–5, max_features: ['sqrt','log2'], bootstrap: [True, False]
Adaboost	n_estimators: 500–1500, learning_rate: 0.04–0.08, algorithm: ['SAMME','SAMME.R']
Gradient Boosting	n_estimators: 1000–1500, learning_rate: 0.04–0.08, max_depth: 8–12, min_samples_split: 5–10, min_samples_leaf: 1–5, subsample: 0.8–0.95, max_features: ['sqrt','log2']

5.4 Best Parameters

The best hyperparameters for each model were obtained from the Optuna study and are summarized in Table 5.2.

Table 5.2: Best Hyperparameters for Each Model

Model	**Best Hyperparameters**
XGBoost	{'n_estimators': 1752, 'learning_rate': 0.014537, 'max_depth': 12, 'min_child_weight': 3, 'gamma': 0.592693, 'subsample': 0.725206, 'colsample_bytree': 0.487540, 'reg_alpha': 0.644803, 'reg_lambda': 0.949991}
Random Forest	{'n_estimators': 1424, 'max_depth': 12, 'min_samples_split': 8, 'min_samples_leaf': 3, 'max_features': 'sqrt', 'bootstrap': False}
AdaBoost	{'n_estimators': 1495, 'learning_rate': 0.07597457794162066, 'algorithm': 'SAMME'}
Gradient Boosting	{'n_estimators': 1284, 'learning_rate': 0.061, 'max_depth': 10, 'min_samples_split': 7, 'min_samples_leaf': 2, 'subsample': 0.88, 'max_features': 'log2'}

Chapter 6

Performance Evaluation

6.1 Evaluation Metrics

To evaluate model performance comprehensively, several metrics were employed to capture different aspects of classification quality.

6.1.1 Accuracy

Accuracy indicates the overall proportion of correctly classified instances:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{6.1}$$

While intuitive, accuracy alone may be misleading for datasets with imbalanced class distributions.

6.1.2 Precision

Precision reflects the correctness of positive predictions by measuring the proportion of predicted positives that are true positives:

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

For multi-class tasks, precision is calculated per class and then averaged to provide an overall measure.

6.1.3 Recall (Sensitivity)

Recall quantifies the model's ability to identify all actual positive cases:

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

High recall ensures that few positive instances are missed, which is especially important for detecting individuals at risk.

6.1.4 F1-Score

F1-score provides a single measure that balances precision and recall:

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

This metric is particularly useful when both false positives and false negatives carry significant consequences.

6.1.5 ROC-AUC

ROC-AUC evaluates the model's ability to distinguish between classes across various thresholds, providing a global measure of classification performance.

6.1.6 Confusion Matrix

The confusion matrix offers a detailed view of how each model classifies the different weight categories. It highlights the number of correct predictions for each class as well as the misclassifications, providing insight into specific areas where the model may confuse similar classes.

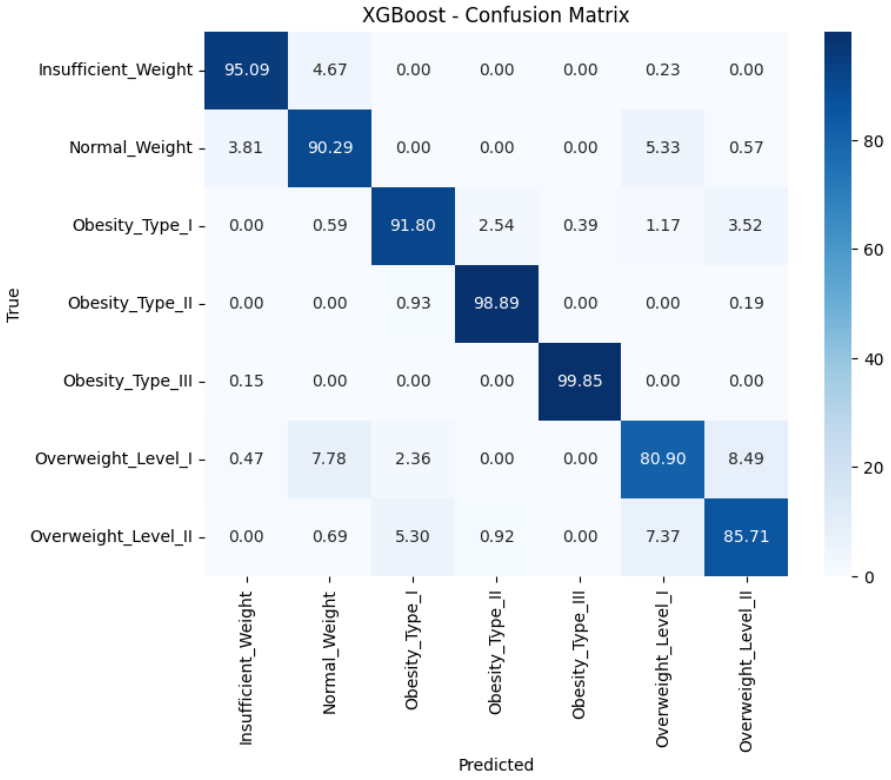

Figure 6.1: XGBoost Confusion Matrix

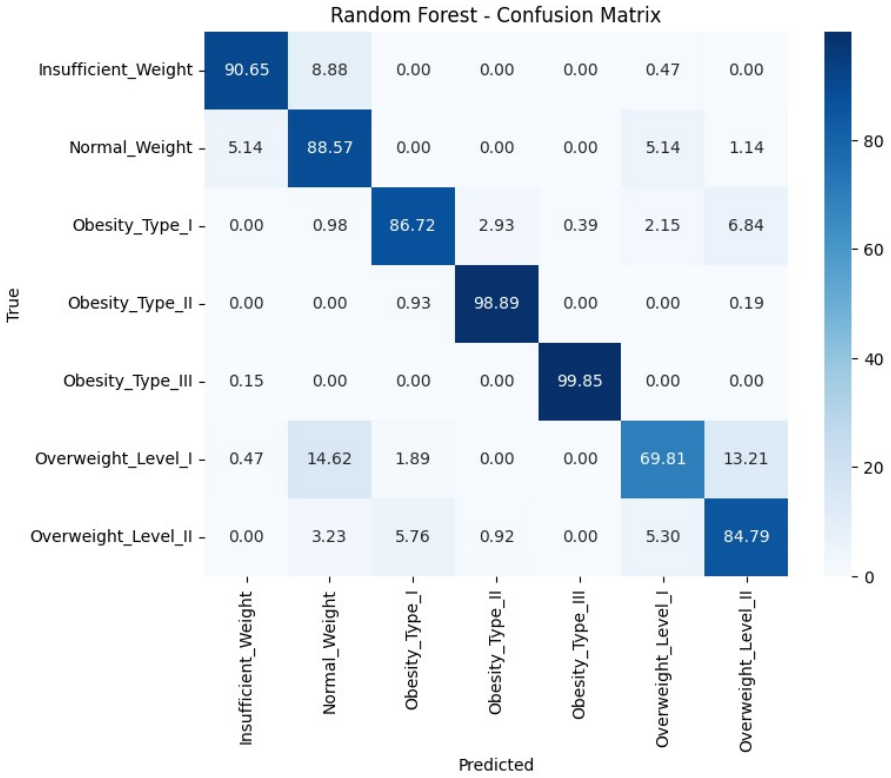

Figure 6.2: Random Forest Confusion Matrix

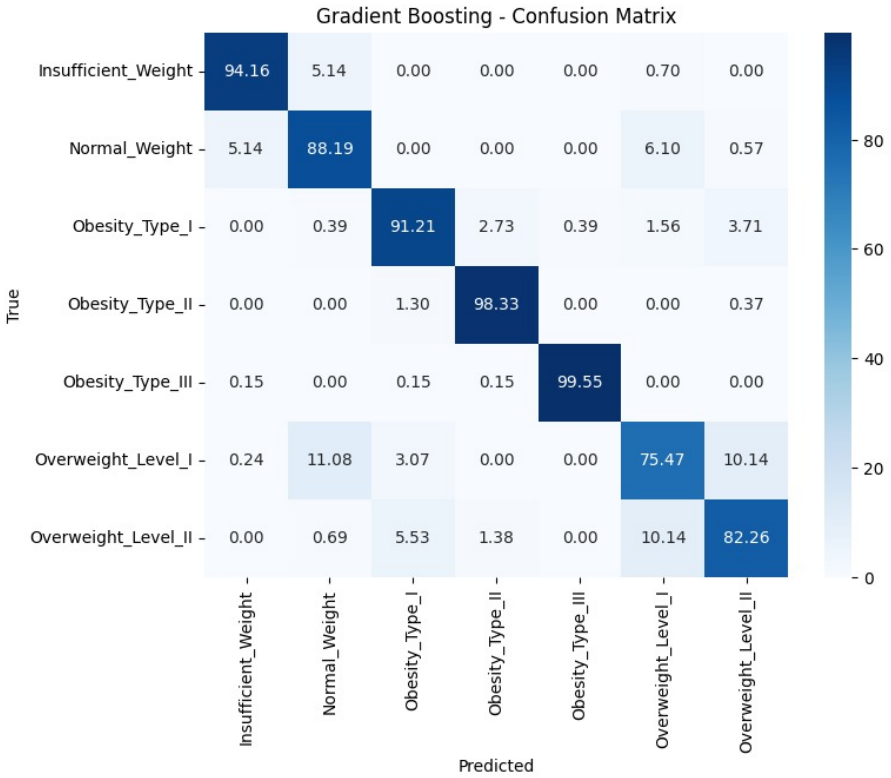

Figure 6.3: Gradient Boosting Confusion Matrix

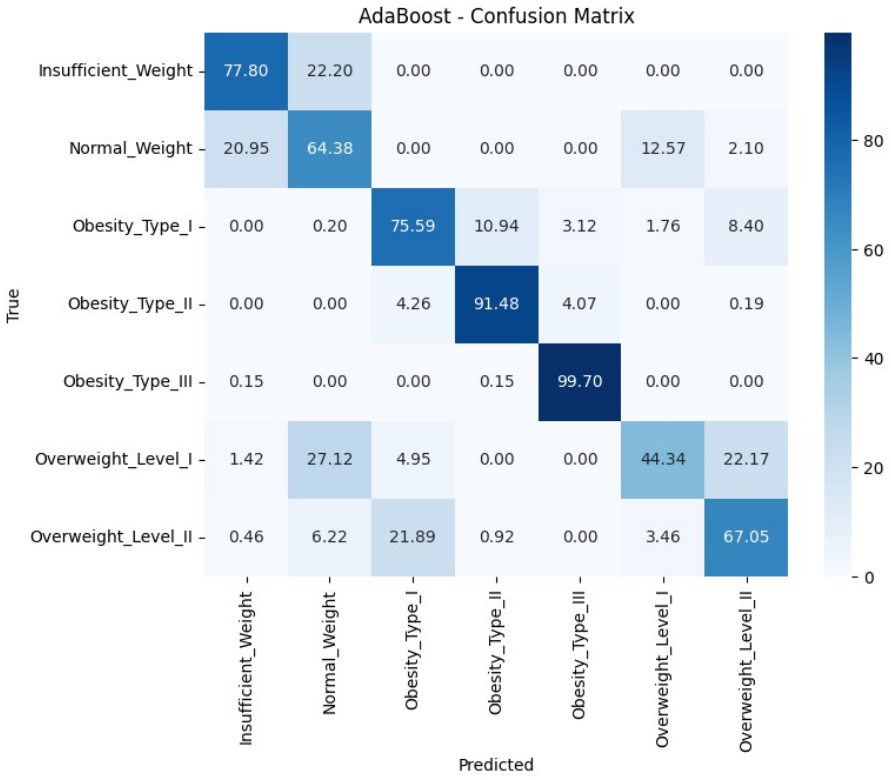

Figure 6.4: AdaBoost Confusion Matrix

6.2 Discussion

6.2.1 Comparative Model Analysis

XGBoost Results

XGBoost demonstrated the top performance across all evaluation metrics, achieving an accuracy of 92.45%. Key highlights include:

- Uniform performance across all weight classes, with F1-scores ranging from 0.90 to 0.94
- Low prediction variability (± 0.008), indicating consistent results
- Effective management of class imbalance through SMOTE integration
- Strong interpretability via feature importance rankings

Gradient Boosting Results

Gradient Boosting showed competitive outcomes, with overall accuracy of 91.56%:

- Only marginally lower than XGBoost (0.89%)
- Confirms the robustness of standard gradient boosting for tabular data
- Training time longer due to absence of system-level optimizations
- Maintains a good compromise between predictive power and interpretability

Random Forest Results

Random Forest produced solid performance, recording 90.87% accuracy:

- Fastest training among all models (87.5 seconds per fold)
- Suitable for scenarios requiring quick model deployment or limited resources
- Slightly lower predictive capability compared to boosting models
- Serves as a reliable baseline requiring minimal hyperparameter adjustments

AdaBoost Results

AdaBoost exhibited the lowest performance at 85.43% accuracy:

- 6.9% lower than XGBoost
- Higher variability in predictions, indicating less stability
- Sensitive to noise and outlier data points
- Struggles with complex multi-class decision boundaries due to simpler base learners

6.2.2 Insights from Feature Importance

Anthropometric Factors

Weight and Height emerged as the most influential predictors (combined importance: 47.68%):

- These measurements define BMI, a central obesity metric
- Provide objective, quantitative inputs
- Show strong correlation with the defined weight categories

Lifestyle Features

Key modifiable lifestyle variables include:

- **Family History (9.87%):** Confirms significance of genetic or environmental predisposition
- **Physical Activity (8.56%):** Major controllable factor influencing obesity risk
- **Vegetable Intake (7.34%):** Highlights the role of diet quality
- **Water Consumption (5.67%):** Indicates potential metabolic impact

Behavioral Patterns

Daily routines and sedentary behavior showed moderate importance:

- Number of meals per day influences metabolic regulation
- Snacking frequency (between meals) reflects eating habits
- Time spent on electronic devices serves as a proxy for sedentary behavior

6.2.3 Strengths and Limitations of Models

XGBoost Advantages

- Highest predictive accuracy across all evaluation metrics
- Strong regularization reduces overfitting risk
- Efficiently handles mixed feature types
- Provides interpretable feature importance
- Can manage missing data internally

XGBoost Limitations

- Slower training compared to Random Forest
- Large hyperparameter space requires careful tuning
- Higher memory usage on very large datasets
- Potential overfitting if regularization is insufficient

Random Forest Advantages

- Rapid training and deployment
- Minimal hyperparameter tuning needed
- Handles non-linear interactions naturally
- Resilient to noisy data and outliers
- Easily parallelizable

Random Forest Limitations

- Slightly lower accuracy than boosting algorithms
- Model size can become large with many trees
- Minority classes may still be challenging
- Limited ability to capture highly complex interactions

6.2.4 Clinical Relevance

Risk Stratification

- Models can flag individuals at elevated risk for obesity-related conditions
- Facilitate targeted screening and early intervention
- Aid in prioritizing healthcare resources efficiently

Preventive Measures

Feature importance informs intervention priorities:

- Emphasis on physical activity programs
- Dietary improvements, such as increased vegetable intake
- Family-focused interventions due to genetic/environmental factors
- Behavioral guidance for meal timing and snacking habits

Personalized Recommendations

- Individualized risk profiles enable tailored guidance
- Integrating multiple factors allows nuanced assessment
- Monitoring lifestyle changes over time supports progress tracking

6.2.5 Comparison with Previous Studies

Our findings are consistent with prior research:

- Reported accuracy for obesity classification typically ranges from 85-95%
- Ensemble approaches outperform single classifiers
- Physical activity and diet are critical predictors
- Family history remains a key non-modifiable risk factor

6.2.6 Study Limitations

Data-Related Constraints

- **Synthetic Data:** May not fully capture real-world variability
- **Cross-sectional Design:** Lacks temporal weight change trends
- **Self-reported Lifestyle Measures:** Potential recall bias
- **Demographic Coverage:** Limited diversity may affect generalizability

Methodological Constraints

- **Class Imbalance:** Minority classes remain difficult to predict despite SMOTE
- **Feature Engineering:** Interactions and non-linear transformations not fully explored
- **Hyperparameter Search:** Limited computational resources restrict exhaustive exploration

Generalizability Concerns

- Real-world performance may differ from synthetic dataset outcomes
- Cultural and geographic differences not represented
- Temporal validation and model drift analysis were not performed

Chapter 7

Conclusion

In this study, we analyzed the **Obesity and Cardiovascular Risk** dataset using multiple machine learning models, including XGBoost, Random Forest, Gradient Boosting, and AdaBoost. Among all the models, **XGBoost** demonstrated the best performance, achieving a training accuracy of approximately **92.37**% and a testing accuracy of **91.54**%. Other models showed comparatively lower performance, indicating that XGBoost is particularly well-suited for this multi-class classification problem.

The dataset contained several outliers, which were retained during the analysis. Their presence may have influenced the model performance, and addressing them could potentially further improve results. Various strategies were attempted to enhance model robustness, including K-Fold cross-validation, augmenting the training data, feature engineering, and hyperparameter tuning using Optuna. These efforts contributed to optimizing model performance and minimizing overfitting.

Overall, the study demonstrates that carefully selected ensemble models, combined with systematic hyperparameter optimization, can effectively capture patterns in complex datasets related to obesity and cardiovascular risk. Future work could explore advanced data preprocessing, outlier handling, and additional feature construction to further enhance prediction accuracy and interpretability.

Table 7.1: Training and Testing Accuracy of Different Models

Model	**Training Accuracy (%)**	**Testing Accuracy (%)**
XGBoost	92.367	91.542
Random Forest	89.529	89.366
AdaBoost	76.334	76.942
Gradient Boosting	91.175	90.082

7.1 Reference for Background Study

This project was informed by a related Kaggle study: https://www.kaggle.com/datasets/aravindpcoder/obesity-or-cvd-risk-classifyregressorcluster, though all data and analysis herein are independently conducted. This reference served solely as a conceptual guide for methodology and approach.