Classification of Weight Categories

Course Project Report

Department of Computer Science and Engineering
International Institute of Information Technology, Bangalore

Submitted by:
Dharani Prasad S
Roll No: MT2025043

Abhishek Prasanna
Roll No: MT2025007

Course: AIT-511 Machine Learning

October 25, 2025

Contents

Chapter 1

Abstract

This project aims to develop a machine learning-based model capable of classifying individuals into distinct weight categories based on demographic, lifestyle, and physiological data. The approach integrates feature engineering, data preprocessing, and ensemble learning methods to achieve high predictive accuracy. Body Mass Index (BMI) was engineered as a key feature to improve model interpretability and performance. Multiple models were compared, and the best configuration was selected through systematic hyperparameter tuning. The results demonstrate the significance of feature engineering and model optimization in improving classification accuracy.

The preprocessing pipeline was crucial for handling the mixed data types present in the dataset. **Categorical features** were handled using appropriate encoding techniques (OrdinalEncoder and OneHotEncoder) to convert them into a numerical format suitable for machine learning algorithms. Furthermore, all **numerical features** were standardized using `StandardScaler` to ensure the models were not unduly influenced by differences in feature scale. This foundational preprocessing work ensured that the subsequent ensemble methods, particularly the high-performing **XGBoost Classifier**, could operate efficiently and robustly. The entire workflow, from BMI calculation to final model prediction, was managed within a structured scikit-learn pipeline for deployment readiness and reproducibility.

Chapter 2

Introduction

The increasing prevalence of obesity and weight-related health conditions necessitates computational models capable of understanding and predicting weight categories from behavioral and physiological indicators. The objective of this study is to design and implement a machine learning pipeline that classifies individuals into weight categories using structured data features such as age, activity level, and dietary habits.

This report outlines the data preprocessing steps, feature engineering techniques, model training strategies, and performance evaluation metrics used. The overall aim is to achieve a high-performing and interpretable model suitable for real-world applications.

The methodology centers on integrating robust feature engineering—specifically calculating the **Body Mass Index (BMI)**—with advanced **ensemble learning techniques**. The pipeline ensures that mixed data types, including categorical and numerical features, are handled appropriately through standardized encoding and scaling prior to model ingestion. Multiple XGBoost and Random Forest configurations are systematically compared and optimized via Grid Search to maximize predictive capacity while maintaining model stability and generalizability across diverse data subsets. The final section discusses the comparative performance of the models, highlighting the significance of feature transformation and hyperparameter tuning in realizing the full potential of the classification system.

Chapter 3

Dataset Overview

The dataset contains both numerical and categorical features describing participants' lifestyle, dietary patterns, and demographic characteristics.

3.1 Data Description

- **Number of samples:** 15533
- **Number of features:** 18
- **Target variable:** WeightCategory (multi-class)
- **Feature types:** Numerical (e.g., Age, Height, Weight), Categorical (e.g., Gender, SMOKE, FAVC, MTRANS)

3.2 Missing Values and Cleaning

Basic data cleaning included removal of missing or inconsistent entries, conversion of categorical data into consistent formats, and type normalization across all columns.

Chapter 4

Data Preprocessing and Feature Engineering

4.1 Feature Engineering

The most critical feature engineering step was calculating the Body Mass Index (BMI) using the formula:

$$\text{BMI} = \frac{\text{Weight (kg)}}{(\text{Height (m)})^2}$$

BMI was introduced as a key feature to enhance the model's understanding of body composition. Original columns *Weight*, *Height*, and *id* were dropped to prevent redundancy and data leakage. This computation was automated using a custom transformer integrated into the pipeline.

4.2 Encoding and Scaling

A preprocessing pipeline was constructed using `ColumnTransformer` to handle heterogeneous data:

- **Binary/Ordinal Features:** Categorical features such as Gender, SMOKE, FAVC, FCVC were encoded using `OrdinalEncoder` preserving inherent order.
- **Nominal Features:** MTRANS (mode of transportation) was encoded using `OneHotEncoder` with `drop='first'`.
- **Numerical Features:** Continuous variables like Age, NCP, BMI were standardized using `StandardScaler`.

4.3 Pipeline Integration

The preprocessing steps were integrated into a complete machine learning pipeline, ensuring consistency across training and test datasets. This design supports reproducibility and prevents data leakage during evaluation.

This robust pipeline guarantees that all steps, including the custom BMI feature calculation and the `StandardScaler` application, are fitted *only* on the training data and then applied to the test data. This methodology prevents the test set's statistics from influencing the training phase. By managing the entire workflow via the pipeline construct, the final optimized model (XGBoost) can be reliably saved and deployed as a single unit, ready to classify individuals into distinct weight categories based on their structured inputs for real-world applications.

Chapter 5

Model Development

5.1 Overview

Four models were developed and evaluated sequentially:

1. Model 1: Random Forest Classifier
2. Model 2: Base XGBoost Classifier
3. Model 3: Tuned XGBoost Classifier via GridSearchCV
4. Model 4: Tuned XGBoost

5.2 Model 1: Random Forest Classifier

- **Configuration:** n_estimators=200, random_state=42
- **Performance:** Accuracy = 88.53%

5.3 Model 2: Base XGBoost Classifier

- **Configuration:** n_estimators=100, max_depth=6, learning_rate=0.1
- **Performance:** Accuracy = 87%

5.4 Model 3: Tuned XGBoost via Grid Search

- **Methodology:** GridSearchCV with 5-fold cross-validation
- **Optimized Parameters:**
  - max_depth: [4, 6, 8]

  - learning_rate: [0.01, 0.05, 0.1]
  - n_estimators: [100, 200]
  - subsample: [0.7, 0.9, 1.0]
  - colsample_bytree: [0.7, 1.0]

- **Performance:** Accuracy = 88.74%

5.5 Model 4: Tuned XGBoost

- **Algorithm:** XGBClassifier
- **Preprocessing:** OneHotEncoder for all categorical features and StandardScaler for numerical features (including BMI)
- **Tuning Method:** GridSearchCV with 5-fold cross-validation
- **Best Parameters Found:** {learning_rate=0.1, max_depth=5, n_estimators=200, subsample=1.0}
- **Performance (Best CV Accuracy):** 91.074%
- **Key Feature Importances:** BMI (27.1%), Gender_Female (25.9%), Weight (10.1%)

Project Repository Links:

- GitHub Repository – Dharani's Project
- GitHub Repository – Abhishek's Project

Chapter 6

Results and Performance Evaluation

Table 6.1: Model Performance Comparison

Model	**Algorithm**	**Accuracy (%)**
Model 1	Random Forest	88.53
Model 2	Base XGBoost	87.00
Model 3	Tuned XGBoost	88.74
Model 4	Tuned XGBoost	91.074

Chapter 7

Discussion

The tuned XGBoost models demonstrated superior accuracy due to optimized hyperparameters controlling learning rate, tree depth, and regularization. Random Forest provided a strong baseline with stable generalization, while the base XGBoost showed faster convergence but moderate overfitting.

Feature importance analysis revealed that BMI, Age, and food consumption frequency (FCVC) were key predictors for Models 1–3. For Model 4, BMI, Gender_Female, and Weight were the most influential features, highlighting the role of consistent OneHotEncoding for categorical features along with BMI engineering.

- **Random Forest vs Base XGBoost:** Random Forest (88.53%) outperformed base XGBoost (87%), showing its robustness as a baseline.
- **Effectiveness of Hyperparameter Tuning:** Optimized XGBoost (88.74%) improved over the base model, demonstrating the significance of tuning for bias-variance tradeoff.
- **Impact of Preprocessing (Model 4):** Model 4 achieved the highest accuracy (91.074%), illustrating that feature engineering and consistent preprocessing enhances predictive performance.