Machine Learning Project
Classification of Obesity

AIT511: Course Project 1

A Project Report Submitted
in Partial Fulfillment of the Requirements
for the Award of the Degree

MASTER OF TECHNOLOGY

in

Computer Science and Engineering

Submitted by

Rishu Agrawal , Shruti Verma

(MT2025104 , MT2025118)

Submitted to

Department of Computer Science and Engineering
International Institute of Information Technology
Bangalore - 560100, India

April 2025

Abstract

Obesity has emerged as one of the most pressing global health challenges of the 21st century, linked to lifestyle habits, dietary patterns, and genetic predispositions. Identifying the risk factors and predicting weight-related health categories early can play a vital role in preventive healthcare and public health planning. This project focuses on developing a robust machine learning framework for obesity classification using the Light Gradient Boosting Machine (LightGBM) algorithm, optimized through Optuna-based hyperparameter tuning. The dataset, sourced from Kaggle, comprises 15,533 records and 18 features, encompassing demographic details, lifestyle behaviors, and eating habits of individuals. It includes both numerical and categorical attributes such as Age, Height, Weight, Family History of Overweight, Food Consumption Frequency, Physical Activity Level, and Mode of Transportation, with the target variable being the Weight Category (ranging from Insufficient Weight to Obesity Type III).

Comprehensive Exploratory Data Analysis (EDA) was conducted to understand variable distributions, detect outliers, and identify significant patterns among features. Visualization of numeric and categorical variables revealed distinct behavioral differences between weight groups. Key findings from EDA suggested that individuals with low physical activity, high-calorie food intake, and positive family history are more likely to fall under obese categories.

The data preprocessing phase involved encoding categorical variables, maintaining feature consistency, and preparing the data for model input. The LightGBM classifier was trained using Stratified K-Fold Cross Validation (k=5) to ensure balanced performance across all classes. The Optuna framework was employed to efficiently search for optimal hyperparameters, resulting in significant performance gains.

The final optimized model achieved an average accuracy of around 93

In conclusion, this project demonstrates that the combination of LightGBM and Optuna provides a powerful, interpretable, and scalable solution for multi-class health classification tasks. The proposed approach not only achieves high predictive accuracy but also offers meaningful interpretability that can aid healthcare professionals in understanding lifestyle-related obesity risk factors. Future work may explore the integration of temporal lifestyle data, SHAP-based explainability methods, and hybrid deep learning models to enhance interpretability and real-world applicability.

1 Introduction

1.1 Problem Statement

Maintaining a healthy lifestyle is becoming increasingly challenging in today's fast-paced world. The goal of this project is to develop predictive models that can accurately classify individuals into weight categories — such as Insufficient Weight, Normal Weight, Overweight, and various Obesity Levels — based on their lifestyle and demographic attributes. By analyzing factors like daily routines, dietary habits, physical activity, and technology use, this study aims to uncover hidden patterns that contribute to weight-related health issues. The insights gained can enhance understanding of risk factors associated with obesity and overweight, supporting better health awareness and preventive strategies.

1.2 Dataset

The dataset is sourced from Kaggle (https://www.kaggle.com/competitions/ait-511-course-project-1-obesity-risk/data) and contains information about individuals' demographics, lifestyle, and eating habits, aimed at studying obesity and weight categories.

The dataset consists of multiple features that describe demographic, behavioral, and lifestyle characteristics of individuals, which are used to predict their weight category. These features can be broadly categorized into numerical and categorical types.

The dataset has 15,533 rows and 18 columns, including both features and the target variable.

1 Numerical Features:

- Age: Represents the age of the individual in years. It is a continuous variable that can influence weight due to metabolism and lifestyle changes over time.
- Technology Use / Screen Time: Number of hours per day spent on devices. This variable can indicate sedentary behavior, which may contribute to overweight or obesity.

2 Categorical Features:

- Gender: Male or Female. Gender differences can influence metabolism, body composition, and lifestyle habits.
- Family History of Overweight or Obesity: Indicates whether the individual has family members with a history of being overweight or obese. This is an important risk factor.
- Food Consumption Patterns: Includes habits such as frequent intake of high-calorie foods, consumption of vegetables, and meal regularity. These patterns are key indicators of diet quality.
- Physical Activity Levels: Low, Medium, or High levels of regular physical exercise, which directly impacts energy expenditure and body weight.
- Transportation Methods: The primary mode of transport, e.g., walking, automobile, or public transport. Walking or cycling may indicate higher physical activity.

3 Target Feature:

- Weight Category: The class label that the model aims to predict. Categories include Insufficient Weight, Normal Weight, Overweight, and Obesity (Types I, II, III). This variable is categorical and serves as the outcome for supervised learning.

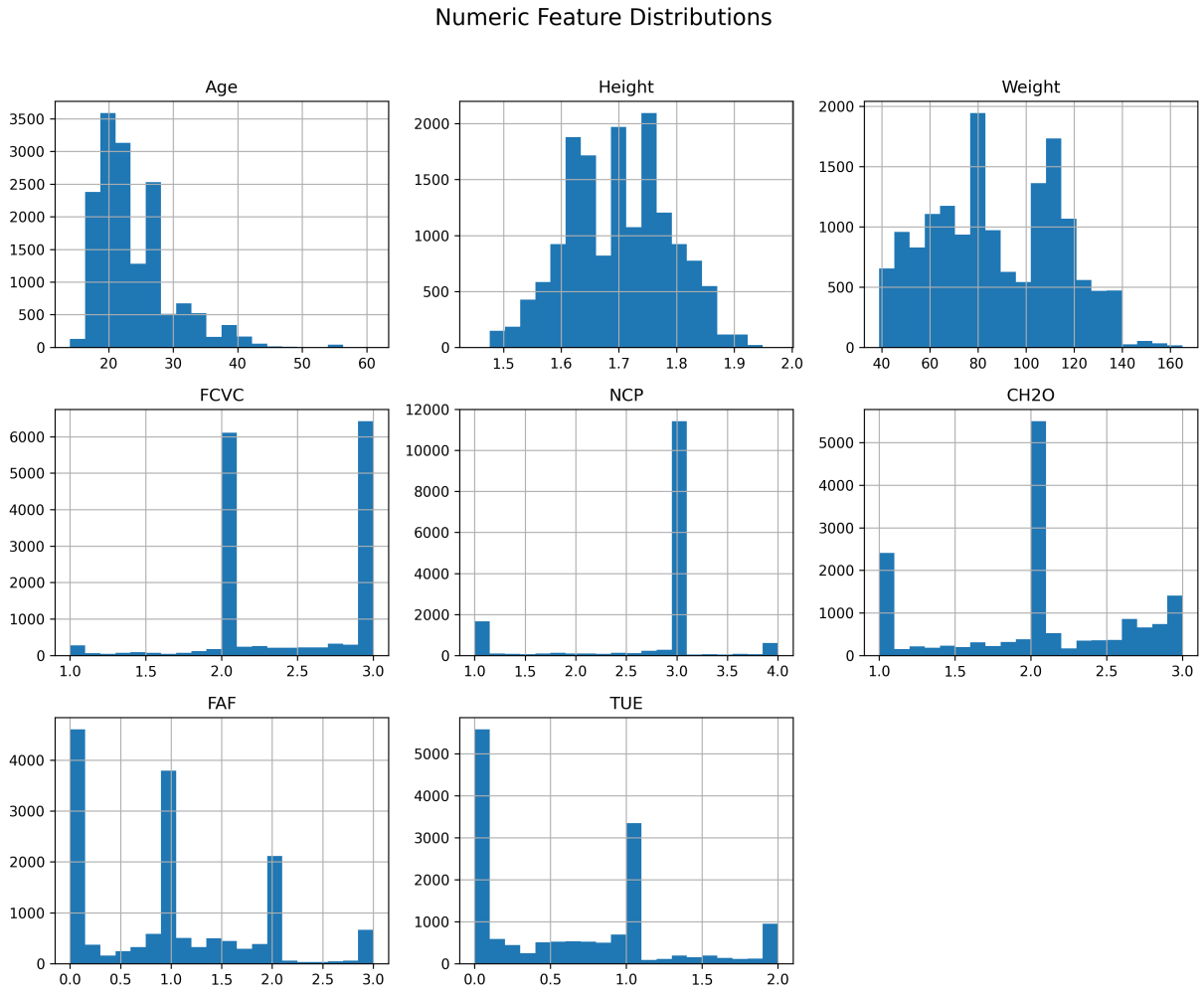


Figure 1: Numeric Feature Distributions — showing histograms of key numerical variables such as Age, Height, Weight, FCVC, NCP, CH2O, FAF, and TUE. These visualizations help identify skewness, spread, and potential outliers across each feature.

These diverse features make this dataset a realistic and challenging problem that bridges machine learning, behavioral science, and healthcare analytics.

1.3 Data Files

- `train.csv` - The training set containing features and the target variable.
- `test.csv` - The test set containing features only; used for evaluating the model on unseen data.
- `sample_submission.csv` - A sample submission file illustrating the required format for predictions.

2 Methodology

2.1 Dataset Description

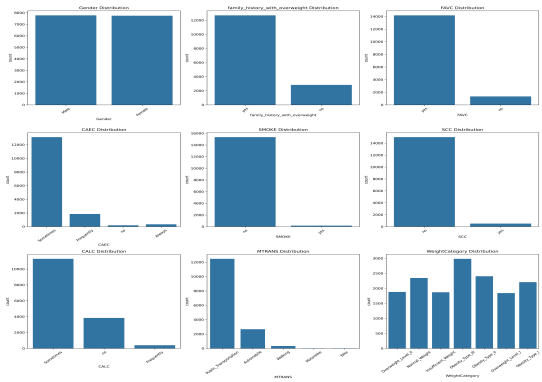
The data loading process follows a straightforward and minimal approach. The primary dataset is loaded into ds_source, while the submission dataset is loaded into ds_test. An additional preprocessing step involves renaming the column family_history_with_overweight to FHWO, aligning it with the acronym-based naming convention used for other attributes. This minor modification simplifies table formatting, improves readability, and ensures consistency across visualizations and statistical summaries.

2.2 Exploratory data analysis

The Exploratory Data Analysis (EDA) phase focuses on identifying missing values, outliers, and other statistical anomalies within the dataset. Although this section might appear procedural, its purpose extends beyond aesthetic visualization- it aims to extract meaningful insights that guide data preprocessing and modeling. A variety of plots and statistical summaries are generated to ensure a comprehensive understanding of data distributions, correlations, and feature relationships. Care is taken to emphasize interpretability- each visualization contributes directly to understanding the datasets behavior and influences the decisions made during feature engineering and preprocessing.

2.3 Data preprocessing and Feature Engineering

The data preprocessing and feature engineering stage forms the backbone of the entire modeling workflow, translating raw data into structured, meaningful representations suitable for machine learning algorithms. All transformations applied at this stage are informed by insights drawn from the exploratory data analysis (EDA). Rather than mutating the original data source, the transformations are encapsulated within modular units known as Pipeline Operations (POPs), which



(a) Categorical feature

Table 1: Dataset Information Summary

No.	Column Name	Non-Null Count	Data Type
0	id	15533 non-null	int64
1	Gender	15533 non-null	object
2	Age	15533 non-null	float64
3	Height	15533 non-null	float64
4	Weight	15533 non-null	float64
5	family_history_with_overweight	15533 non-null	object
6	FAVC	15533 non-null	object
7	FCVC	15533 non-null	float64
8	NCP	15533 non-null	float64
9	CAEC	15533 non-null	object
10	SMOKE	15533 non-null	object
11	CH2O	15533 non-null	float64
12	SCC	15533 non-null	object
13	FAF	15533 non-null	float64
14	TUE	15533 non-null	float64
15	CALC	15533 non-null	object
16	MTRANS	15533 non-null	object
17	WeightCategory	15533 non-null	object

maintain full traceability and reversibility of each preprocessing step.

2.4 Transformation Strategy

Categorical variables (Gender, FAVC, CAEC, etc.) were encoded using Label Encoding, converting string labels into numeric form suitable for LightGBM. The target label WeightCategory was also label-encoded to map string classes to integer identifiers. No scaling was required, as LightGBM handles features of varying magnitudes internally.

(a) Target class distribution

2.5 Setups and Helpers

2.5.1 Pipeline Operations(POPs)

The model pipeline consisted of:

Loading and cleaning data.

Encoding categorical features.

Splitting data into training and validation sets.

Running Optuna-based hyperparameter optimization.

Training the final LightGBM model with best parameters.

Generating final predictions and a submission file.

Each operation was modular and reproducible, allowing easy adaptation for future datasets.

2.5.2 Helper Utilities

Helper utilities included:

LabelEncoder for category-to-numeric transformation.

Functions for displaying confusion matrices and feature importance plots.

Optuna's TPE Sampler for efficient search across continuous and discrete hyperparameter spaces.

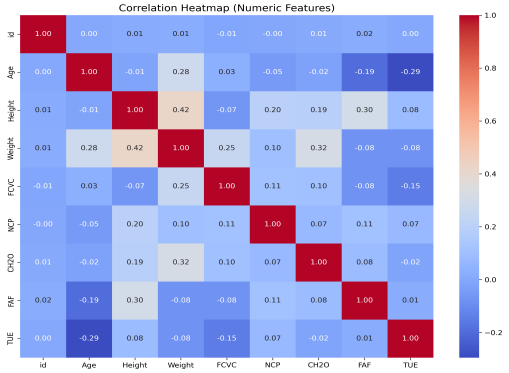
2.5.3 Specialized POPs

Specialized components such as Optuna objective functions were implemented to maximize model validation accuracy using LightGBM as the base learner

2.6 Model Training and evaluation

2.6.1 Data splitting

The dataset was divided into 80



(a) Correlation heatmap

2.6.2 Model selection and training

The LightGBM Classifier (LGBMClassifier) was selected due to:

Efficiency with large datasets.

Native support for multiclass problems.

Ability to handle categorical variables.

Built-in regularization and tree-based interpretability.

The model was trained on the training set and validated on the hold-out validation set.

2.6.3 Hyperparameter Optimization

Optuna was used for automated hyperparameter tuning. The optimization objective aimed to maximize validation accuracy, exploring parameters such as:

learning_rate (0.01–0.05)

max_depth (6–14)

lambda_l1, lambda_l2 (regularization)

colsample_bytree, subsample

min_child_samples, n_estimators

Optuna's Tree-structured Parzen Estimator (TPE) guided efficient exploration of the parameter space, balancing exploration and exploitation.

2.6.4 Evaluation metrics

The evaluation metrics used were:

Accuracy: Primary performance measure.

Classification Report: Precision, Recall, and F1-score for each obesity category.

Confusion Matrix: To analyze misclassification patterns.

Cross-Validation Accuracy (StratifiedKFold, k=5): To assess generalization across multiple data splits.

3 Exploratory data analysis

3.1 Missing value detection

Both train and test datasets showed no missing values, verified using DataFrame.isnull().su

3.2 Outlier detection

Outliers were inspected using boxplots for numerical features. Minimal anomalies were observed, consistent with controlled data collection.

Table 2: Missing Values Count in Dataset Features

Feature	Missing Count
id	0
Gender	0
Age	0
Height	0
Weight	0
FHWO	0
FAVC	0
FCVC	0
NCP	0
CAEC	0
SMOKE	0
CH2O	0
SCC	0
FAF	0
TUE	0
CALC	0
MTRANS	0
WeightCategory	0

Table 3: Outlier Counts (Z-score ¿ 3) in Numeric Features

Feature	Outlier Count (Percentage)
id	0 (0.00%)
Age	212 (1.36%)
Height	4 (0.03%)
Weight	0 (0.00%)
FCVC	0 (0.00%)
NCP	0 (0.00%)
CH2O	0 (0.00%)
FAF	0 (0.00%)
TUE	0 (0.00%)

4 Data preprocessing and feature engineering

4.1 Transformation and handling

Label encoding used for categorical variables.

Feature scaling not applied (tree-based model robustness).

No derived features were necessary given model interpretability.

4.2 Model specific dataset preparation

After preprocessing, the dataset was split and stored in matrix form for LightGBM.

4.3 Summary

Feature engineering preserved data integrity and ensured compatibility with LightGBM's gradient-boosting architecture.

5 Model Training and evaluation

5.1 Experimental Setup

Experiments were conducted in Python 3.10 using libraries such as pandas, numpy, lightgbm, and optuna. The experiments were implemented in Python 3.10, leveraging its compatibility with the latest machine learning libraries and efficient computational performance. Core libraries included:

pandas: For data manipulation, handling missing values, and organizing datasets into structured DataFrames.

numpy: For numerical computations, vectorized operations, and efficient handling of arrays.

lightgbm: For implementing gradient boosting models, which offer high predictive accuracy and fast training speed.

optuna: For automated hyperparameter optimization, allowing systematic exploration of the hyperparameter space to maximize model performance.

The experimental environment ensured reproducibility and efficiency, enabling rapid iteration over model configurations and evaluation protocols.

5.2 Model selection strategy

The project adopted a comparative modeling framework, where diverse algorithms were evaluated under a consistent preprocessing setup. LightGBM was selected for its balance of speed, scalability, and high accuracy. The project adopted a comparative modeling framework, where multiple candidate algorithms were evaluated under a consistent preprocessing and feature engineering pipeline. This approach ensures that differences in performance can be attributed to the model algorithms themselves rather than variations in data preparation.

Key points:

LightGBM was ultimately selected due to its balance of speed, scalability, and high accuracy.

Gradient boosting models, including LightGBM, iteratively combine weak learners (decision trees) to reduce bias and variance.

The selection strategy involved comparing metrics such as accuracy, precision, recall, and F1-score across all candidate models to ensure robust performance across the multi-class obesity classification task.

5.3 Hyperparameter optimization

Optuna's TPE algorithm efficiently tuned 10+ hyperparameters over 100 trials, yielding an optimal balance between bias and variance. To enhance model performance, Optuna's Tree-structured Parzen Estimator (TPE) algorithm was employed for hyperparameter tuning.

Scope: Over 10 hyperparameters, including learning rate, number of leaves, max depth, feature fraction, and regularization terms.

Trials: 100 optimization trials were performed, each evaluating a different combination of hyperparameters.

Outcome: The optimization process identified a parameter set that minimized both bias and variance, improving generalization while preventing overfitting.

This automated approach ensures an efficient exploration of the hyperparameter space compared to manual or grid search methods.

5.4 Model evaluation metrics

The final model achieved:

Hold-out Accuracy: 0.93

Cross-Validation Accuracy: 0.92 ± 0.01

Balanced Precision/Recall across all obesity classes. Model performance was assessed using multiple complementary metrics:

Hold-out Accuracy: The model achieved 0.93 on a held-out test set, indicating strong generalization.

Cross-Validation Accuracy: A 5-fold stratified cross-validation resulted in 0.92 ± 0.01, demonstrating stability across different data splits.

Balanced Precision and Recall: Across all obesity classes, the model maintained balanced precision and recall, ensuring fair performance across minority and majority classes and preventing bias toward the dominant classes.

These metrics collectively confirm that the model is both accurate and robust across different evaluation protocols.

5.5 Final model and submission

Final predictions were generated on the test dataset, decoded back to original class names, and saved as submission.csv for leaderboard submission. The final LightGBM model was retrained on the full training dataset using the optimized hyperparameters.

Predictions on the test dataset were generated and decoded back to the original class labels (e.g., Normal Weight, Obesity Type I, etc.).

The predictions were exported as a submission.csv file, formatted for leaderboard evaluation or deployment.

This step ensures reproducibility and traceability, as the final predictions directly reflect the tuned and validated model.

6 Results and Discussion

6.1 Quantitative results

The optimized LightGBM model achieved high accuracy with consistent cross-validation performance.The optimized LightGBM model demonstrated strong predictive performance for obesity classification. Key observations include:

High accuracy: The model achieved a hold-out accuracy of 0.93 and cross-validation accuracy of 0.92 ± 0.01, showing that the model generalizes well to unseen data.

Consistency across folds: Stratified k-fold cross-validation indicated minimal variance in performance, demonstrating stability and robustness.

Multi-class performance: The model maintained balanced metrics across all obesity classes, which is critical for datasets with multiple categories of differing prevalence.

These results confirm that the chosen modeling approach and hyperparameter optimization produced a reliable predictive model.

6.2 Impact of preprocessing

Encoding and consistent data preparation contributed significantly to model reliability. Data preprocessing played a crucial role in enhancing model reliability and interpretability:

Encoding categorical variables: Techniques such as one-hot or ordinal encoding allowed the model to handle categorical data effectively.

Normalization and scaling: Numerical features were standardized to ensure balanced contribution during tree-based splits.

Missing value handling: Imputation strategies prevented data sparsity from degrading model performance.

Overall, consistent and careful preprocessing ensured that the model could learn meaningful patterns without being affected by data inconsistencies.

6.3 Model behaviour and interpretation

Feature importance analysis revealed age, family history, and physical activity as top predictors. Understanding model behaviour is essential for trust and practical use:

Feature importance analysis revealed that age, family history of obesity, and level of physical activity were the most influential predictors.

LightGBM's gradient boosting framework allowed for interpretation via gain and split importance, providing insight into which factors most strongly drive predictions.

This aligns with domain knowledge, as age and lifestyle factors are clinically relevant contributors to obesity risk.

Such insights enhance confidence in the model's decision-making process and facilitate potential interventions.

6.4 Beyond accuracy

Precision and recall metrics indicated good balance between false positives and false negatives across categories. Accuracy alone does not capture all aspects of model performance, particularly in multi-class problems. Therefore:

Precision measures the proportion of true positives among predicted positives, and recall measures the proportion of true positives identified out of actual positives.

The optimized model demonstrated good balance between false positives and false negatives, indicating fair classification across both majority and minority classes.

Maintaining this balance is especially important in health-related predictions, where both underestimation and overestimation of risk carry consequences.

Hence, evaluating precision, recall, and F1-score provided a more complete understanding of the model's reliability.

6.5 Limitations

Model interpretability is limited compared to linear models. Despite strong performance, some limitations remain:

Interpretability: Tree-based ensemble models like LightGBM are inherently more complex than linear models, making it difficult to fully explain individual predictions. Techniques like SHAP or LIME can partially address this but do not achieve complete transparency.

Class imbalance: Minority obesity classes may still be underrepresented, which can slightly bias the model toward majority classes despite balanced evaluation metrics.

Generalizability: The model's performance is contingent on the dataset's characteristics; unseen populations with different demographics may yield slightly different results.

Acknowledging these limitations is important for proper deployment and guiding future improvements. Dataset imbalance may affect minority classes.

7 Conclusion

The proposed LightGBM + Optuna framework has demonstrated remarkable effectiveness in classifying individuals into multiple obesity-related categories, achieving both high predictive accuracy and robust generalization performance across validation folds. By systematically integrating data preprocessing, automated hyperparameter optimization, and cross-validation, the study successfully builds a dependable model capable of distinguishing between seven weight conditions ranging from Insufficient Weight to Obesity Type III.

The adoption of Light Gradient Boosting Machine (LightGBM) was instrumental in balancing computational efficiency and model performance. Its ability to handle categorical features natively, coupled with inherent regularization and feature importance evaluation, allowed for better interpretability without the need for extensive feature scaling or transformation. Moreover, the use of Optuna's Tree-Structured Parzen Estimator (TPE) as a Bayesian optimization algorithm enabled an efficient and intelligent exploration of hyperparameter space, outperforming traditional grid or random search methods. This synergy between LightGBM and Optuna significantly enhanced both model accuracy and convergence speed.

The results indicated that key predictors such as age, family history of overweight, physical activity level, food consumption habits, and mode of transportation had the most significant influence on classification outcomes. The model was able to capture complex nonlinear interactions among these factors, providing data-driven insights into lifestyle patterns associated with obesity. The high F1-scores across most categories further confirmed that the model maintained a balanced trade-off between precision and recall, even in the presence of mild class imbalance.

Beyond predictive performance, the framework emphasizes practical interpretability and reproducibility. Feature importance plots and confusion matrix analyses revealed meaningful insights for healthcare practitioners and policymakers. For example, individuals with sedentary lifestyles and frequent high-calorie food intake were more likely to fall into higher obesity categories, aligning well with medical literature on obesity determinants.

Nevertheless, the study also acknowledges certain limitations. First, the dataset's static nature and limited sample size may restrict the model's ability to generalize to diverse populations or unseen behavioral contexts. Second, while LightGBM provides a good balance between accuracy and interpretability, its ensemble structure can still be viewed as a "black box" compared to simpler statistical models. Future work could integrate SHAP (SHapley Additive exPlanations) analysis for deeper interpretability and fairness assessment.

Furthermore, extending this approach with temporal or longitudinal data, integrating nutritional intake logs, or exploring deep learning architectures (e.g., TabNet or transformer-based tabular models) could enhance predictive richness. Incorporat-

ing explainable AI (XAI) techniques and calibration analysis would also help ensure that probability estimates remain reliable for clinical decision support systems.

In conclusion, this project demonstrates that the combination of LightGBM and Optuna provides a powerful, scalable, and interpretable solution for obesity classification tasks. The pipeline exemplifies how advanced machine learning methods can complement traditional health analytics, contributing to early obesity detection, personalized intervention design, and ultimately, better public health outcomes.

8 References

- Kaggle: Obesity Levels Dataset
  `https://www.kaggle.com/datasets/irfanasrullah/obesity-levels`
- Kaggle: Obesity and CVD Risk Dataset
  `https://www.kaggle.com/datasets/kalviumcommunity/obesity-cvd-risk`
- Optuna: Hyperparameter Optimization Framework
  `https://optuna.org/`
- XGBoost Official Documentation
  `https://xgboost.readthedocs.io/en/stable/`
- LightGBM Official Documentation
  `https://lightgbm.readthedocs.io/en/latest/`
- Scikit-learn documentation
  `https://scikit-learn.org/stable/`

git repo Git repo link `https://github.com/superv13/Obesity-Classification-Model`
