Hertie School

Obesity Prediction

The Scale Doesn't Lie — But Does Our Model?

Final Report
Supervised Machine Learning · Spring 2025
Hertie School · MDS

Authors
Nadine Daum · Ashley Razo · Jasmin Mehnert · Nicolas Reichardt

GitHub: https://github.com/nicolasreichardt/ml-project-obesity-prediction
Submission: 12 May 2025

Summary

This project applies supervised machine learning to classify individuals into obesity risk categories based on biometric and lifestyle data. We implemented and evaluated multiple models — including logistic regression, KNN, tree-based models, and a neural network — using a shared preprocessed dataset to ensure consistent and fair comparison.

Our best-performing models achieved test accuracy scores above 85%, with interpretable insights from tree-based approaches and strong generalization from the neural network.

Table of Contents

Team

- Nadine Daum – GitHub | Email
- Jasmin Mehnert – GitHub | Email
- Ashley Razo – GitHub | Email
- Nicolas Reichardt – GitHub | Email

Project Overview

This project aims to classify individuals into seven obesity risk categories based on various biometric and behavioral factors. Using a labeled dataset of 2,111 individuals from Mexico, Peru, and Colombia, our models predict obesity levels ranging from *Insufficient Weight* to *Obesity Type III*.

The goal is to explore how well machine learning models can predict obesity status — and how these predictions might support future public health decisions, risk assessment tools, or individual recommendations.

GitHub repo: nicolasreichardt/ml-project-obesity-prediction

1. Dataset Description

We used the **Obesity Levels Estimation Dataset**, which contains demographic, behavioral, and biometric data for 2,111 individuals from Mexico, Peru, and Colombia. The dataset was designed for multi-class classification and is labeled with 7 obesity categories.

Dataset Overview:

- **Size**: 2,111 samples × 17 features + 1 target
- **Features**: mix of categorical (e.g., gender, transport_mode) and numerical (e.g., height, weight, age)
- **Target variable**: `obesity_level` with 7 classes:
  - Insufficient Weight
  - Normal Weight
  - Overweight Level I
  - Overweight Level II
  - Obesity Type I
  - Obesity Type II
  - Obesity Type III
- **ML relevance**: Multi-class classification task with imbalanced class distribution
- **Input shape for models**: ~43 features after encoding (based on one-hot transformation)

The data was collected via a cross-sectional survey and is publicly available on Kaggle, supported by this research article.

> 📝 **Jasmin – please add 1–2 sentences here about EDA findings**
> For example: were there correlations, outliers, imbalances, or interesting clusters?

All team members used a shared train/test split to ensure model comparability.

2. Preprocessing & Feature Engineering

Before modeling, the dataset required thorough cleaning and transformation. This step was led primarily by **Ashley Razo** and **Jasmin Mehnert**, with feedback and reviews from all team members.

Preprocessing Goals

- Ensure consistent input format across models
- Improve model performance and comparability
- Reduce noise, redundancy, and scaling-related bias

Key Steps

- **Feature selection**: Retained 17 relevant input features capturing diet, behavior, and biometrics
- **Target formatting**: Standardized and renamed the class column to `obesity_level`
- **Encoding**: Applied one-hot encoding to 13 categorical features (e.g., `gender`, `transport_mode`)
- **Scaling**: Used `StandardScaler` to normalize all numerical features (e.g., `age`, `height_m`, `weight_kg`)
- **Output dimensions**: Final input to the models included ~43 encoded features
- **Train/test split**: 80/20 split applied uniformly to ensure fair model evaluation
- **File formats**: Datasets exported as both `.csv` and `.feather` (for faster access)

> 📝 **@Ashley** – feel free to insert 1–2 sentences on your preprocessing pipeline: decisions around feature selection, encoding strategies, or challenges during cleaning
> 📝 **@Jasmin** – you can briefly note how you supported the pipeline and flag any edge cases or quirks in the data

Implementation

📒 Notebook: `notebooks/preprocessing.ipynb`
🧾 Script: `processed_data/data_preparation.py`

All models consumed the same cleaned and scaled training and testing data.

3. Model Overviews

All models used the same preprocessed data for consistency.

Logistic Regression

📒 logistic_regression.ipynb

- Simple baseline with good interpretability

Ridge Logistic Regression

📒 ridge_logistic_regression.ipynb

- Regularized version of logistic regression

K-Nearest Neighbors (KNN)

📒 PCA_KNN.ipynb

- PCA helped reduce dimensionality and improved KNN performance

Neural Network

📒 neural_network.ipynb

- Multi-layer architecture with ReLU and softmax
- Test accuracy: **83.9%**
- Balanced performance across all obesity categories

Neural Network Training Curves

Tree-Based Models

📒 tree-based-models.ipynb

- Random Forest & XGBoost achieved top performance (~86%)
- Screen time, calorie tracking, and water intake were key features

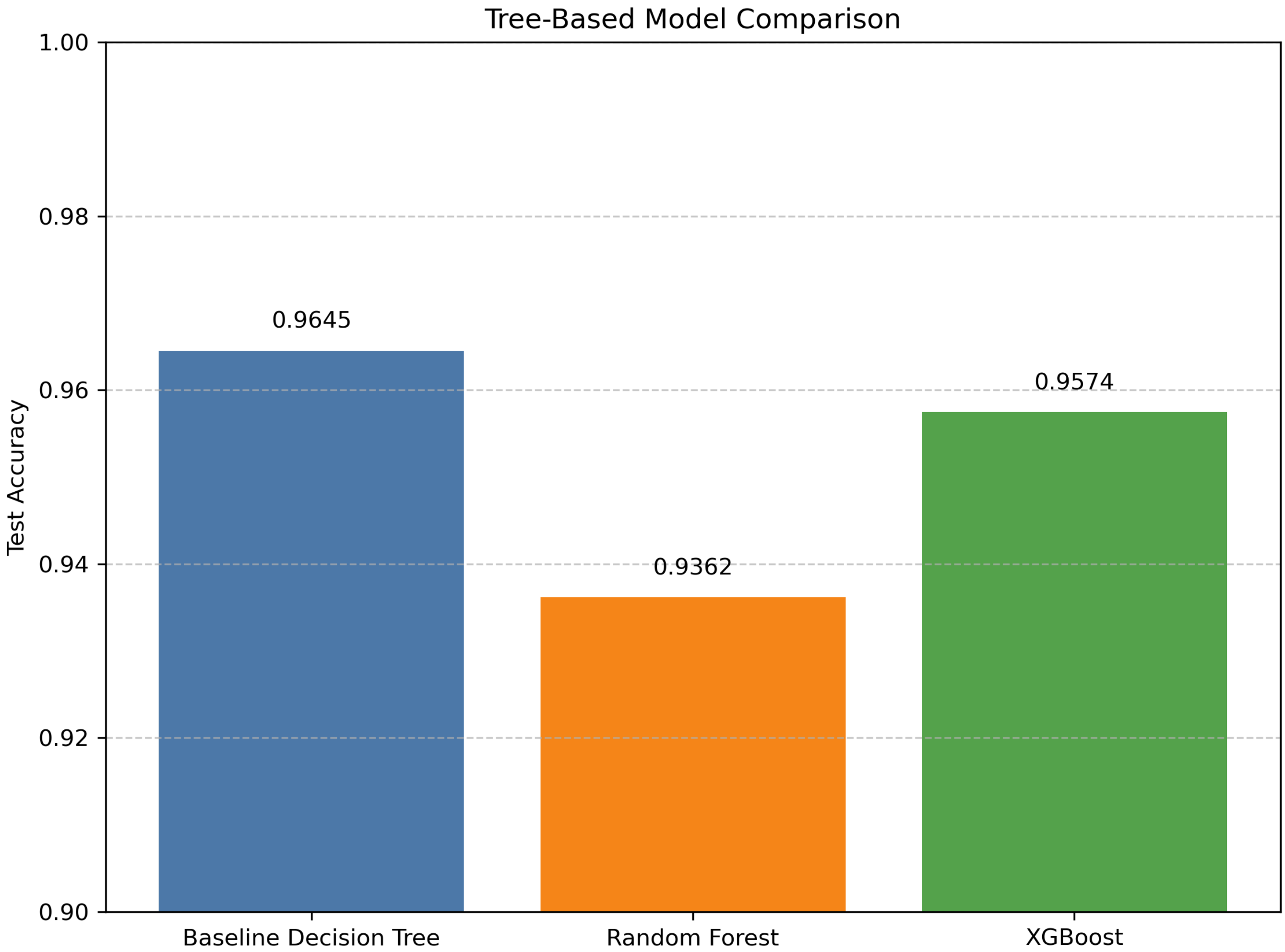
4. Model Comparison

Model	Test Accuracy	Notes
Logistic Regression	~75%	Simple, interpretable
Ridge Logistic Regression	~76%	Slight improvement with regularization
KNN	~77%	Better with PCA
Neural Network	**83.9%**	Strong generalization
Random Forest	~85–86%	Robust, interpretable
XGBoost	~86%	Top performer with best generalization

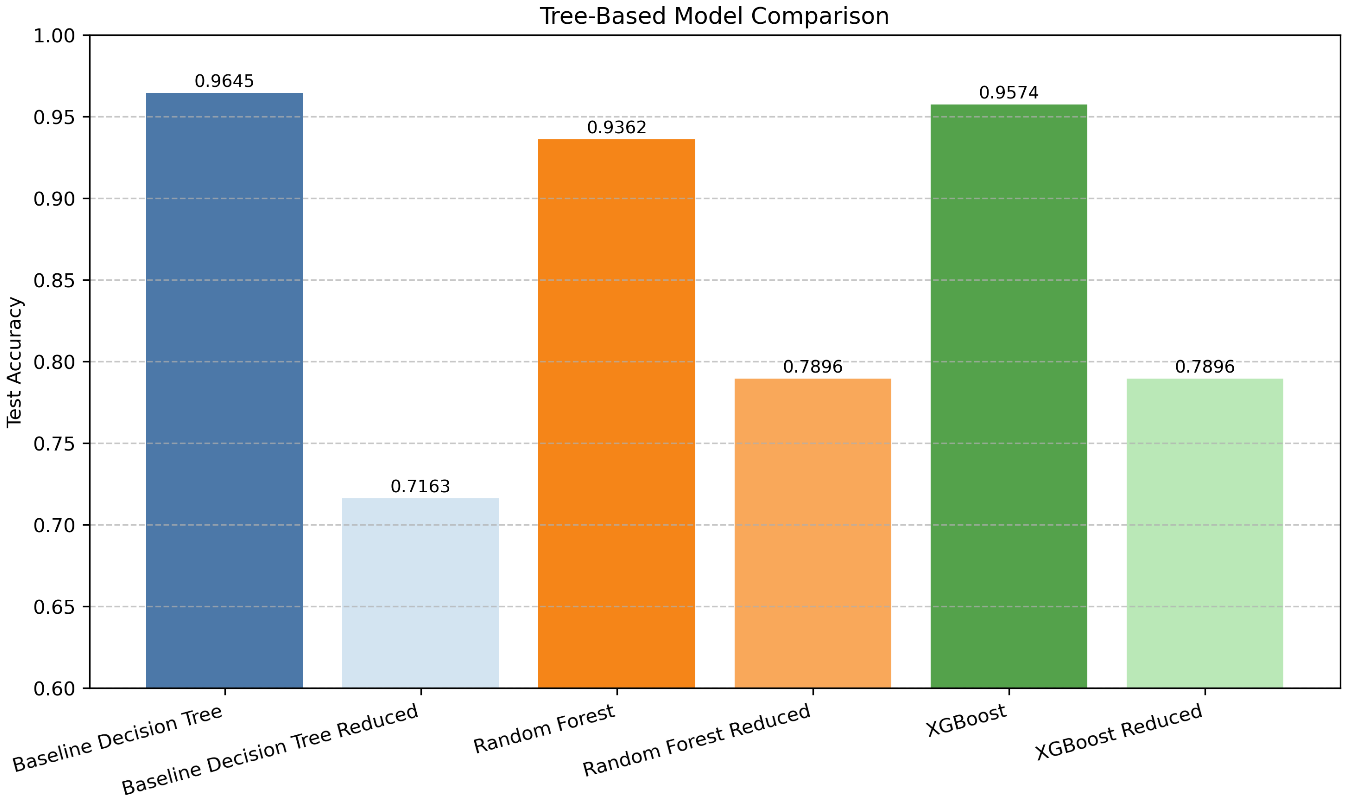
Feature Importance – Tree-Based Models

Feature_DT	Importance_DT	Feature_RF	Importance_RF	Feature_XGB	Importance_XGB
weight_kg	0.6249	weight_kg	0.3335	gender_Female	0.385
height_m	0.1965	age	0.114	weight_kg	0.1489
gender_Male	0.1208	height_m	0.1105	high_caloric_food_freq	0.0805
age	0.0237	gender_Male	0.0489	alcohol_consumption_freq	0.0659
high_caloric_food_freq	0.01	vegetables_freq	0.0424	snacking_freq	0.0475

Model Comparison Overview



Model Comparison with Feature Exclusion



5. Reflections

- Preprocessing made a big difference across all models
- Tree-based models helped us understand what mattered most
- Neural networks were surprisingly manageable and performed well
- Sharing the same train/test split helped standardize evaluation
- We improved our understanding of ML pipelines, GitHub collaboration, and reproducibility

Appendix A: Links & Files

- **GitHub Repository**: nicolasreichardt/ml-project-obesity-prediction
- **Cleaned dataset (CSV)**: `processed_data/obesity_cleaned.csv`
- **Train/Test files**:
  - `processed_data/train_data.feather`
  - `processed_data/test_data.feather`
- **Model notebooks**: in `notebooks/`
- **Generated plots**: in `plots/`

Appendix B: Team Contributions

- **Nadine Daum** – Neural network, Ridge/Lasso regression
- **Ashley Razo** – Preprocessing, logistic regression
- **Jasmin Mehnert** – PCA & KNN, preprocessing support
- **Nicolas Reichardt** – Random Forest, XGBoost, evaluation
  All team members contributed to meetings, reviews, and report writing.