

Comparison of classical machine learning methods in the task of obesity level classification

Paweł Biesaga*, Paweł Powroźnik

Department of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract

Obesity is a critical public-health problem amenable to data-driven risk assessment. We compare eight classical classifiers on the UCI ObesityDataSet (2087 samples, 7 classes) under a leakage-free protocol: stratified 10-fold cross-validation, randomised-search tuning, pairwise McNemar and Wilcoxon tests with Holm–Bonferroni correction, and multi-method interpretability (split gain, SHAP, permutation, LIME). XGBoost reached 96.17% test accuracy ($F1 = 0.962$, $\kappa = 0.955$); paired tests show no significant edge over tuned LightGBM, SVM-RBF, Random Forest or Logistic Regression. All four importance measures rank body weight and height first, and LIME recovers BMI-based thresholds, supporting gradient boosting for clinical decision support.

Keywords: machine learning; obesity classification; gradient boosting; statistical significance; SHAP; LIME

*Corresponding author

Email address: biesagapawel@gmail.com (P. Biesaga)

Published under Creative Common License (CC BY 4.0 Int.)

1. Introduction

Obesity and overweight are among the most rapidly growing health problems of the 21st century. According to the World Health Organization (WHO), they affect more than one billion adults worldwide, and their consequences — cardiovascular diseases, type 2 diabetes, and musculoskeletal disorders — directly reduce both the quality and the expected length of life [1]. Accurate and early identification of the level of obesity constitutes the foundation for treatment planning and the development of personalised dietary recommendations.

In the era of progressing digitalisation of healthcare, intelligent Clinical Decision Support Systems (CDSS) implemented using machine learning (ML) methods are gaining increasing importance. Classical ML algorithms are capable of rapidly detecting complex relationships in structured tabular data, such as medical interview records that include age, body weight, height, and dietary habits [2]. The advantage of tree-based algorithms over deep learning methods on this type of data is well documented in the literature [3, 4].

Although machine learning is increasingly used for obesity-level assessment, existing studies on the ObesityDataSet rarely combine a leakage-free evaluation protocol, formal statistical testing of the differences between models, and a multi-method interpretability analysis within a single reproducible comparison. The aim of this study is therefore to provide a methodologically rigorous and fully reproducible comparison of classical machine learning algorithms for multi-class obesity-level classification, and to establish whether the performance differences commonly reported between such models are statistically meaningful or merely artefacts of a particular data split.

The main contribution of this paper is a methodologically rigorous empirical comparison of eight classical machine learning algorithms in the task of multi-class

classification of obesity levels. Specifically, the work contributes:

- a leakage-free experimental protocol with stratified 10-fold cross-validation (CV) and randomised-search hyperparameter tuning performed in an inner cross-validation loop on the training partition only;
- pairwise statistical significance testing using McNemar’s test on held-out predictions and Wilcoxon signed-rank tests over CV folds, with Holm–Bonferroni correction for the 28 pairwise comparisons;
- a multi-method interpretability analysis that combines global feature-importance measures (split gain, SHapley Additive exPlanations (SHAP) values, permutation importance) with Local Interpretable Model-agnostic Explanations (LIME) [5] explanations of representative predictions;
- explicit reporting of standard deviations across CV folds, which makes it possible to dissociate genuine performance differences from sampling variance.

The research hypothesis under verification states that ensemble gradient boosting algorithms based on decision trees (LightGBM, XGBoost) achieve significantly higher predictive performance than linear and probabilistic models on structured medical data, while maintaining acceptable training and inference times.

2. Literature review

Applications of machine learning in the classification and diagnosis of metabolic diseases have attracted growing interest. Chicco and Jurman [6] conducted a widely cited comparative study of more than a dozen classifiers on cardiological datasets, identifying Random Forest and gradient boosting as consistently the most effective methods. Similar conclusions were drawn by Fernández-Delgado et al. [7], who, in a meta-analysis covering 179 classifiers and 121 datasets, demonstrated that random forests and boosting methods dominate

over the remaining algorithms in tabular classification tasks.

In the context of obesity and overweight classification, the ObesityDataSet published by Palechor and de la Hoz Manotas [2] has become one of the most commonly used benchmarks. In the original work, the authors achieved a baseline accuracy of around 80–85% using distance-based algorithms. Subsequent studies, such as the work of Ferdowsy et al. [8], analysed this dataset using decision trees and logistic regression, obtaining results in the range of 88–93%, but without considering more recent boosting methods.

Chen and Guestrin [9] demonstrated that the sequential construction of trees in the XGBoost method yields exceptional results on tabular datasets, including those containing categorical variables. Ke et al. [10] further developed this idea in the LightGBM algorithm by introducing a leaf-wise tree growth strategy and histogram-based encoding, which substantially accelerates training while preserving high accuracy. Rokach [11] and Mienye and Sun [12] provide comprehensive reviews of ensemble methods, confirming their advantage over single classifiers through prediction-variance reduction.

Of particular relevance is the body of research on the superiority of tree-based algorithms over deep learning methods on tabular data. Grinsztajn et al. [3] showed, on 45 datasets, that tree-based models consistently dominate over neural networks when the data contain uncorrelated or irregularly distributed features. These results were independently confirmed by Shwartz-Ziv and Armon [4]. Support vector machines have gained wide application in medical classification [13], although they require careful hyperparameter tuning and data normalisation.

Despite this rich body of work, three gaps persist. First, most comparative studies on the ObesityDataSet report only raw accuracy and omit formal significance testing, leaving it unclear whether the differences between the leading models are real or attributable to sampling variability. Second, the evaluation protocols are frequently vulnerable to data leakage, particularly when hyperparameter tuning is not isolated from the test set. Third, interpretability is usually restricted to a single feature-importance method. The present study addresses these gaps by combining a leakage-free protocol, pairwise statistical testing with multiple-comparison correction, and a four-method interpretability analysis.

3. Materials and methods

3.1. Dataset

The publicly available *ObesityDataSet* [2] from the UCI Machine Learning Repository (ID 544) was used. It comprises 2111 records collected via questionnaires from inhabitants of Colombia, Peru, and Mexico. Twenty-four exact duplicate rows were removed prior to any further processing (Section 3.2), leaving 2087 unique samples. The target variable *NObeyesdad* takes seven ordered values reflecting the WHO classification of body-mass status: *Insufficient_Weight*, *Normal_Weight*,

Overweight_Level I, *Overweight_Level II*, *Obesity_Type I*, *Obesity_Type II*, and *Obesity_Type III*. The class distribution is approximately balanced (between 12.8 % and 16.8 % per class); the per-class sample counts are reported in Table 1.

Table 1: Class distribution of the ObesityDataSet (2087 unique samples).

Class	Samples (n)	Share (%)
Insufficient_Weight	267	12.8
Normal_Weight	282	13.5
Overweight_Level I	276	13.2
Overweight_Level II	290	13.9
Obesity_Type I	351	16.8
Obesity_Type II	297	14.2
Obesity_Type III	324	15.5
Total	2087	100.0

Among the 16 input features, the following groups were distinguished: 8 numerical (*Age*, *Height*, *Weight*, *FCVC*, *NCP*, *CH2O*, *FAF*, *TUE*), 5 binary (*Gender*, *family_history_with_overweight*, *FAVC*, *SMOKE*, *SCC*), 2 ordinal (*CAEC*, *CALC*, both encoded on a four-level ordinal scale from *no* to *Always*), and 1 nominal with five categories (*MTRANS*). Here, *FAVC* denotes frequent consumption of high-caloric food; *FCVC*, frequency of vegetable consumption; *NCP*, number of main meals; *CAEC*, consumption of food between meals; *CH2O*, daily water consumption; *SCC*, calories consumption monitoring; *FAF*, physical activity frequency; *TUE*, time using technology devices; *CALC*, consumption of alcohol; and *MTRANS*, the means of transportation used.

3.2. Preprocessing

Two preprocessing pipelines were used, matched to the inductive bias of the model family. For tree-based models (Decision Tree, Random Forest, XGBoost, LightGBM), label encoding of the *MTRANS* variable was used and scaling was omitted, since tree splits are invariant to monotonic transformations of features. For the remaining models (Logistic Regression, KNN, SVM, Naive Bayes), one-hot encoding was applied to *MTRANS* (yielding 20 features in total) along with standardisation of the numerical and ordinal variables using *StandardScaler*:

$$z = \frac{x - \mu}{\sigma}, \quad (1)$$

where μ and σ were determined exclusively on the training set, preventing data leakage. The cleaned dataset was split in a stratified 80/20 ratio (1669 training and 418 test samples).

3.3. Algorithms

Eight classifiers spanning six methodological families were evaluated. Key formal definitions are summarised below.

Logistic regression [14] models class probabilities through the sigmoid of a linear combination of features, generalised by softmax to the seven-class setting and trained by cross-entropy minimisation:

$$\sigma(z) = \frac{1}{1 + e^{-z}}. \quad (2)$$

The decision surface is linear in the original feature space.

Gaussian Naive Bayes [15] applies Bayes' rule

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \quad (3)$$

under conditional independence of features, modelling each $P(X_j|C)$ as a univariate normal. The independence assumption is clearly violated for our questionnaire data (Weight and Height are strongly correlated), which explains the markedly lower accuracy observed in Section 4.1.

k-nearest neighbours (KNN) [16] is a lazy learner whose training stage reduces to memorising the training set. A new sample is labelled by the (distance-weighted) majority vote of its k closest neighbours under the Euclidean metric

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}. \quad (4)$$

Feature standardisation is required because the distance is dominated by variables of large numerical range.

Support vector machine (SVM) [17] finds the widest-margin separating hyperplane and lifts the data into a higher-dimensional space via the Gaussian radial basis function (RBF) kernel

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad (5)$$

where γ controls the radius of influence of each support vector. The seven classes are decomposed using the one-versus-one strategy (21 binary classifiers, majority vote).

Decision tree [18] recursively partitions the feature space; at each node the variable and threshold are chosen so as to minimise the Gini impurity

$$G = 1 - \sum_{i=1}^C p_i^2, \quad (6)$$

where p_i is the empirical class frequency at the node and $C = 7$. The model is highly interpretable but prone to overfitting at large depths.

Random forest [19] aggregates B decorrelated trees, each fitted on a bootstrap sample and using a random subset of features at every split. The ensemble output is the (soft) majority vote of the individual trees,

$$\hat{f}(x) = \frac{1}{B} \sum_{b=1}^B f_b(x). \quad (7)$$

Bagging combined with feature subsampling substantially reduces the variance component of the generalisation error relative to a single tree.

XGBoost [9] fits trees sequentially; at iteration t a new tree f_t is added to reduce the residual error of the current ensemble, with objective

$$\mathcal{L}^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t), \quad (8)$$

where $\Omega(f_t)$ combines ℓ_1/ℓ_2 regularisation on the leaf weights with a complexity cost on the number of leaves. Splits are evaluated via second-order Taylor expansion.

LightGBM [10] accelerates gradient boosting through (i) leaf-wise tree growth (only the leaf with the largest expected loss reduction is expanded) and (ii) histogram-based binning of continuous features. The two modifications cut both the training time and the memory footprint while preserving accuracy.

3.4. Hyperparameter tuning

To avoid the bias of manual parameter selection and obtain a fair comparison between models, every algorithm except Naive Bayes was tuned with Randomized-SearchCV (60 candidate configurations sampled from the search spaces defined in Table 2). Naive Bayes, whose only relevant parameter is `var_smoothing`, was tuned with an exhaustive GridSearchCV (12 candidates on a logarithmic grid). To prevent data leakage during hyperparameter selection, tuning was performed in an *inner* stratified 5-fold cross-validation loop run *exclusively on the training partition*; the held-out test set was never seen during this stage. The optimisation criterion was macro-averaged F_1 , which weights all seven classes equally. The selected hyperparameters are summarised in Table 2.

Table 2: Hyperparameter configurations selected by the inner cross-validation. Continuous parameters were sampled from log-uniform or uniform distributions and integer parameters from discrete uniform distributions over typical ranges; the values below are those of the best-scoring candidate for each model. Parameters not listed take the Scikit-Learn 1.8 defaults.

Model	Selected hyperparameters
Logistic Regression	solver: lbfgs, penalty: L2, C: 63.95,
KNN	max_iter: 2000 $k = 3$, weights: uniform, Minkowski $p = 1$ (Manhattan)
Decision Tree	criterion: gini, max_depth: 15, min_samples_split: 2, min_samples_leaf: 2
Random Forest	n_estimators: 229, max_depth: none, min_samples_split: 3, min_samples_leaf: 1, max_features: all
XGBoost	n_estimators: 244, max_depth: 5, learning_rate: 0.058, subsample: 0.90, colsample_bytree: 0.99, reg_alpha: 0.039, reg_lambda: 0.054
LightGBM	n_estimators: 119, num_leaves: 79, max_depth: 30, learning_rate: 0.055, subsample: 0.62, colsample_bytree: 0.80, reg_alpha: 0.002, reg_lambda: 0.039
SVM (RBF)	C: 54.57, γ : 0.0159, kernel: RBF

Model	Selected hyperparameters
Naive Bay- es	distribution: Gaussian, var_smoothing: $6.6 \cdot 10^{-7}$

3.5. Evaluation protocol

Evaluation followed a two-stage protocol. In the first stage, every tuned model was re-fitted within a stratified *10-fold* cross-validation on the training partition. This yields, for each model, ten accuracy and ten macro- F_1 scores from which the mean and standard deviation are reported; the per-fold scores also serve as paired observations for the Wilcoxon signed-rank test (Section 3.6). The same fold splits, generated with a fixed random seed, were used for every model so that comparisons are paired rather than independent. In the second stage, each model was re-fitted on the full training partition and evaluated on the held-out test set using the metrics defined below.

- Accuracy — the proportion of correctly classified samples:

$$\text{Accuracy} = \frac{\sum_{i=1}^c TP_i}{N}, \quad (9)$$

where TP_i is the number of correct predictions for class i and N is the total number of samples.

- Macro F_1 -score — the harmonic mean of precision and recall, averaged uniformly across classes:

$$F1_{\text{macro}} = \frac{1}{c} \sum_{i=1}^c \frac{2 \cdot \text{Prec}_i \cdot \text{Rec}_i}{\text{Prec}_i + \text{Rec}_i}. \quad (10)$$

- Cohen's kappa coefficient (κ) — a measure of agreement between predictions and labels that accounts for the probability of agreement by chance.
- Log-loss — the logarithmic loss function measuring the calibration of the model's output probabilities [20].
- ROC-AUC (one-vs-rest, macro) — the area under the receiver operating characteristic (ROC) curve computed in a one-vs-rest fashion for each class and averaged uniformly [20], which measures the model's discriminative power independently of any classification threshold.

3.6. Statistical significance testing

To verify whether the observed differences between models are systematic or compatible with sampling variability, two complementary paired tests were applied to every one of the $\binom{8}{2} = 28$ pairs of models:

- McNemar's test [21], applied to the held-out test predictions. For two classifiers A and B , it analyses the off-diagonal entries of the contingency table (samples that one classifier predicts correctly while the other does not).
- Wilcoxon signed-rank test [22], applied to the ten paired per-fold scores (accuracy and macro- F_1 separately) obtained from the cross-validation stage.

Both tests were carried out at $\alpha = 0.05$. To control the family-wise error rate across the 28 simultaneous com-

parisons, p -values were adjusted by the Holm-Bonferroni step-down procedure [23]; pairs are reported as significant only when $p_{\text{Holm}} < \alpha$.

3.7. Interpretability

Four complementary methods were used to quantify the role of individual features:

- *Split-gain importance* — the native `feature_importances_` attribute of the gradient-boosting models, expressing the cumulative loss reduction obtained from all splits performed on a given feature.
- *SHAP values* [24], computed with the TreeExplainer on a stratified subsample of 200 test instances, giving a model-agnostic and theoretically grounded measure of each feature's global contribution.
- *Permutation importance*, computed on the test set with 10 repeats and the macro- F_1 scoring function, which is comparable across model families.
- *Local LIME explanations* [5]: for each of the seven obesity classes, one representative test instance was selected (the correctly classified sample with the highest predicted probability for its true class), and a sparse linear surrogate was fitted in a neighbourhood of that instance to identify the features most responsible for the local prediction.

3.8. Test environment

The experiments were carried out in the Python 3.12 environment using Scikit-Learn 1.8, XGBoost 3.2, LightGBM 4.6, SHAP 0.51 and LIME 0.2. The computations were performed exclusively on the CPU without GPU acceleration, which corresponds to typical requirements of e-health systems. The platform specification is presented in Table 3. The random seed was fixed at 42 throughout the pipeline (train/test split, CV fold generation, randomised search and stochastic model initialisation).

Table 3: Hardware and software specification of the test platform.

Component	Specification
Processor (CPU)	Intel Core i7-12700H (14 cores, max. 4.7 GHz)
RAM	16 GB LPDDR5 (4800 MHz)
GPU	NVIDIA RTX 3060 (unused)
Operating system	Windows 11 Pro (64-bit)
ML environ- ment	Python 3.12, Scikit-Learn 1.8, XGBoost 3.2, LightGBM 4.6, SHAP 0.51, LIME 0.2

4. Results and discussion

4.1. Analysis of the main results

Figure 1 presents accuracy and macro- F_1 on the test set. The gradient boosting methods — XGBoost (accuracy 96.17 %, $F_1 = 0.962$) and LightGBM (accuracy 95.69 %, $F_1 = 0.956$) — achieved the highest results in both metrics. Logistic regression, after tuning, reached an accuracy of 95.93 % (which approaches the boosting models

but, as shown below, with substantially worse probability calibration), SVM-RBF obtained 95.45 %, and Random Forest 94.74 %. Decision Tree (92.58 %) and KNN (87.80 %) are clearly weaker; Naive Bayes, with 55.50 %, falls dramatically below the rest, consistent with the strong feature correlations in the dataset (body weight and height) that violate the conditional-independence assumption underpinning the model.

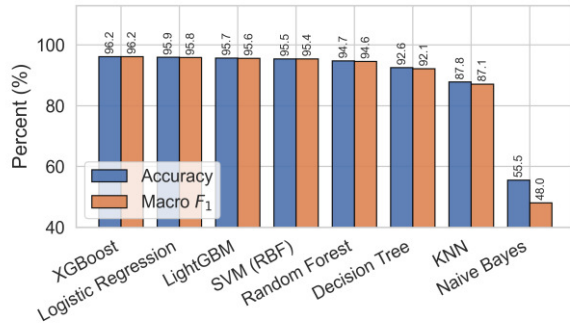


Figure 1: Comparison of Accuracy and Macro F1-score values on the test set for all evaluated models.

The log-loss column of Table 5 reflects probability calibration. The boosting models (XGBoost: 0.098; LightGBM: 0.115), SVM-RBF (0.132) and Logistic Regression (0.171) are well calibrated, whereas the Decision Tree (1.769) and KNN (2.053) produce poorly calibrated probabilities despite their decent accuracy — a significant drawback in medical applications. Naive Bayes (3.684) is the worst-calibrated model. The ROC-AUC ranking is consistent: XGBoost = LightGBM = 0.999, SVM-RBF = 0.998, Logistic Regression = 0.997, Random Forest = 0.996, while Decision Tree (0.970), KNN (0.963) and Naive Bayes (0.912) trail noticeably.

Figure 2 illustrates the stability of the models within the stratified 10-fold cross-validation; the per-fold means and standard deviations are summarised in Table 4. XGBoost achieved the highest mean CV accuracy (97.01 % $\pm$ 1.54 percentage points (p.p.)); LightGBM reached 96.47 % $\pm$ 1.64 p.p., and SVM-RBF 96.11 % $\pm$ 0.94 p.p. with the lowest variability among the top performers. Decision Tree (92.27 % $\pm$ 2.34 p.p.) shows the largest variability among the tree-based models, confirming its known susceptibility to overfitting in individual folds. The fact that all eight evaluated models exhibit per-fold standard deviations below 2.4 p.p. across ten independent folds is the empirical justification for the claim that the headline test-set numbers reflect genuine performance differences rather than artefacts of a particular random split.

Table 4: Cross-validation performance on the training partition (stratified 10-fold CV, hyperparameters from Table 2). Mean $\pm$ standard deviation across folds.

Model	CV accuracy	CV macro- F_1
XGBoost	0.9701 $\pm$ 0.0154	0.9688 $\pm$ 0.0157
LightGBM	0.9647 $\pm$ 0.0164	0.9631 $\pm$ 0.0170
SVM (RBF)	0.9611 $\pm$ 0.0094	0.9591 $\pm$ 0.0098
Logistic Regression	0.9545 $\pm$ 0.0180	0.9524 $\pm$ 0.0190
Random Forest	0.9515 $\pm$ 0.0199	0.9497 $\pm$ 0.0207
Decision Tree	0.9227 $\pm$ 0.0234	0.9203 $\pm$ 0.0240
KNN	0.8700 $\pm$ 0.0222	0.8604 $\pm$ 0.0237
Naive Bayes	0.5381 $\pm$ 0.0197	0.4522 $\pm$ 0.0254

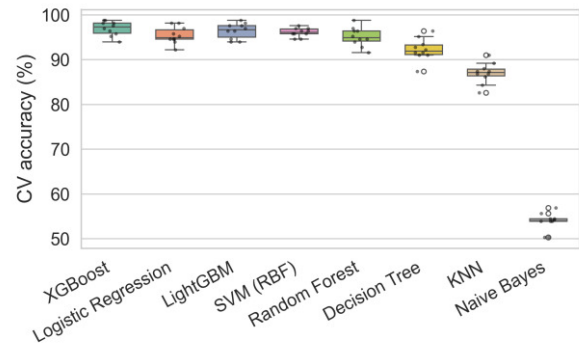


Figure 2: Distribution of accuracy across stratified 10-fold cross-validation on the training partition. Individual dots show the ten per-fold scores; the box widths reflect the prediction stability of each model.

Figure 3 compares accuracy with training time. LightGBM completes a full training pass in 0.58 s while reaching 95.69 % accuracy — a favourable property when frequent retraining is required. XGBoost (0.88 s, 96.17 %) and SVM-RBF (0.35 s, 95.45 %) offer comparable compromises. Decision Tree (0.03 s, 92.58 %) is fastest but trades accuracy and calibration for that speed.

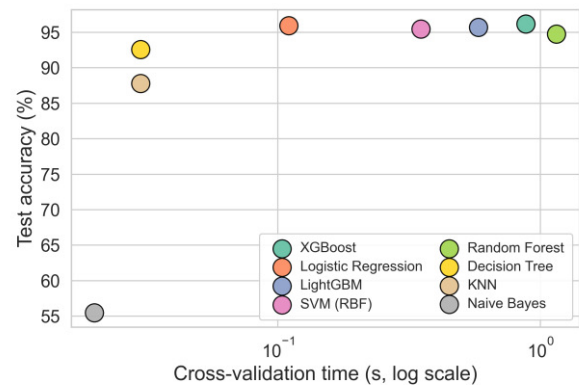


Figure 3: Test-set accuracy vs. cross-validation training time. Boosting models combine high accuracy with short training time.

A complete summary of test-set performance is presented in Table 5.

Table 5: Classification results on the held-out test set ($n = 418$).

Model	Acc	F_1	κ	Log-loss	ROC-AUC
XGBoost	0.962	0.962	0.955	0.098	0.999
Log. Regression	0.959	0.958	0.952	0.171	0.997
LightGBM	0.957	0.956	0.950	0.115	0.999
SVM (RBF)	0.955	0.954	0.947	0.132	0.998
Random Forest	0.947	0.946	0.938	0.233	0.996
Decision Tree	0.926	0.921	0.913	1.769	0.970
KNN	0.878	0.871	0.857	2.053	0.963
Naive Bayes	0.555	0.480	0.483	3.684	0.912

The Cohen's κ values reported in Table 5 confirm the same ordering: XGBoost ($\kappa = 0.955$) and LightGBM ($\kappa = 0.950$) reach very high agreement with the true labels, well above the level of agreement by chance. Under the Landis and Koch convention, $\kappa > 0.80$ qualifies as “almost perfect”; seven of the eight evaluated models fall within this range, the sole exception being Naive Bayes ($\kappa = 0.483$).

4.2. Statistical significance of pairwise differences

Applying McNemar's test to the test-set predictions of all 28 model pairs, then adjusting the resulting p -values with the Holm–Bonferroni procedure, yields 12 significant pairwise differences at $\alpha = 0.05$. The detailed pattern is informative. Naive Bayes is significantly worse than every other model (all $p_{\text{Holm}} < 10^{-3}$), and KNN is significantly worse than each of the five top-performing models (XGBoost, Logistic Regression, LightGBM, SVM-RBF, Random Forest). Within the leading cluster — XGBoost, LightGBM, Random Forest, SVM-RBF, Logistic Regression and (to a smaller extent) Decision Tree — no pairwise difference survives correction: in particular, XGBoost vs. LightGBM gives $p_{\text{Holm}} = 1.000$, Random Forest vs. XGBoost gives $p_{\text{Holm}} = 1.000$, and LightGBM vs. SVM-RBF gives $p_{\text{Holm}} = 1.000$. Two pairs involving Decision Tree fall just outside significance after correction ($p_{\text{Holm}} = 0.082$ vs. XGBoost; $p_{\text{Holm}} = 0.159$ vs. LightGBM), which reflects the modestly lower accuracy of a single tree relative to boosted ensembles.

Figure 4 visualises the Holm-adjusted p -values across all 28 pairs and makes the cluster structure of the comparison immediately visible: a top performing group (XGBoost, Logistic Regression, LightGBM, SVM-RBF, Random Forest) with no significant pairwise differences, an intermediate Decision Tree, and clearly weaker KNN and Naive Bayes that are significantly worse than the top group.

The Wilcoxon signed-rank test on the ten paired per-fold scores yields no pair that is significant after Holm correction. This is a power issue rather than an absence

of effect: with only ten paired observations the smallest two-sided p from a Wilcoxon test is $2/2^{10} \approx 0.002$, which after Holm-correction for 28 comparisons becomes ≈ 0.055 , just above α . Pairs that are well separated in mean CV scores — e.g. KNN vs. XGBoost (median $\Delta F_1 = -0.115$, raw $p = 0.002$) — therefore approach but do not reach corrected significance. We treat the McNemar results as the primary statistical evidence, with the Wilcoxon test providing a fold-level consistency check. The converging picture of low per-fold standard deviations (Table 4), twelve significant McNemar pairs, and a five-model top cluster (Figure 4) supports the interpretation that the observed ranking is not an artefact of the random train/test split.

4.3. Model interpretability

Following the protocol described in Section 3.7, four complementary feature-importance methods were applied to the best-performing model on the test set (XGBoost) and, for comparison, to LightGBM. Figure 5 presents the split-gain importance of LightGBM, which expresses the cumulative loss reduction resulting from splits on each feature. The variables with the highest predictive contribution are body weight (gain 4015) and height (3534) — directly related to the body mass index (BMI) — followed by age (2110), physical-activity frequency FAF (1102), the frequency of vegetable consumption FCVC (1050) and screen-time TUE (1029). Variables with a marginal contribution (notably tobacco smoking SMOKE and calorie monitoring SCC) were practically disregarded by the model.

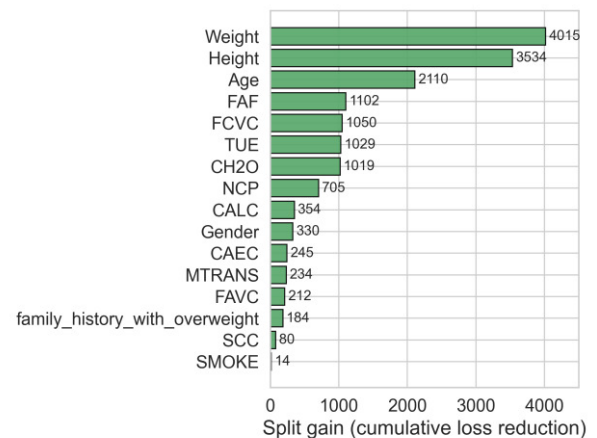


Figure 5: Split-gain feature importance of the LightGBM model. Body weight, height and dietary habits dominate as predictors of obesity level.

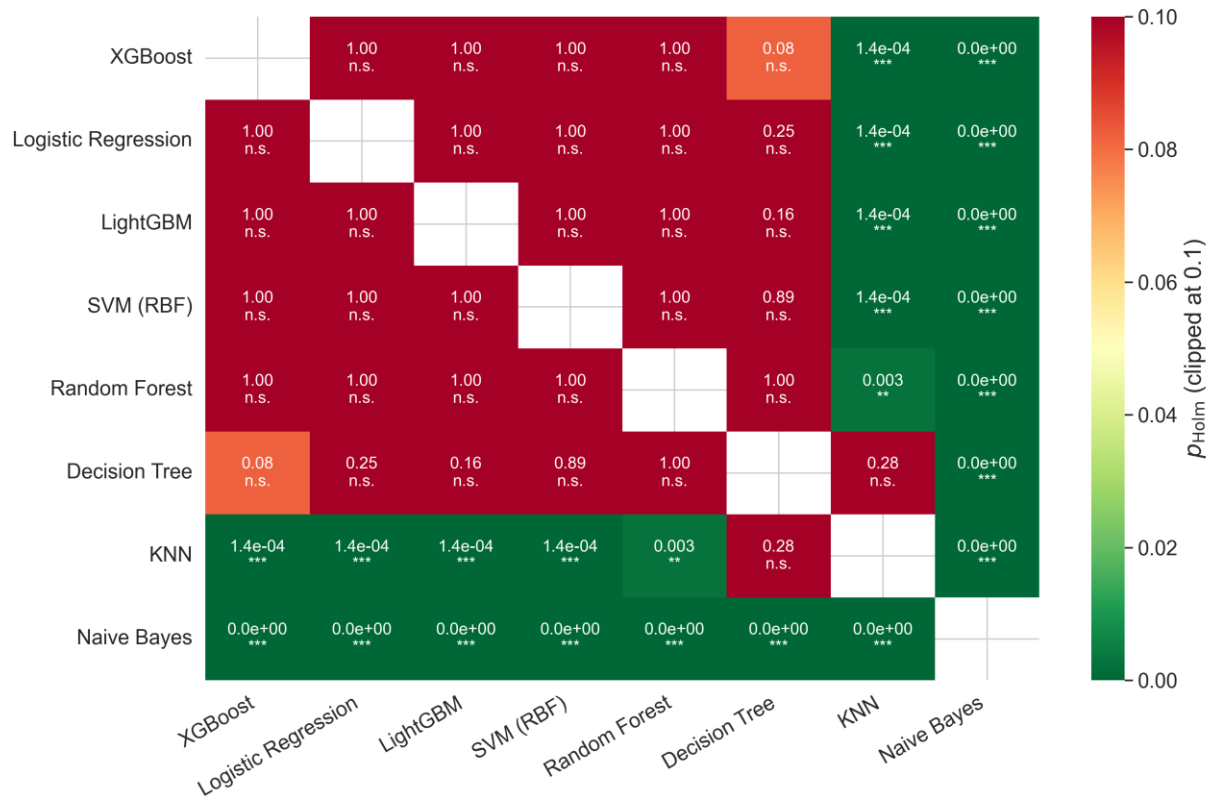


Figure 4: Holm-Bonferroni-corrected p -values from pairwise McNemar tests on the held-out test predictions. Cells with $p_{\text{Holm}} < 0.05$ are marked with an asterisk and shown in green; non-significant cells are shown in red. Models are ordered by test accuracy.

Three additional measures of feature importance were compared with the split-gain ranking. *SHAP global importance* (mean $|\text{SHAP}|$ aggregated across classes for XGBoost) places Weight, Height and Gender at the top, with a clear separation from the behavioural variables. *Permutation importance* on the test set with macro- F_1 scoring identifies the same top-three; the largest drop — ≈ 0.76 in macro- F_1 — is observed when Weight is permuted, followed by Height (≈ 0.25) and Gender (≈ 0.14). *Local LIME explanations* for seven correctly classified test instances (one per obesity class) are shown in Figure 6; the induced local rules are clinically consistent — e.g. *Insufficient_Weight* is explained by $\text{Weight} \leq 65.1 \text{ kg}$, *Overweight_Level I* by $65.1 < \text{Weight} \leq 83.0 \text{ kg}$, *Obesity_Type III* by $\text{Weight} > 107.2 \text{ kg}$ — in each case combined with smaller contributions from Height, Age, Gender and dietary habits.

All four importance measures place body weight and height at the top of the ranking; the third position is gender (SHAP, permutation) or age (LightGBM’s split gain). The fact that a built-in tree statistic, a Shapley-value approach, a model-agnostic permutation test, and a sparse local surrogate independently recover the BMI definition strengthens the interpretability claim and demonstrates that tree-based methods can extract clinically interpretable rules from questionnaire data.

4.4. Discussion

The results provide qualified support for the formulated hypothesis. The XGBoost test-set accuracy of 96.17% ($\kappa = 0.955$) constitutes a substantial improvement over the baselines of Palechor and de la Hoz Manotas [2] (80–85%) and Ferdowsy et al. [8] (88–93%). The statistical analysis in Section 4.2 reveals, however, that gradient boosting models are *not* significantly better than a properly tuned SVM-RBF, Random Forest or Logistic Regression on this dataset; the clear separation predicted by the hypothesis materialises only against Decision Tree, KNN and Naive Bayes. The likely reason is the highly informative pair of anthropometric features (Weight, Height), from which BMI — effectively the deterministic generator of the label — is computed: once these features are available, even simple decision surfaces suffice to reach above 95% accuracy, and the inductive advantage of boosting becomes hard to detect on a dataset of this size. The low effectiveness of Naive Bayes (55.50%) confirms that conditional independence is violated, while KNN’s degradation and the poor calibration of the Decision Tree illustrate how strongly model assumptions interact with the data structure.

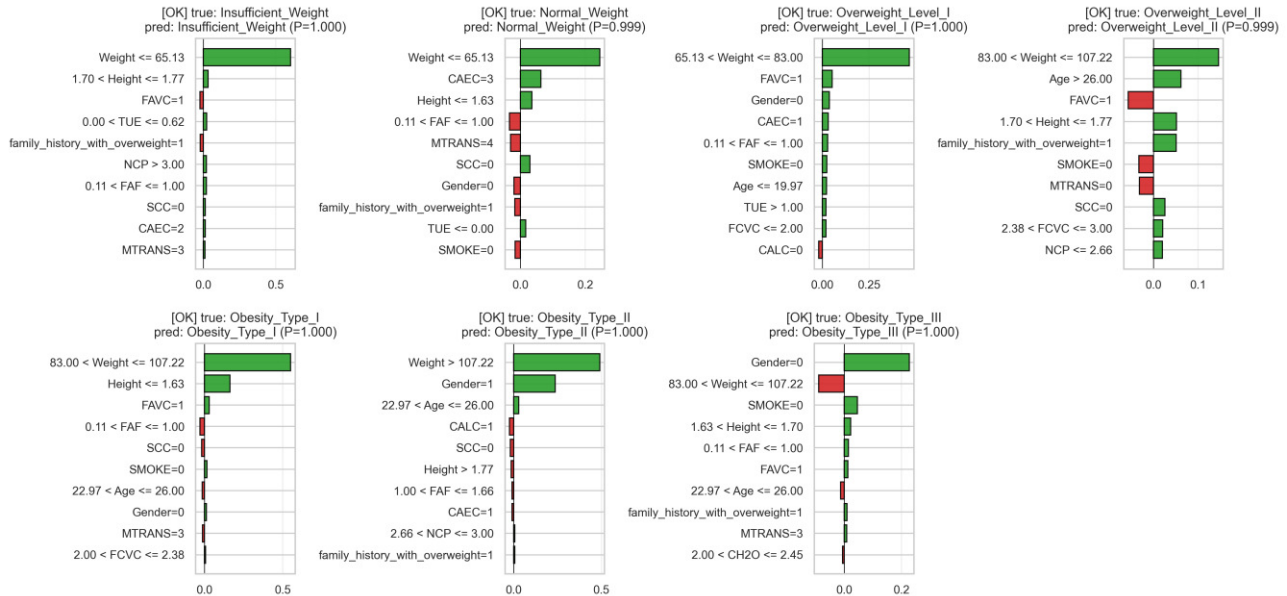


Figure 6: Local LIME explanations of LightGBM for one representative test instance per obesity class (seven panels, one per class). Green bars indicate features pushing the prediction towards the true class; red bars indicate features pushing against it. The numerical thresholds on Weight match the BMI-based boundaries between adjacent obesity categories.

5. Conclusions

Gradient boosting tree algorithms achieve the numerically highest classification effectiveness on this dataset — XGBoost reached 96.17 % accuracy ($\kappa = 0.955$) on the held-out test set and the highest mean CV accuracy (97.01 %), with LightGBM matching it within sampling variability. Both models combine short training times (≤ 1 s) with excellent probability calibration (log-loss ≤ 0.115 , ROC-AUC = 0.999) and are therefore strong candidates for deployment in clinical decision support systems running on resource-constrained devices. The McNemar tests with Holm–Bonferroni correction, however, identify a top-performing cluster of five models (XGBoost, Logistic Regression, LightGBM, SVM-RBF, Random Forest) between which no pairwise difference reaches significance — a methodologically important nuance that a raw-accuracy reading would have missed. The multi-method interpretability analysis converged on body weight and height as dominant predictors, and the local LIME explanations independently recovered weight thresholds consistent with the BMI definition — the model effectively re-learned a clinical definition from questionnaire data.

5.1. Directions for further research

Four extensions for future studies appear natural: (i) addressing residual class imbalance through the Synthetic Minority Over-sampling Technique (SMOTE) [25] or related oversampling techniques; (ii) replacing the randomised search by sequential model-based Bayesian optimisation [26], which typically reaches comparable configurations within a smaller candidate budget; (iii) benchmarking against deep tabular architectures such as the multilayer perceptron (MLP), in line with the open question raised by [3, 4]; (iv) applying ensemble learning methods for improving classification metrics [27].

References

- [1] World Health Organization, Obesity and overweight – Fact sheet, <https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight>, [2022].
- [2] F. M. Palechor, A. de la Hoz Manotas, Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru and Mexico, Data in Brief 25 (2019) 104344, <https://doi.org/10.1016/j.dib.2019.104344>.
- [3] L. Grinsztajn, E. Oyallon, G. Varoquaux, Why do tree-based models still outperform deep learning on typical tabular data?, Advances in Neural Information Processing Systems 35 (2022) 507–520, <https://arxiv.org/abs/2207.08815>.
- [4] R. Shwartz-Ziv, A. Armon, Tabular Data: Deep Learning is Not All You Need, Information Fusion 81 (2022) 84–90, <https://doi.org/10.1016/j.inffus.2021.11.011>.
- [5] M. T. Ribeiro, S. Singh, C. Guestrin, “Why Should I Trust You?”: Explaining the Predictions of Any Classifier, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016) 1135–1144, <https://doi.org/10.1145/2939672.2939778>.
- [6] D. Chicco, G. Jurman, Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone, BMC Medical Informatics and Decision Making 20(1) (2020) 16, <https://doi.org/10.1186/s12911-020-1023-5>.
- [7] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, Journal of Machine Learning Research 15 (2014) 3133–3181, <https://jmlr.org/papers/v15/delgado14a.html>.
- [8] F. Ferdowsy, K. S. A. Rahi, M. Jabiullah, M. T. Habib, A machine learning approach for obesity risk prediction, Current Research in Behavioral Sciences 2 (2021) 100053, <https://doi.org/10.1016/j.crbeha.2021.100053>.

- [9] T. Chen, C. Guestrin, XGBoost: A Scalable Tree Boosting System, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016) 785–794, <https://doi.org/10.1145/2939672.2939785>.
- [10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A Highly Efficient Gradient Boosting Decision Tree, Advances in Neural Information Processing Systems 30 (2017) 3146–3154, <https://papers.nips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html>.
- [11] L. Rokach, Ensemble-based classifiers, Artificial Intelligence Review 33(1–2) (2010) 1–39, <https://doi.org/10.1007/s10462-009-9124-7>.
- [12] I. D. Mienye, Y. Sun, A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects, IEEE Access 10 (2022) 99129–99149, <https://doi.org/10.1109/ACCESS.2022.3207287>.
- [13] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, A. Lopez, A comprehensive survey on support vector machine classification: Applications, challenges and trends, Neurocomputing 408 (2020) 189–215, <https://doi.org/10.1016/j.neucom.2019.10.118>.
- [14] D. W. Hosmer, S. Lemeshow, R. X. Sturdivant, Applied Logistic Regression, 3rd ed., John Wiley & Sons, Hoboken, NJ, 2013, <https://doi.org/10.1002/9781118548387>.
- [15] I. Rish, An empirical study of the naive Bayes classifier, In IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence (2001) 41–46.
- [16] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13(1) (1967) 21–27, <https://doi.org/10.1109/TIT.1967.1053964>.
- [17] C. Cortes, V. Vapnik, Support-vector networks, Machine Learning 20(3) (1995) 273–297, <https://doi.org/10.1007/BF00994018>.
- [18] L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone, Classification and Regression Trees, Wadsworth, Belmont, CA, 1984.
- [19] L. Breiman, Random Forests, Machine Learning 45(1) (2001) 5–32, <https://doi.org/10.1023/A:1010933404324>.
- [20] T. Fawcett, An introduction to ROC analysis, Pattern Recognition Letters 27(8) (2006) 861–874, <https://doi.org/10.1016/j.patrec.2005.10.010>.
- [21] T. G. Dietterich, Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms, Neural Computation 10(7) (1998) 1895–1923, <https://doi.org/10.1162/089976698300017197>.
- [22] J. Demšar, Statistical Comparisons of Classifiers over Multiple Data Sets, Journal of Machine Learning Research 7 (2006) 1–30, <https://jmlr.org/papers/v7/demsar06a.html>.
- [23] S. Holm, A Simple Sequentially Rejective Multiple Test Procedure, Scandinavian Journal of Statistics 6(2) (1979) 65–70, <https://www.jstor.org/stable/4615733>.
- [24] S. M. Lundberg, S.-I. Lee, A Unified Approach to Interpreting Model Predictions, Advances in Neural Information Processing Systems 30 (2017) 4765–4774, <https://papers.nips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html>.
- [25] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: Synthetic Minority Over-sampling Technique, Journal of Artificial Intelligence Research 16 (2002) 321–357, <https://doi.org/10.1613/jair.953>.
- [26] J. Snoek, H. Larochelle, R. P. Adams, Practical Bayesian Optimization of Machine Learning Algorithms, Advances in Neural Information Processing Systems 25 (2012) 2951–2959, <https://papers.nips.cc/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html>.
- [27] A. Szymczyk, M. Skublewska-Paszkowska, P. Powroźnik, CNN-Based Ensemble Architectures with Explainable AI for Cutaneous Melanoma Identification, Acta Mechanica et Automatica 20(2) (2026) 317–330, <https://doi.org/10.65731/ama/2026-0033>.