arXiv:2509.02826v1 [cs.LG] 2 Sep 2025

Ensemble Learning for Healthcare: A Comparative Analysis of Hybrid Voting and Ensemble Stacking in Obesity Risk Prediction

Towhidul Islam[1*] and Md Sumon Ali[2]

[1*]Information & Computer Science, King Fahd University of Petroleum and Minerals, Academic Belt Road, Dhahran, 31261, Dammam, Saudi Arabia.

[2]Information & Computer Science, King Fahd University of Petroleum and Minerals, Academic Belt Road, Dhahran, 31261, Dammam, Saudi Arabia.

*Corresponding author(s). E-mail(s): g202416880@kfupm.edu.sa;
Contributing authors: g202320610@kfupm.edu.sa;

Abstract

Obesity is a critical global health issue driven by dietary, physiological, and environmental factors, and is strongly associated with chronic diseases such as diabetes, cardiovascular disorders, and cancer. Machine learning has emerged as a promising approach for early obesity risk prediction, yet a comparative evaluation of ensemble techniques—particularly hybrid majority voting and ensemble stacking—remains limited. This study aims to compare hybrid majority voting and ensemble stacking methods for obesity risk prediction, identifying which approach delivers higher accuracy and efficiency. The analysis seeks to highlight the complementary strengths of these ensemble techniques in guiding better predictive model selection for healthcare applications. Two datasets were utilized to evaluate three ensemble models: Majority Hard Voting, Weighted Hard Voting, and Stacking (with a Multi-Layer Perceptron as meta-classifier). A pool of nine Machine Learning (ML) algorithms, evaluated across a total of 50 hyperparameter configurations, was analyzed to identify the top three models to serve as base learners for the ensemble methods. Preprocessing steps involved dataset balancing, and outlier detection, and model performance was evaluated using Accuracy and F1-Score. On Dataset-1, weighted hard voting and stacking achieved nearly identical performance (Accuracy: 0.920304, F1: 0.920070), outperforming majority hard voting. On Dataset-2, stacking demonstrated superior results (Accuracy:

0.989837, F1: 0.989825) compared to majority hard voting (Accuracy: 0.981707, F1: 0.981675) and weighted hard voting, which showed the lowest performance. The findings confirm that ensemble stacking provides stronger predictive capability, particularly for complex data distributions, while hybrid majority voting remains a robust alternative.

Keywords: Obesity, Risk-Prediction, Machine Learning, Ensemble Learning

1 Introduction

Obesity is one of the most pressing global health challenges, affecting millions of people worldwide and contributing to a significant burden of chronic diseases such as diabetes, cardiovascular disorders, and cancer [1][2]. Its prevalence has increased dramatically in recent decades due to dietary habits, sedentary lifestyles, physiological predispositions, and environmental influences [3]. According to the World Health Organization, obesity is now recognized as a critical public health concern, demanding scalable and effective approaches for early detection and prevention.

Traditional diagnostic methods rely largely on Body Mass Index (BMI), waist-to-hip ratios, or clinical evaluation, which often fail to capture the multifactorial causes of obesity. Recent advances in Machine Learning (ML) have enabled the development of predictive models that integrate diverse factors such as demographics, dietary patterns, physical activity levels, and physiological indicators [4]. These data-driven approaches have demonstrated strong potential in early risk prediction, enabling personalized interventions and targeted public health strategies within healthcare applications.

However, while numerous studies have applied individual ML algorithms for obesity prediction, balancing high predictive performance with generalizability remains a major challenge. In addition, the comparative effectiveness of ensemble learning techniques—particularly hybrid majority voting and ensemble stacking—in this domain has not been thoroughly investigated. Ensemble methods combine multiple classifiers to improve predictive robustness [5]. Majority Voting aggregates the outputs of the base classifiers in either hard or weighted voting schemes, providing a simple but effective approach [6]. In contrast, Stacking employs a meta-classifier to learn how best to combine the predictions of base models, potentially capturing more complex interactions among them [7].

A systematic evaluation of these approaches for the prediction of obesity risk is still lacking. This study aims to address the following research questions:

1. Which individual machine learning algorithms serve as the strongest base models for obesity prediction?
2. How does hybrid majority voting compare with ensemble stacking in terms of predictive performance and generalizability?

3. What potential techniques can further enhance the robustness and interpretability of ensemble models for obesity prediction?

This study investigates the design and comparative evaluation of two ensemble learning approaches—Hybrid Majority Voting and Ensemble Stacking—for the prediction of obesity risk using structured tabular data. To achieve this, two benchmark obesity datasets were employed [8][9], with preprocessing steps including dataset balancing and outlier detection to improve data quality and reliability. Although prior research has applied individual machine learning models to obesity prediction, no systematic study has compared the effectiveness of Hybrid Majority Voting—which incorporates Majority Hard Voting and Weighted Hard Voting—against Ensemble Stacking, which leverages a Multi-Layer Perceptron (MLP) as the meta-classifier, within a unified experimental framework.

To construct the ensembles, nine ML algorithms were explored under a total of 50 hyperparameter configurations. From these, the top three models were selected as base learners. This extensive search served two key purposes: first, to identify strong standalone classifiers that capture different levels of predictive complexity; and second, to maximize diversity among base learners by drawing from multiple algorithmic families, including decision trees, gradient-based methods, instance-based learners, and neural networks. Such heterogeneity is essential for ensemble models, as it enhances robustness and overall predictive performance.

Existing literature has not systematically examined the comparative performance of Hybrid Majority Voting and Ensemble Stacking for obesity risk prediction. Furthermore, no prior work has integrated an extensive hyperparameter tuning process with ensemble construction in this domain. This paper makes the following three key contributions:

1. A comprehensive comparative analysis of Hybrid Majority Voting (including Majority Hard Voting and Weighted Hard Voting) and Ensemble Stacking for obesity risk prediction, addressing a gap in the literature.
2. A rigorous model selection process, evaluating nine ML algorithms under 50 hyperparameter configurations, to ensure robust base learner identification for ensemble construction.
3. Actionable insights into the strengths and limitations of the ensemble, demonstrating how different approaches can guide accurate and scalable obesity risk prediction, with implications for targeted public health campaigns and personalized healthcare interventions.

2 Related Works

Several research has been explored by the use of machine learning to predict obesity. For instance, a study by Pinar et al., 2024 [2] demonstrated the application of logistic regression in classifying the levels of obesity in affected individuals.

Again, Dutta et al., 2024 [10] applied Support Vector Machine (SVM), Random Forest (RF), and Decision Tree (DT) algorithms to identify the risk of obesity. Here, RF gave the best prediction accuracy compared to the other models.

Similarly, a 2023 paper by Talari et al. [11] leveraged hybrid majority voting techniques for the prediction and classification of obesity. Before applying ensemble model, seven distinct ML algorithms were used on a UCI ML repository dataset to select the best models that outperformed in terms of performance metrics. Based on the performance, later, Gradient Boosting Classifier, Extreme Gradient Boosting, and a MLP were used to build majority hard voting ensemble model that outperformed the other individual models.

In 2022, an article by Musa et al. [12] used public clinical dataset to predict obesity status, where five ML algorithms were applied. Here Gboost gave the highest accuracy as compared to the other models while K Nearest Neighbor (KNN) gave the relatively strong accuracy.

Again, a 2021 article by Rodríguez et al. [13] applied ML models to develop an intelligent model for the identification of people with obesity or overweight, where RF outperformed compared to the other models.

Similarly, a 2018 study by Jindal et al. [14] demonstrated R ensemble technique and python interface for the early prediction of obesity. In terms of future work, more than three ML models can be applied to observe the strengths of different individual models. Additionally, the findings can be improved by including precise obesity and BMI numbers for differently abled individuals.

However, a comparative analysis between the hybrid majority voting and ensemble stacking remains an area with significant potential for improvement. This project aims to build two proposed models to demonstrate their strengths and weakness for predicting the risk of obesity of affected individuals.

3 Methodology

3.1 Design of the Experiment

Purpose: To analyze and evaluate the effectiveness of Hybrid Majority Voting and Ensemble Stacking methods for early prediction of obesity using structured tabular data.

Issue: By accurately classifying obesity risk levels across two benchmark datasets and identifying which ensemble technique—Majority Voting or Stacking—offers higher predictive performance and robustness.

Object: Using model performance metrics such as Accuracy and F1-Score to assess predictive capability, while comparing the relative strengths and limitations of the two ensemble approaches.

Viewpoint: From the viewpoint of researchers and healthcare professionals interested in developing scalable, accurate, and data-driven tools for early obesity risk prediction to guide public health interventions and personalized healthcare.

So, according to the Goal Question Metric (GQM) framework [15], the main objective is to analyze and evaluate the effectiveness of Hybrid Majority Voting and Ensemble Stacking methods for early prediction of obesity using tabular datasets, employing model performance metrics (Accuracy and F1-Score) to assess predictive accuracy and robustness, from the viewpoint of researchers and healthcare professionals aiming to improve early detection and decision-making in obesity management. In Fig. 1, the detailed methodology of our work is illustrated.

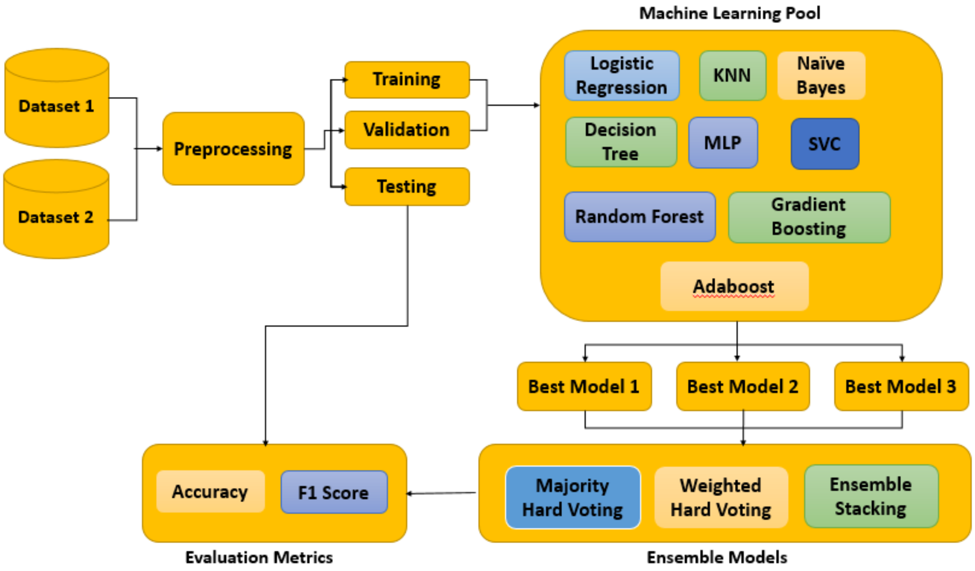


Fig. 1 Detailed methodology of proposed work.

3.2 Dataset Description

In this experiment, two distinct datasets were utilized to develop a predictive system for obesity classification. One dataset was taken from Kaggle [8], and the other was obtained from GitHub [9]. These datasets provided essential demographic, lifestyle, and physiological data points to train and evaluate machine learning models effectively.

3.2.1 Obesity Dataset-1 (Kaggle)

The first dataset, available on Kaggle, contains a total of 14 features, all of which are numerical (int64). These features represent demographic and lifestyle-related attributes closely associated with obesity risk. The representation of features are illustrated in Table 1.

Table 1: Features of dataset 1 and their corresponding data types.

Features	Types
'Sex', 'Age', 'Height', 'Overweight_Obese_Family', 'Consumption_of_Fast_Food', 'Frequency_of_Consuming_Vegetables', 'Number_of_Main_Meals_Daily', 'Food_Intake_Between_Meals', 'Smoking', 'Liquid_Intake_Daily', 'Calculation_of_Calorie_Intake', 'Physical_Excercise', 'Schedule_Dedicated_to_Technology', 'Type_of_Transportation_Used'	int64

The output label for this dataset is the *Class* column, representing obesity categories. The label distribution with the count is represented in Table 2.

Table 2: Distribution of Class Labels in dataset 1

Class Label	Count
Underweight	73
Normal	658
Overweight	592
Obesity	287

This dataset provides a relatively balanced representation of obesity-related categories, though some classes (e.g., Underweight) are underrepresented.

3.2.2 Obesity Dataset-2 (GitHub)

The second dataset, obtained from GitHub, contains 16 features of mixed types: 8 numerical (float64) and 8 categorical (object) variables. The detailed features of dataset 2 are presented in Table 3.

Table 3: Features of dataset 2 and their corresponding data types.

Features	Types
Gender, family_history_with_overweight, FAVC, CAEC, SMOKE, SCC, CALC, MTRANS	object
Age, Height, Weight, FCVC, NCP, CH2O, FAF, TUE	float64

The output label for this dataset is the NObeyesdad column, representing obesity levels. Since this column is categorical, its values were converted into numerical form using Label Encoding. The label distribution with the with the count is represented in Table 4.

Table 4: Distribution of Class Labels in dataset 2

Class Label	**Count**
Insufficient Weight	272
Normal Weight	287
Overweight Level I	351
Overweight Level II	297
Obesity Type I	324
Obesity Type II	290
Obesity Type III	290

Compared to Dataset-1, Dataset-2 provides a more fine-grained classification of obesity, covering seven distinct categories. Its balanced distribution across multiple obesity levels makes it highly suitable for evaluating advanced classification models.

3.2.3 Comparison of Datasets

Table 5 compares the two datasets used in this study. While Dataset-1 is simpler, with 14 fully numerical features and 4 obesity categories, Dataset-2 is more complex, combining both categorical and numerical features and expanding the classification into 7 categories. Together, they provide complementary testbeds for evaluating the robustness and generalizability of ML models. The detailed comparison between the datasets are illustrated in Table 5.

Table 5: Comparison between Obesity Dataset-1 and Obesity Dataset-2

Attribute	**Dataset-1 (Kaggle)**	**Dataset-2 (GitHub)**
No. of Features	14 (all numerical)	16 (8 numerical, 8 categorical)
Output Label	Class	NObeyesdad
Label Type	Numerical	Object (encoded)
No. of Classes	4	7
Distribution Balance	Moderate	Relatively balanced

3.3 Execution of Experiment

The following section outlines the methodology of this study, which includes the steps undertaken for dataset selection and preprocessing, building a pool of ML algorithms with hyperparameter configurations, selecting the top-performing models, constructing ensemble models using Hybrid Majority Voting and Ensemble Stacking, and finally evaluating their performance using Accuracy and F1-Score.

3.3.1 Experimental Setup

The Experimental setup for this study was conducted in a Local Machine which used Python 3.11.4 as a runtime environment. The Local Machine used Ryzen 1600 with 6 Core and 12 Threads CPU, 16GB of Ram and an RTX 3060 12GB of GPU which was used for efficient training of the models. Pandas, Seaborn and Scikit Learn were used for Data Preprocessing and Splitting, scaling and model creation [16] [17] [18]. Imbalanced Learn was used for Over Sampling of the data. Matplotlib was used to visualize the training curves, evaluate the models and create confusion matrices. This setup allowed for efficient and effective experimentation with no compute unit constraints.

3.3.2 Data preparation

To achieve the best performance, the following data preparation steps are taken.

Planned Preprocessing: The following preprocessing steps are applied on both of the datasets.

- **Encoding:** As almost all of the data are numerical in dataset-1, encoding was not needed to apply for the dataset-1. On the other hand, for dataset-2, Label Encoding was applied to convert the object type value into the numerical value.
- **Normalization/Scaling:** Normalization was applied on both of the datasets before applying the ML algorithm. Normalization is needed because it may help to improve the performance for algorithms which are sensitive to scale like LR, SVM, etc.
- **Handling Outliers:** The datasets were checked to identify whether there was any outlier or not. There was no outlier in any of the datasets. Outlier checking is important in preprocessing because it ensures the dataset's integrity and prevents them from skewing or negatively impacting the ML model's performance.

Most Important Feature: Correlation analysis was applied on both of the datasets to identify the most important feature that was mostly close to the labels. From the analysis, 'Age' was found asthe most important feature for dataset-1 and in respect of dataset-2, 'Weight' was considered as the most important feature.

Dealing with Missing Values: No missing values have been found on any of the dataset. However, if future missing values are encountered, the following approaches can be applied:

- The missing values can be replaced with mean or mode or median value if the data are continuous and for categorical data, missing values can be replaced by the most frequent value.
- The features or rows with too much missing values can be removed, depending on their impact on the dataset.

Dealing with Imbalanced Dataset: Here both of the datasets were imbalanced. For both of the datasets, oversampling technique Synthetic Minority Over-sampling

Technique (SMOTE) was used to handle imbalance issue. After balancing the datasets, the proposed approach was further applied. The performance of the proposed model had been improved significantly after applying SMOTE approach on both datasets as compared to the performance on the imbalanced datasets.

3.3.3 Model Selection and Training:

After preprocessing, the datasets were split, and a pool of nine Synthetic Minority Over-sampling Technique classifiers was created for obesity risk prediction. A total of 50 hyperparameter configurations were explored, as detailed in the following tables. The search space included regularization strengths for LR, neighbor counts for KNN, kernel parameters for SVC, and estimator counts with sampling strategies for tree-based and boosting methods. These settings were informed by standard practices and preliminary trials. A grid-styled sweep was applied on the training data to reduce overfitting and ensure balanced performance. This tuning process identified the strongest base learners and provided the diversity required for constructing robust ensemble models.

Dataset Splitting and Model Selection Scheme: Both of the datasets were split into 60:20:20 (train : validation : test) ratios. That means 60 percent of the datasets were taken in training purpose, 20 percent of the datasets were taken in both validation and testing purpose. For model selection scheme, 10-fold stratified cross validation was used on both of the datasets during training to evaluate the model's performance.

Algorithms Selected for Machine Learning Pool with Hyper-parameters: In this project, a diverse set of algorithms was chosen to include both simple and complex models for creating the ML pool. For this experiment, nine different ML algorithms with in total 50 hyper-parameters setting were applied on both datasets to predict the obesity problem. The detailed representation of nine ML models with 50 hyper-parameters setting were discussed in the following section.

Logistic Regression (LR): LR is a straightforward linear model that works well for multiclass classification [19]. In this experiment, in total 6 hyper parameters of LR were used on both of the datasets for the prediction of obesity. The detailed hyper-parameters setting is represented in Table 6.

Table 6: Logistic Regression – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
Penalty	l2, none
C	0.5, 0.75, 0.8, 1.0
Multi-class	default, ovr

K-Nearest Neighbors (KNN): KNN is a classification approach that uses the majority vote of neighbors. Variations are found in k, algorithms, and metrics provide flexibility in K decision boundaries [20]. In this experiment, in total 6 hyper-parameters of KNN were applied on both of the datasets for the prediction of obesity. The details of hyper-parameters of KNN that were utilized in this experiment are shown in Table 7.

Table 7: K-Nearest Neighbor – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
N_neighbors	2, 3, 5, 6, 9, 10
Algorithm	Kd_tree, brute, ball_tree
P - value	7, 10
Metric	Manhattan, cosine, euclidean, minkowski

Naïve Bayes (NB): NB classifiers presume feature independence. The Gaussian version is used for continuous data, whereas the Bernoulli version is for binary feature sets [21]. In this experiment, both of the Gaussian version (NB1) and Bernoulli version (NB2) were used on both of the datasets for predicting obesity problem.

Decision Tree (DT): DT are nonlinear models that learn splits depending on features [22]. In this experiment, in total 5 hyper-parameters of DT were utilized on both of the datasets to make the prediction. The details of hyper-parameters of DT that were utilized are shown in Table 8.

Table 8: Decision Tree – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
criterion	gini, entropy, log_loss
splitter	random
max_depth	6, 8
min_samples_leaf	2, 3, 4
max_features	sqrt, log2, none
min_impurity_decrease	0.0, 0.01, 0.001
random_state	0

Random Forest (RF): RF is an ensemble of decision trees used for classification, with models varying in tree depth, number of estimators, feature selection approach, and whether bootstrapping is employed [23]. In this experiment, in total 8 hyper-parameters of RF were applied on both of the datasets to measure the performance of prediction of obesity problem. The details of the hyper-parameters of RF that were employed in this experiment are stated in Table 9.

Table 9: Random Forest – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
n_estimators	100, 500, 1000
criterion	entropy, log_loss, gini
max_features	log2, sqrt, none
bootstrap	False
random_state	0

Gradient Boosting (GB): GB is an ensemble approach that builds models sequentially and focuses on correcting previous model errors, with variations in loss functions, estimators, and criteria for splitting [24]. In this project, in total 8 hyper-parameters of GB were used on both of the datasets to predict the obesity. The details of hyper-parameters of GB that were applied in this project are represented in Table 10.

Table 10: Gradient Boosting – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
n_estimators	400, 500, 800, 1000, 1100, 1200
loss	log_loss
criterion	squared_error, friedman_mse
min_samples_split	2, 3, 5
min_impurity_decrease	0.01, 0.001, 0.0001
random_state	0

AdaBoost: AdaBoost is an ensemble technique that iteratively re-weights misclassified data points and modifies estimators in order to combine weak classifiers to create a strong model [25]. In this project, in total 5 hyper-parameters of AdaBoost were applied on both of the datasets to make the prediction of the obesity problem. The details of hyper-parameters of Adaboost that were applied in this project are illustrated in Table 11.

Table 11: AdaBoost – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
n_estimators	200, 500, 1000, 1200
algorithm	SAMME, SAMME.R
learning_rate	2
random_state	0

Support Vector Classifiers (SVC): Hyperplanes are used by the SVC to divide data into classes. To modify model complexity, it can employ several kernels (linear, RBF) and parameters, including the regularization parameter C [26]. In this project, 5 hyper-parameters of SVC were used on both of the datasets for predicting the obesity. The details of hyper-parameters of SVC that were employed in this project are represented in Table 12.

Table 12: Support Vector Classifier – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
probability	True
C	0.5, 0.75, 1.0, 1.25
kernel	linear
random_state	0

Multi-Layer Perceptron (MLP): MLP is a type of neural network model with fully connected layers where the configurations differ in the number of iterations, learning rate, and network architecture [27]. In this project, 5 different hyper-parameters of MLP were applied on both of the datasets to predict the obesity problem. The details of hyper-parameters of MLP that were used in this project are illustrated in Table 13.

Table 13: Multi-Layer Perceptron – Classification Hyper-parameter Combination

Hyper-parameter	**Hyper-parameter Values**
max_iter	200, 500, 1000
hidden_layer_sizes	(100, 100)
learning_rate	adaptive, invscaling
random_state	0

3.3.4 Ensemble Model Construction

To leverage the strengths of the best-performing classifiers, ensemble models were constructed from the selected base learners. Two ensemble learning strategies were implemented in this study.

Hybrid Majority Voting: The first approach, Hybrid Majority Voting, combined the predictions of the top three classifiers using both Majority Hard Voting and Weighted Hard Voting schemes. In Majority Hard Voting, the final class label was determined by the majority decision among the base learners, while in Weighted Hard Voting, classifiers were assigned weights proportional to their validation performance to improve prediction reliability [6].

Ensemble Stacking: The second approach, Ensemble Stacking, utilized a Multi-Layer Perceptron as the meta-classifier. In this framework, the predictions from the base learners were used as input features, and the meta-classifier was trained to capture higher-level patterns and interactions between them. By integrating diverse base learners, these ensemble methods aimed to enhance generalization, reduce variance, and provide a more accurate and robust model for obesity risk prediction [7].

3.3.5 Evaluation Metrics

To measure the performance of proposed models, the following metrics are considered.

ROC_AUC: It is mainly used for binary, multiclass, and multi-label classification problem. Since the given problems were multiclass problem, this metric might be a good option to apply. It is a probabilistic approach [28].

Average Precision Score: It is also a probabilistic approach. Since it focuses on the positive class, providing a better evaluation when one class is rare, this metric could also be a good option to apply to given multiclass problems [29].

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives. It indicates how many of the predicted positive instances were correct [30].

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

Recall: Recall is the ratio of correctly predicted positive observations to all actual positive observations. It measures the ability of a model to capture all relevant instances [31].

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

F1 Score: F1 score could be a good measure for the given imbalanced data sets that focus on the performance of minority classes [32].

$$F1\text{-Score} = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{3}$$

Accuracy: After applying SMOTE approach, this metric also might be a good option to apply for the given multiclass obesity prediction problem [33].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

In this experiment, `ROC_AUC` Score, Average Precision Score, Accuracy, and F1 Score metrics were used to measure the performance of the ML pool algorithms. Since, `ROC_AUC` Score and Average Precision Score metric work on predicting the probabilities, these metrics were not applicable for the hybrid majority voting approach (ensemble model). As a result, for majority hard voting and ensemble stacking, Accuracy, and F1 Score metrics were applied to measure the performance of the final model.

4 Results and Evaluation

This section presents the comparative result analysis of different ML models, which are visualized using tables and confusion matrices.

4.1 Model Performance Analysis

The best three top performing models with specific hyper-parameters were selected based on 10-fold stratified cross validation for the further test set evaluation for both datasets.

Performance on Dataset 1: The summary of each ML model with its' best hyper-parameter performance on dataset-1 based on the `ROC_AUC` Score, Average Precision Score, Accuracy, and F1 Score metrics is represented in Table 14.

Table 14: Performance Analysis for Each Model (Dataset-1)

Model Name	**ROC_AUC**	**Average Precision**	**Accuracy**	**F1**
Random Forest (RF6)	0.972905	0.938594	0.875018	0.869964
Gradient Boosting (GB5)	0.953547	0.874677	0.822244	0.811672
Multi-Layer Perceptron (MLP5)	0.941460	0.834402	0.790468	0.741200
Support Vector Classifier (SVC1)	0.933948	0.822115	0.781129	0.701566
Logistic Regression (LR2)	0.916110	0.767217	0.734563	0.664019
K-Nearest Neighbor (KNN)	0.908061	0.743446	0.753101	0.721543
Decision Tree (DT4)	0.878490	0.637156	0.685641	0.563693
Naïve Bayes (NB2)	0.875353	0.657063	0.676290	0.555421
AdaBoost (AdaBoost2)	0.861153	0.621522	0.669368	0.580382

According to the performance on the training and validation data of dataset-1, the top three models namely RF6, GB5, and MLP5 were selected to create new ensemble models using majority and weighted hard voting and ensemble stacking. These models were selected because of demonstrating strong generalization ability as well as their optimized hyper-parameters effectively balanced complexity and performance, making them superior among the pool of 9 models with 50 hyper-parameter settings.

Performance on Dataset 2: The summary of each ML model with its' best hyper-parameter performance on dataset-2 based on the `ROC_AUC` Score, Average Precision Score, Accuracy, and F1 Score metrics is represented in Table 15.

Table 15: Performance Analysis for Each Model (Dataset-2)

Model Name	**ROC_AUC**	**Average Precision**	**Accuracy**	**F1**
Gradient Boosting (GB3)	0.997926	0.989671	0.965649	0.964740
Random Forest (RF7)	0.997473	0.984930	0.952610	0.951769
Support Vector Classifier (SVC5)	0.996872	0.981405	0.947873	0.945678
Multi-Layer Perceptron (MLP5)	0.994484	0.968530	0.937204	0.935054
Logistic Regression (LR2)	0.994145	0.973653	0.957344	0.956506
Decision Tree (DT5)	0.976979	0.926502	0.934224	0.932592
K-Nearest Neighbor (6NN)	0.954322	0.834497	0.795611	0.779721
Naïve Bayes (NB1)	0.910193	0.679336	0.611391	0.578667
AdaBoost (AdaBoost2)	0.862743	0.456811	0.452599	0.382094

According to the performance on the training data of Dataset-2, the top three models namely GB3, RF7, and SVC5 were selected to create new ensemble models using majority and weighted hard voting and ensemble stacking. These models were selected due to their ability to handle feature interactions and class imbalances effectively. They were the best options out of the 50 models for dataset-2 since their hyper-parameters were adjusted to strike a balance between generalization and model complexity.

The differences in top-performing models across the two obesity datasets are caused by variations in data distribution, feature correlations, and noise levels. RF6, GB5, and MLP5 excelled in the first dataset because of their likely higher feature complexity or nonlinear patterns, on the other hand, GB3, RF7, and SVC5 performed better in the second dataset due to their adaptability to differences in class separability or feature importance.

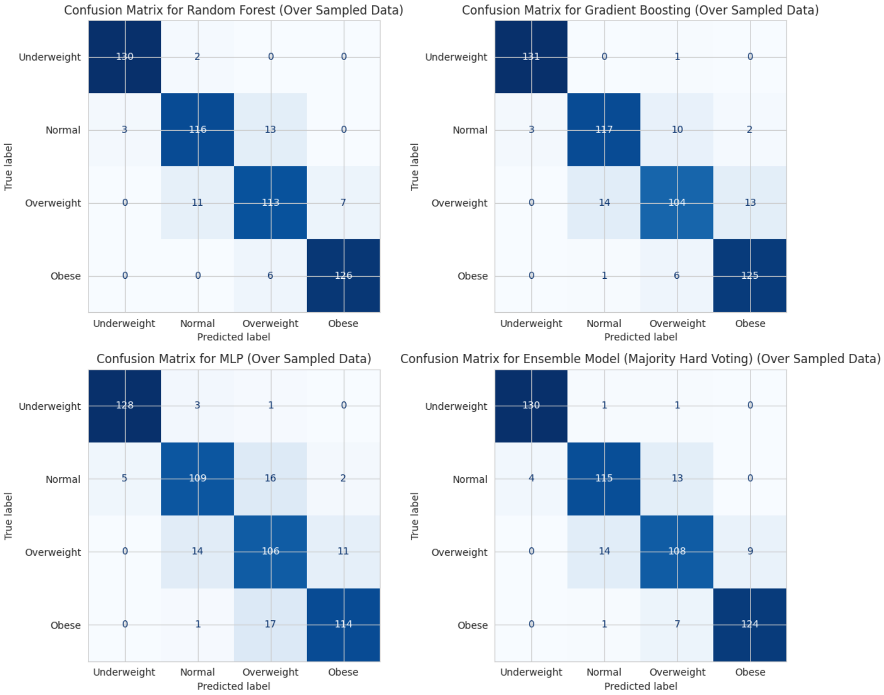
4.2 Test Set Evaluation

This section represents the test set evaluation of the proposed model on both of the datasets and demonstrates the strength and weekness of the proposed model.

Performance on Dataset 1: For Dataset-1, the top three performing models from the ML pool were found as RF6, GB5, and MLP5. Then these three models were combined to make ensemble models using majority hard voting and weighted hard voting. Finally, ensemble stacking model was created by aggregating RF6, GB5, and MLP5 as the base models and MLP as the final model. Since the dataset-1 was an imbalanced dataset, later SMOTE approach was applied to make it as a balanced dataset. Then all of the individual models and ensemble models were applied on the entire training dataset and evaluated on the test dataset. Finally, a comparative analysis was given among all of the individual and proposed models to show the strengths, and the weakness of the proposed model. A detailed comparison among the base models and the proposed models is represented in Table 16.

Table 16: Comparative analysis of proposed model on Dataset 1

Model Name	**Accuracy**	**F1 Score**
Random Forest (RF6)	0.920304	0.920070
Gradient Boosting (GB5)	0.905123	0.903960
Multi-Layer Perceptron (MLP5)	0.867173	0.867583
Majority Hard Voting	0.905123	0.904647
Weighted Hard Voting	0.920304	0.920070
Ensemble Stacking (MLP)	0.920304	0.920010



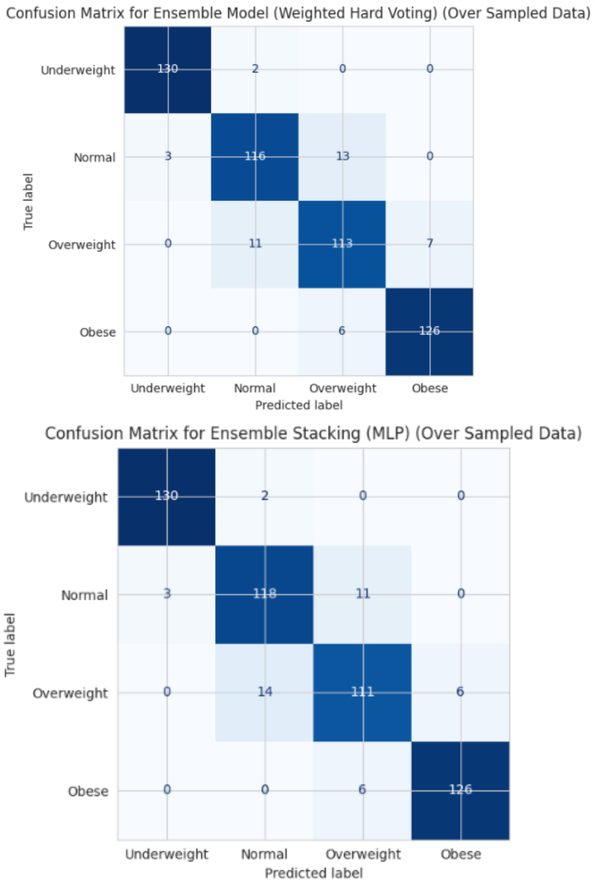


Fig. 2 Confusion matrices of baseline models (Random Forest, Gradient Boosting, MLP) and ensemble models (Majority Hard Voting, Weighted Hard Voting, and Ensemble Stacking) for obesity risk prediction on Dataset 1.

The comparative analysis between the proposed ensemble models and the base classifiers highlights several important findings. Overall, the dataset preparation significantly improved the performance of all individual models compared to the earlier setting.

For Majority Hard Voting, the ensemble achieved better results than Gradient Boosting and MLP across both Accuracy and F1 Score; however, it remained slightly inferior to Random Forest. The confusion matrix further confirms this observation, showing that Majority Hard Voting struggles in predicting the Overweight class more than RF, which explains the marginally lower overall performance.

In contrast, Weighted Hard Voting shows a substantial improvement over both Gradient Boosting and MLP, surpassing them consistently in both metrics. Interestingly, this ensemble achieves exactly the same performance as RF in terms of Accuracy and F1 Score. The confusion matrix supports this result, as Weighted Hard Voting demonstrates nearly identical predictive strength across all classes compared to RF, while maintaining an edge over Gradient Boosting and MLP.

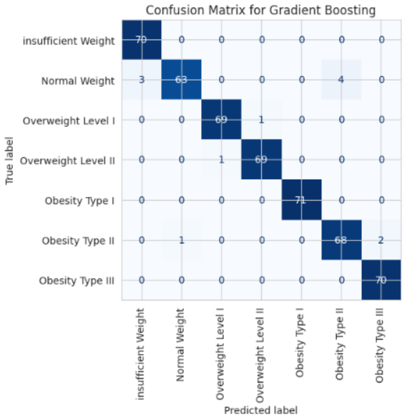
Finally, Ensemble Stacking also performs strongly, outperforming Gradient Boosting and MLP across both performance measures. Compared to RF, Ensemble Stacking achieves the same Accuracy but falls just slightly behind in F1 Score. The confusion matrix reveals that this minor drop is primarily due to its slightly weaker ability to predict the Overweight class, even though its performance on the other classes remains competitive.

Taken together, these results demonstrate that while RF remains a highly robust baseline, the proposed ensemble methods—particularly Weighted Hard Voting and Ensemble Stacking—provide competitive and, in some cases, equivalent performance. This confirms that ensemble strategies can effectively leverage the diversity of multiple classifiers to achieve strong and reliable results for obesity risk prediction.

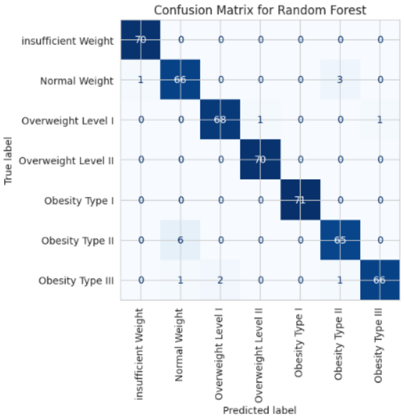
Performance on Dataset 2: For dataset-2, the top three performing models from the ML pool were found GB3, RF7, and SVC5. Then these three models were combined to make ensemble models using majority hard voting and weighted hard voting. Finally, ensemble stacking model was created by aggregating GB3, RF7, and SVC5 as the base models and multi-layer perceptron MLP as the final model. Since the dataset-2 was also an imbalanced dataset, later SMOTE approach was applied to make it as a balanced dataset. Then all of the individual models and ensemble models were applied on the entire training dataset (balanced) and evaluated on the test dataset (balanced). Finally, a comparative analysis was given among all of the individual and proposed models to show the strengths, and the weakness of the proposed model. A detailed comparison for among the base models and the proposed models is represented in Table 17.

Table 17: Performance Analysis for proposed models on Dataset 2

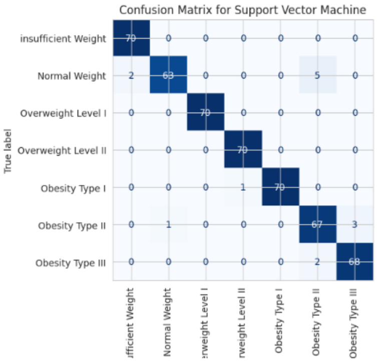
Model Name	**Accuracy**	**F1 Score**
Random Forest (RF7)	0.967480	0.967485
Gradient Boosting (GB3)	0.975610	0.975388
Support Vector Classifier (SVC5)	0.971545	0.971530
Majority Hard Voting	0.981707	0.981675
Weighted Hard Voting	0.975610	0.975388
Ensemble Stacking (MLP)	0.989837	0.989825



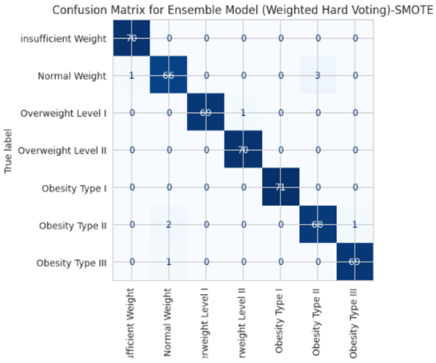
Confusion Matrix for Gradient Boosting



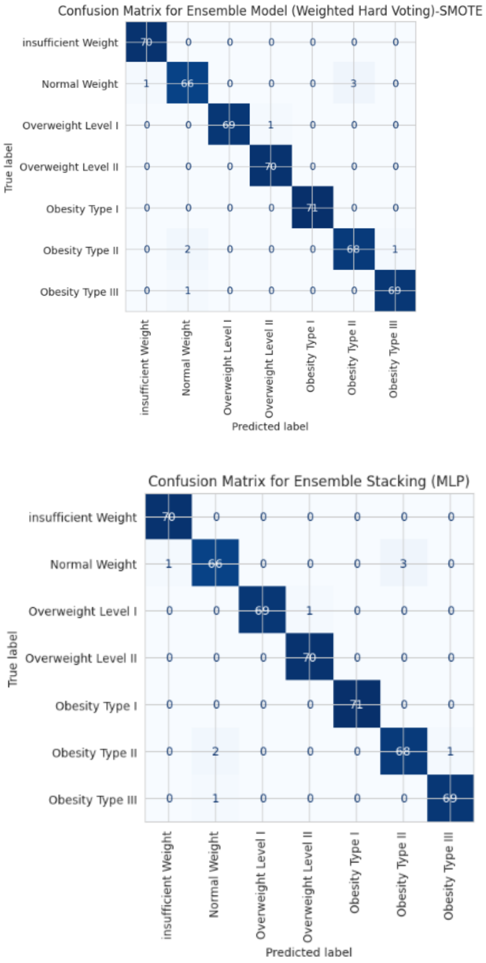
Confusion Matrix for Random Forest



Confusion Matrix for Support Vector Machine



Confusion Matrix for Ensemble Model (Weighted Hard Voting)-SMOTE



Confusion Matrix for Ensemble Model (Weighted Hard Voting)-SMOTE

	insufficient Weight	Normal Weight	Overweight Level I	Overweight Level II	Obesity Type I	Obesity Type II	Obesity Type III
insufficient Weight	70	0	0	0	0	0	0
Normal Weight	1	66	0	0	0	3	0
Overweight Level I	0	0	69	1	0	0	0
Overweight Level II	0	0	0	70	0	0	0
Obesity Type I	0	0	0	0	71	0	0
Obesity Type II	0	2	0	0	0	68	1
Obesity Type III	0	1	0	0	0	0	69

Confusion Matrix for Ensemble Stacking (MLP)

	insufficient Weight	Normal Weight	Overweight Level I	Overweight Level II	Obesity Type I	Obesity Type II	Obesity Type III
insufficient Weight	70	0	0	0	0	0	0
Normal Weight	1	66	0	0	0	3	0
Overweight Level I	0	0	69	1	0	0	0
Overweight Level II	0	0	0	70	0	0	0
Obesity Type I	0	0	0	0	71	0	0
Obesity Type II	0	2	0	0	0	68	1
Obesity Type III	0	1	0	0	0	0	69

Fig. 3 Confusion matrices of baseline models (Random Forest, Gradient Boosting, SVC) and ensemble models (Majority Hard Voting, Weighted Hard Voting, and Ensemble Stacking) for obesity risk prediction on Dataset 2.

The comparative analysis between the proposed ensemble models and the base classifiers highlights several important findings. Overall, dataset preparation led to a notable improvement in the performance of all individual models compared to the earlier setting.

For Majority Hard Voting, the results show that the ensemble clearly outperforms all of its constituent base models, namely RF, Gradient Boosting, and SVC, across both Accuracy and F1 Score. This indicates that combining the predictions through majority voting significantly boosts predictive performance beyond what the individual models can achieve. The confusion matrix further supports this observation, where Majority Hard Voting demonstrates superior predictive capability overall, although it still struggles with misclassifications in the Normal Weight and Obesity Type II categories. Addressing these weaknesses could potentially raise the model's performance even further.

In the case of Weighted Hard Voting, the model performs better than RF and SVC in both metrics and matches Gradient Boosting in terms of Accuracy and F1 Score. However, its performance remains lower than Majority Hard Voting, suggesting that the weighting scheme did not provide additional benefit beyond simple majority aggregation in this setting. The confusion matrix confirms this finding, showing that the predictive behavior of Weighted Hard Voting closely resembles that of Gradient Boosting, achieving stronger class-level accuracy than RF and SVC but still trailing behind Majority Hard Voting.

Finally, Ensemble Stacking achieves the best overall performance, surpassing all individual models as well as the other ensemble methods. With an Accuracy of 0.989837 and an F1 Score of 0.989825, Ensemble Stacking delivers the highest results among all tested approaches. The confusion matrix analysis reinforces this outcome, as Ensemble Stacking addresses several of the weaknesses observed in the base models, particularly improving the classification of the Normal Weight, Obesity Type II, and Obesity Type III categories where Gradient Boosting, RF, and SVC struggled. By effectively learning how to combine the strengths of its base models through a meta-classifier, Ensemble Stacking demonstrates its ability to achieve more balanced and robust predictions across all classes.

Taken together, these findings confirm that while base classifiers such as RF and Gradient Boosting are individually strong, ensemble methods—especially Majority Hard Voting and Ensemble Stacking—consistently enhance predictive performance. Among them, Ensemble Stacking emerges as the most effective approach for obesity risk prediction, providing the highest accuracy and F1 Score as well as improved class-wise predictions.

5 Discussion

In this study we have demonstrated that ensemble learning methods—particularly Weighted Hard Voting and Ensemble Stacking—consistently outperformed individual base learners in the task of obesity risk prediction from structured tabular data. On Dataset-1, Weighted Hard Voting, Ensemble Stacking, and RF achieved nearly identical performance, clearly surpassing Gradient Boosting and MLP. These results indicate that RF remains a strong individual model, but ensemble-based strategies add robustness and help to reduce variance across predictions. On Dataset-2, Ensemble Stacking with a MLP meta-classifier achieved the best performance with an accuracy of 98.98% and an F1 Score of 98.98%, outperforming not only individual models such as RF (96.74%) and SVC (97.15%) but also Majority Hard Voting (98.17%) and Weighted Hard Voting (97.56%). These findings directly address our research objectives by confirming that (1) RF and Gradient Boosting are strong individual predictors, but (2) ensemble strategies, especially Weighted Hard Voting and Stacking, provide statistically meaningful gains in discriminative performance.

Analysis of the confusion matrices provided further insights into class-level prediction behavior. For Dataset-1, the best models struggled particularly with the Overweight, Obese, and Normal classes, leading to some degree of misclassification. Similarly, for Dataset-2, Ensemble Stacking, while the most accurate overall, showed limitations in correctly identifying Obesity Type II and Normal Weight classes. These misclassifications highlight the inherent challenge in differentiating between closely related obesity categories. Nevertheless, both Weighted Hard Voting and Ensemble Stacking consistently reduced false positives and false negatives relative to weaker base classifiers, which is critical in medical and healthcare applications where the cost of misclassification can be significant.

The errors observed in both datasets point toward promising future directions. For Dataset-1, performance can be further improved by analyzing which features contribute most to the misclassification of the overweight, obese, and normal classes. Model explainability tools such as SHapley Additive exPlanations (SHAP) or Local Interpretable Model-agnostic Explanations (LIME) can be applied to identify misleading or weak predictors. Additionally, collecting more data for underrepresented classes and experimenting with deeper stacking frameworks that integrate multiple algorithms could lead to further gains. For Dataset-2, similar improvements can be made by focusing on the misclassified Obesity Type II and Normal Weight classes, applying feature-attribution methods to refine the feature space, and gathering additional representative samples. Stacking multiple base learners in more sophisticated ensemble architectures also holds potential to further improve the model's already high performance.

Together, these results demonstrate that hybrid ensemble approaches—specifically Weighted Hard Voting and Ensemble Stacking—leverage the complementary decision boundaries of multiple classifiers, which reduces overfitting and variance inherent to single models. Importantly, the methodological pipeline, which included careful preprocessing, stratified oversampling, extensive hyperparameter optimization across nine classifiers and fifty parameter configurations, and the exploration of advanced ensemble strategies, provided the foundation for these improvements. The overall findings confirm that ensemble learning is a powerful and practical approach for obesity prediction and offers significant promise for deployment in healthcare analytics.

6 Conclusions and Future Works

This study presented a comparative analysis of hybrid majority voting and ensemble stacking methods for obesity prediction. In hybrid majority voting, both Majority Hard Voting and Weighted Hard Voting were implemented, while Ensemble Stacking employed a Multi-Layer Perceptron as the meta-classifier. On Dataset-1, Weighted Hard Voting and Ensemble Stacking achieved nearly identical performance, both outperforming Majority Hard Voting. On Dataset-2, Ensemble Stacking delivered the best results with an accuracy of 98.98% and an F1 Score of 98.98%, surpassing both Majority Hard Voting and Weighted Hard Voting. These findings demonstrate that ensemble strategies, particularly Ensemble Stacking, provide superior predictive capability compared to individual classifiers and hybrid voting schemes. Looking ahead, several approaches could further enhance performance. Explainable AI techniques such as SHAP or LIME may help identify features contributing to misclassification. Collecting additional data for underrepresented classes could improve class balance, while exploring more advanced ensemble modeling techniques has the potential to yield even stronger results. Together, these directions highlight the promise of ensemble learning in advancing accurate and reliable obesity risk prediction within healthcare analytics.

References

[1] Solomon, D.D., Khan, S., Garg, S., Gupta, G., Almjally, A., Alabduallah, B.I., Alsagri, H.S., Ibrahim, M.M., Abdallah, A.M.A.: Hybrid majority voting: Prediction and classification model for obesity. Diagnostics **13**(15), 2610 (2023)

[2] Pinar, A., Yagin, F.H., Georgian, B.: Use of logistic regression method in predicting obesity levels with machine learning method. Journal of Exercise Science & Physical Activity Reviews **2**(1), 104–113 (2024)

[3] Lin, X., Li, H.: Obesity: epidemiology, pathophysiology, and therapeutics. Frontiers in endocrinology **12**, 706978 (2021)

[4] Jeon, J., Lee, S., Oh, C.: Age-specific risk factors for the prediction of obesity using a machine learning approach. Frontiers in Public Health **10**, 998782 (2023)

[5] Liu, L., Wei, W., Chow, K.-H., Loper, M., Gursoy, E., Truex, S., Wu, Y.: Deep neural network ensembles against deception: Ensemble diversity, accuracy and robustness. In: 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), pp. 274–282 (2019). IEEE

[6] Leon, F., Floria, S.-A., Bădică, C.: Evaluating the effect of voting methods on ensemble-based classification. In: 2017 IEEE International Conference on

INnovations in Intelligent Systems and Applications (INISTA), pp. 1–6 (2017). IEEE

[7] Dey, R., Mathur, R.: Ensemble learning method using stacking with base learner, a comparison. In: International Conference on Data Analytics and Insights, pp. 159–169 (2023). Springer

[8] Koklu, N., Sulak, S.A.: Using artificial intelligence techniques for the analysis of obesity status according to the individuals' social and physical activities. Sinop Üniversitesi Fen Bilimleri Dergisi **9**(1), 217–239 (2024) https://doi.org/10.33484/sinopfbd.1445215

[9] pymche: Machine-Learning-Obesity-Classification. https://github.com/pymche/Machine-Learning-Obesity-Classification. GitHub repository, accessed August 26, 2025 (2020)

[10] Dutta, R.R., Mukherjee, I., Chakraborty, C.: Obesity disease risk prediction using machine learning. International Journal of Data Science and Analytics, 1–10 (2024)

[11] Talari, P., N, B., Kaur, G., Alshahrani, H., Al Reshan, M.S., Sulaiman, A., Shaikh, A.: Hybrid feature selection and classification technique for early prediction and severity of diabetes type 2. Plos one **19**(1), 0292100 (2024)

[12] Musa, F., Basaky, F., *et al.*: Obesity prediction using machine learning techniques. Journal of Applied Artificial Intelligence **3**(1), 24–33 (2022)

[13] Rodríguez, E., Rodríguez, E., Nascimento, L., Silva, A.F., Marins, F.A.S.: Machine learning techniques to predict overweight or obesity. In: IDDM, pp. 190–204 (2021)

[14] Jindal, K., Baliyan, N., Rana, P.S.: Obesity prediction using ensemble machine learning approaches. In: Recent Findings in Intelligent Computing Techniques: Proceedings of the 5th ICACNI 2017, Volume 2, pp. 355–362. Springer, ??? (2018)

[15] Basili, V.R., Weiss, D.M.: A methodology for collecting valid software engineering data. In: Proceedings of the International Conference on Software Engineering (ICSE), pp. 75–77 (1984)

[16] team, T.: pandas-dev/pandas: Pandas. Zenodo (2020). https://doi.org/10.5281/zenodo.3509134 . https://doi.org/10.5281/zenodo.3509134

[17] Seaborn Developers: Seaborn Documentation — Version 0.13.2. https://seaborn.pydata.org/. Accessed: 2025-08-29 (2025)

[18] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)

[19] Scikit-learn developers: sklearn.linear_model.LogisticRegression — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[20] Scikit-learn developers: sklearn.neighbors.KNeighborsClassifier — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

[21] Scikit-learn developers: 1.9. Naive Bayes — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/naive_bayes.html

[22] Scikit-learn developers: sklearn.tree.DecisionTreeClassifier — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

[23] Scikit-learn developers: sklearn.ensemble.RandomForestClassifier — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[24] Scikit-learn developers: sklearn.ensemble.GradientBoostingClassifier — scikit-learn documentation. Accessed: 27 August 2025 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

[25] Scikit-learn developers: sklearn.ensemble.AdaBoostClassifier — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html

[26] Scikit-learn developers: sklearn.svm.SVC — scikit-learn documentation. Accessed: 27 August 2025 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

[27] Scikit-learn developers: sklearn.neural_network.MLPClassifier — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

[28] Scikit-learn developers: sklearn.metrics.roc_auc_score — scikit-learn 1.5.0 documentation. Accessed: 2025-08-27 (2024). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html

[29] Scikit-learn developers: sklearn.metrics.average_precision_score — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html

[30] Scikit-learn developers: sklearn.metrics.precision_score — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html

[31] Scikit-learn developers: sklearn.metrics.recall_score — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.recall_score.html

[32] Scikit-learn developers: sklearn.metrics.f1_score — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

[33] Scikit-learn developers: sklearn.metrics.accuracy_score — scikit-learn documentation. Accessed: 2025-08-27 (2025). https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html