8. USER MANUAL

Table of Contents

1. Introduction
Welcome to the Ensemble Methods User Manual. This manual provides step-by-step instructions on how to use the provided code to implement ensemble methods for binary classification using the Pima Indians Diabetes dataset. The code utilizes libraries such as pandas, scikit-learn, matplotlib, and seaborn to perform data analysis and modeling.

2. Prerequisites
Before you begin, make sure you have the required libraries installed. You will need:

- pandas
- numpy
- scikit-learn
- matplotlib
- seaborn

3. Getting Started
Follow these steps to get started:
1. Download the Pima Indians Diabetes dataset from the provided link.
2. Save the dataset as "diabetes.csv" in the same directory as the code.
3. Open the Python environment and run the provided code.

4. Understanding the Dataset

- The code loads the dataset, removes rows with missing values, and drops duplicates.
- Basic information about the dataset, including data types and non-null counts, is displayed.
- The first few rows of the dataset and column names are printed.
- The code checks for missing values and displays the count of missing values for each column.
- Descriptive statistics of the dataset are shown.

5. Data Visualization

- Histograms and distribution plots are generated to visualize the distribution of each feature.
- A correlation matrix heat-map is plotted to show the correlations between features.
- Box plot is plotted for understanding the distribution of each feature.

6. Data Pre-processing

- The dataset is split into features (X) and target (y).
- Features are scaled using StandardScaler to standardize data.
- Data is split into training and testing sets using train_test_split.
- Feature selection is performed using SelectKBest and f_classif.

7. Methods

Three types of methods are implemented:

1. Bagging: Combines predictions from multiple models.
2. Boosting: Enhances the performance of a weak base model.
3. Traditional Algorithms: K-NN, Logistic Regression, and Decision Tree classifiers.

8. Using the Code

- The code provides a menu with options to choose the ensemble method or exit.
- For bagging, choose the base estimator (KNN or Logistic Regression) and the number of bags and for k-NN we can choose number of nearest neighbors.
- For Boosting, select the base estimator (Decision Tree or Logistic Regression) and the number of estimators and for decision tree we can decide depth of tree.
- For Traditional Algorithms, choose the model (KNN, Logistic Regression, or Decision Tree) and tune their parameters accordingly.

9. Interpreting Results

- After executing each ensemble method or traditional algorithm, a confusion matrix heat-map is displayed.
- The confusion matrix helps visualize the accuracy and misclassifications.
- The accuracy of the model is also printed as a percentage.

10. Conclusion

You have successfully used the provided code to implement ensemble methods and traditional algorithms. The code allows you to explore different ensemble techniques and compare their performance to traditional algorithms.

Feel free to experiment with different settings, such as the number of bags, estimators, and hyper-parameters, to understand how they affect the results.

For any further assistance or questions, please refer to the documentation of the libraries used.