National University of Computer and Emerging Sciences

Lab Manual 08
CL461-Data Mining Lab

Course Instructor	Eesha Tur Razia Babar
Lab Instructor (s)	Abdul Rehman Mateen Fatima
Section	BDS-6A
Semester	Spring 2024

Lab Task:

“Ensemble” is a technique where multiple models are combined to improve the overall performance of a system. The basic idea is that by combining the predictions of multiple models, the strengths of individual models can compensate for each other's weaknesses, leading to more accurate and robust predictions.

There are several types of ensemble methods, including:

1. Bagging (Bootstrap Aggregating):
   This method involves training multiple instances of the same base learning algorithm on different subsets of the training data (sampled with replacement) and then combining their predictions. Random Forest is a popular example of a bagging ensemble algorithm.

2. Boosting:
   Boosting algorithms iteratively train weak learners (models that are only slightly better than random guessing) and give more weight to misclassified instances in subsequent iterations, focusing on the areas where previous models performed poorly. Examples of boosting algorithms include AdaBoost and Gradient Boosting Machines (GBM).

3. Stacking:
   Stacking, also known as stacked generalization, involves training a meta-model (often a simple model like linear regression) on the predictions of several base models. The meta-model learns to combine the predictions of the base models to make the final prediction.

4. Voting:
   In this, multiple models make predictions independently, and the final prediction is determined by a majority vote (for classification tasks) or averaging (for regression tasks) of the individual predictions.

In this lab, you will learn to perform sentiment analysis on a dataset of Airbnb’s dataset using ensemble methods. You will explore techniques such as bagging, boosting, and stacking to improve the accuracy of sentiment classification.

You have been given some sample codes, you are to use them and make your own code from scratch to perform the following tasks:

1. Data Preprocessing:
   - Load the dataset.
   - Preprocess the text data by removing stopwords, punctuation, and performing tokenization.
   - Convert the text data into numerical features using techniques like TF-IDF or CountVectorizer.

2. Model Training:
- Train individual base models using different algorithms such as Naive Bayes, Decision Trees, and Support Vector Machines (SVM).
- Implement ensemble techniques:
    - Bagging: Train a Random Forest classifier.
    - Boosting: Train an AdaBoost classifier.
    - Stacking: Train a meta-model using predictions from base models.

3. Model Evaluation:
- Evaluate the performance of each base model and ensemble methods using metrics such as accuracy, precision, recall, and F1-score.
- Visualize the performance metrics using plots like confusion matrix and ROC curve.

4. Analysis:
- Compare the performance of individual models with ensemble methods.
- Discuss the strengths and weaknesses of each approach.
- Explore potential improvements or modifications to the ensemble methods.

The goal of Airbnb's marketing team in this exercise was to improve its users' performance so it could reap the benefits of ongoing host and renter fees. If the company's hosts were not happy, they were not likely to continue listing their properties through Airbnb, and in a competitive and burgeoning marketplace, such attrition could be devastating.

Answer the following question:

1. What could the Airbnb marketing team offer to improve its users' experience? Should it rank properties it suggested to users based on some metric such as review sentiment? How would review sentiment compare to summary-rating value in terms of its ability to predict revenues?

2. Given what we know about the performance of properties in Miami and Paris, did Airbnb need a region-specific strategy? Could the company suggest optimal pricing for hosts, or suggest other ways hosts could improve overall earnings?

Make sure to back up your claims with statistical results and plots shown in your code.

<u>Instructions:</u>

1. Implement the above-mentioned algorithms
2. Write a detailed analysis