CSE 6363 MACHINE LEARNING

PROGRAMMING ASSIGNMENT - 1

Student Id: 1002112710

Manasa Vardhini Betha

Professor :
Jesus A. Gonzalez

TABLE OF CONTENTS

1. Data Pre-processing

Preprocessing is a crucial step in the data preparation phase of any machine learning project. It involves transforming raw data into a format suitable for machine learning models. The goal is to enhance the quality of the data and make it ready for analysis, improving the performance of the model.

Data Splitting

The dataset was divided into features (X) and the target variable (y), with 'class' representing the personality type we aimed to predict. Further subdivision into training and testing sets followed, using a standard 80-20 split ratio.

2. KNN Algorithm from scratch with k-fold cross validation

K-fold cross-validation is a prevalent methodology in machine learning for assessing and choosing models. The technique involves partitioning the dataset into k subsets or folds. The model is trained on k-1 of these folds and then evaluated on the remaining fold. This cycle is reiterated k times, with a different fold utilized for evaluation in each iteration, providing robust performance estimation.

K-fold cross-validation is a pivotal approach in machine learning for assessing models and making informed selections. It offers an unbiased evaluation of model performance, aiding in the detection of overfitting and underfitting. The technique is versatile, serving purposes like model selection, hyperparameter tuning, and identifying the most appropriate model for a given dataset. However, it can be computationally demanding for large datasets or complex models, and the choice of k can influence the performance metric. In summary, k-fold cross-validation is an indispensable tool for constructing robust and accurate machine learning models.

Algorithm Steps:

- Partition the dataset into k folds of equal size.
- Take one fold as the validation set and train the model on the other k-1 folds.
- Assess the model's performance using the validation set and record the performance metric.
- Iterate through all folds, using each one as the validation set in turn.
- Compute the mean performance metric across all k folds.

- Select the model with the highest performance metric for subsequent predictions.

Variants of k-fold cross-validation include:

Stratified k-fold cross-validation:

This technique ensures that each fold maintains a comparable class distribution to the original dataset. It is particularly useful for imbalanced datasets.

Leave-one-out cross-validation:

This variation involves using k equal to the number of samples in the dataset, resulting in each sample acting as a validation set once. While it provides an unbiased performance estimate, it is computationally intensive.

Shuffle-split cross-validation:

Here, a random subset of data is chosen for training and testing. This approach is beneficial for large datasets.

3. KNN Algorithm using scikit learn library with k-fold cross validation

Scikit-learn provides a simple and efficient way to implement the K-Nearest Neighbors algorithm. Scikit-learn, often abbreviated as sklearn, is one of the most widely used and comprehensive libraries for machine learning in Python. It provides a variety of tools and functionalities to build, evaluate, and deploy machine learning models efficiently.

The scikit-learn library provides a convenient and efficient way to implement the K-Nearest Neighbors algorithm. By incorporating K-Fold Cross Validation, we ensure a robust evaluation of the model's performance. This approach allows us to assess the accuracy of the model while considering variations in the training and validation data across different folds. K-Fold Cross Validation is a crucial tool for building reliable and accurate machine learning models.

K-Nearest Neighbors (KNN) is a popular and intuitive machine learning algorithm used for classification and regression tasks. It's a versatile algorithm known for its simplicity and effectiveness. KNN works on the principle of finding the K closest data points in the training set

to a given test point and predicting the class of the test point based on the majority class among those K neighbors.

4. Accuracy Comparison

Data Set	Accuracy for knn algorithm from scratch with k-fold cross validation	Accuracy for knn algorithm using scikit learn library with k-fold cross validation
Hayes - Roth	56.92%	59.01%
Car Evaluation	65.81%	82.07%
Breast Cancer	65.71%	72.40%

5. Hypothesis Testing results

Data Set	p-value
Hayes - Roth	0.50051
Car Evaluation	0.00034
Breast Cancer	0.48667

We performed paired t-tests for hypothesis testing

Since p-value > 0.05 for Hayes-roth dataset, we accept Ho (Null hypothesis).

Since p-value < 0.05 for Car evaluation dataset, we reject Ho (Null hypothesis).

Since p-value > 0.05 for Breast-cancer dataset, we accept Ho (Null hypothesis).

6. References

- https://machinelearningmastery.com/k-fold-cross-validation/
- https://archive.ics.uci.edu/ml/datasets/Hayes-Roth
- https://archive.ics.uci.edu/ml/datasets/Breast+Cancer
- https://archive.ics.uci.edu/ml/datasets/Car+Evaluation