k-fold cross-validation

Group Members:

1. Ruben Chevez
2. Kratika Naskulwar

The complete source code for this report can be found in the following link:
https://colab.research.google.com/drive/1y8zfz3esy0x5Pf4KZkQZxHdgZz6-GqdE

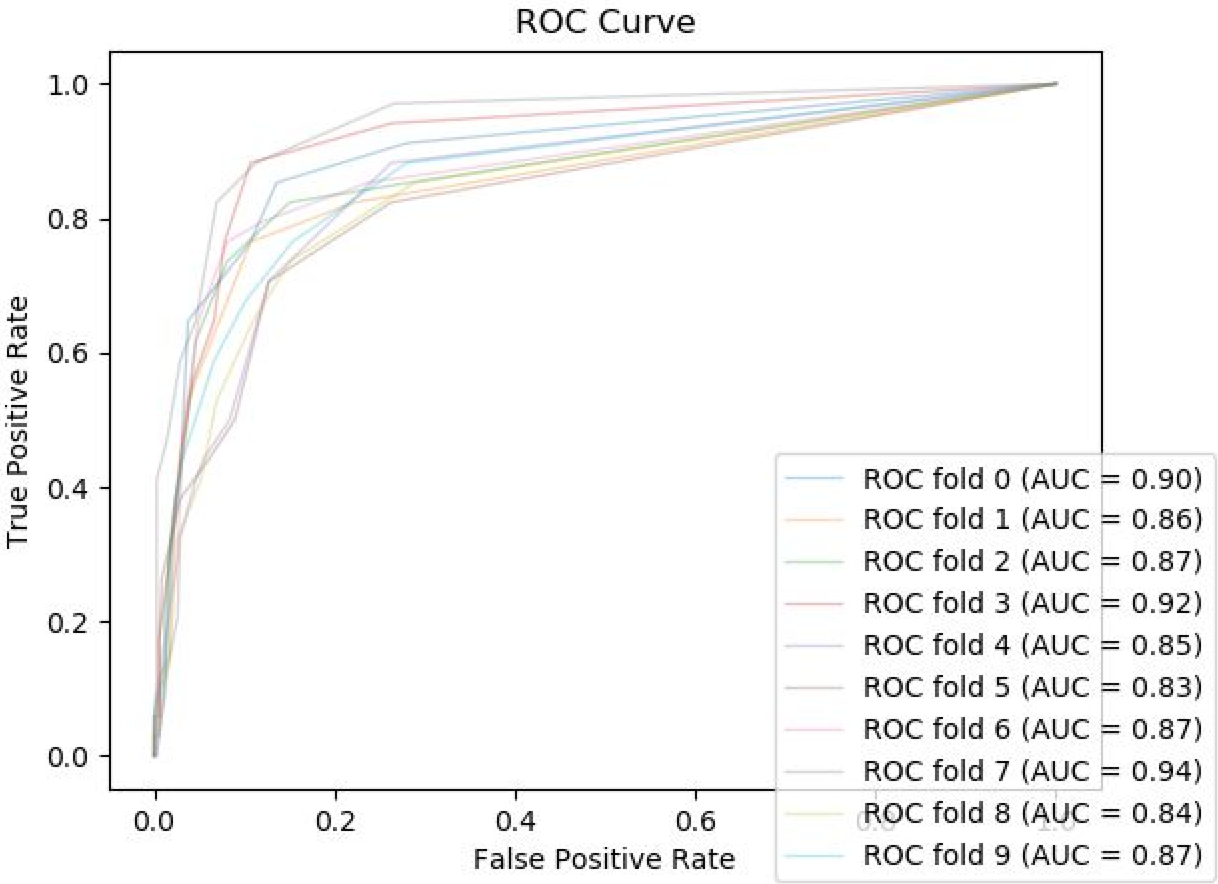
In this document, we describe our steps taken to implement a k-fold cross-validation algorithm from scratch.

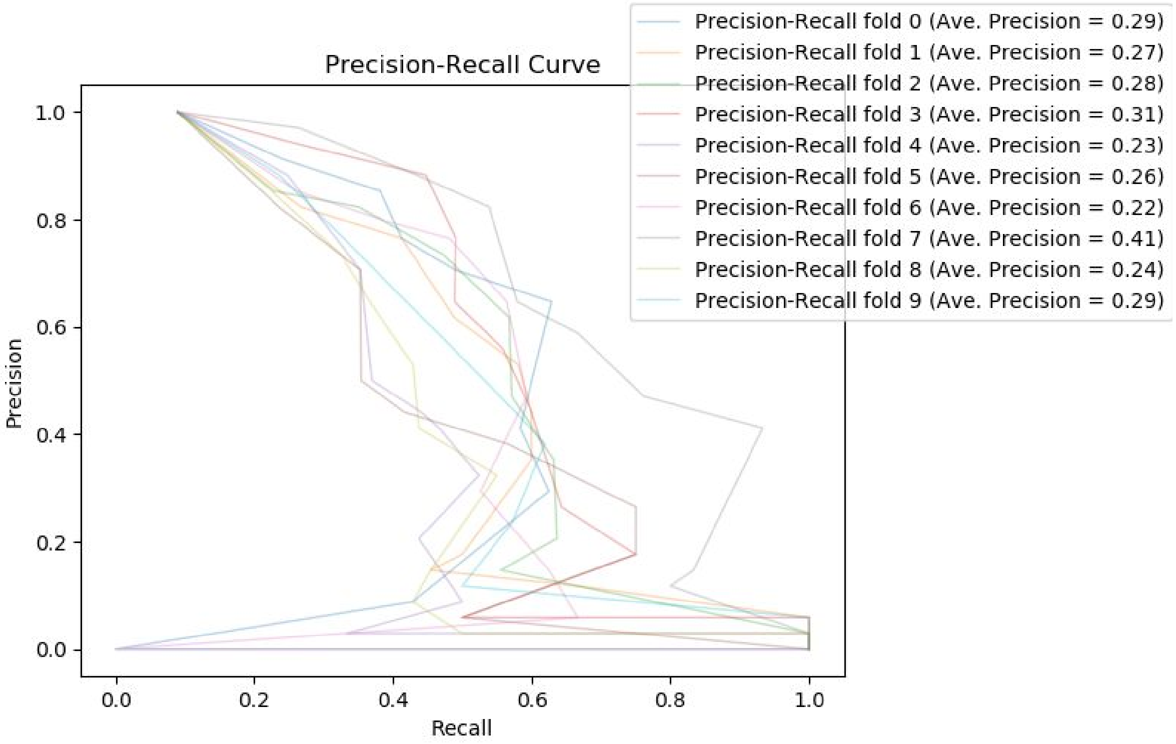
The steps are the following:

0. *run python A2_t2.py A2_t2_dataset.tsv*
1. Declare the hyperparameters: Name of the data file
2. Using the sys library, we obtain the arguments from the command line.
3. Using pandas, read the TSV file, parse it with \t and convert to numpy array.
   Function Called : **loadDataset(filename)**
4. Order By Variance (Descendent) and Filter the features that don't meet a threshold (0.001)
   Function Called : **OrderByVariance(data)**
5. Separate Positives and Negatives, Create 10 Folds of Each. Join each resulting fold with its counter-part (Example: *positive_folds[0]* + *negative_folds[0]*) To distribute negatives and positives correctly in each fold. (*The data shows that there are fewer positives than negatives*)
   Function Called : **EqualyDistributePositiveAndNegativeAndSplit(data)**
6. Select Best Model Using Cross Validation
   Function Called : **CrossValidation(folds, data, K)**
7. Plot Precision Recall-Curve,and ROC Curve
   Function Called : **PlotCurves(folds, bestModel, data)**

Note:

1. The indexes selected and printed in the console are not the same as the ones in the original database. The function **VarianceThreshold(varianceThreshold).fit_transform** filters the database based on a variance threshold and outputs a new filtered dataset, losing the old indexes. Because of time, we focus on the implementation of the algorithm but this feature can be added later.
2. A table (Features VS K) of all possible models is created, containing the average AUC of all the folds from the possible combinations. You can see it in the file created: *PossibleModelsTable.csv*
3. Tune the Hyperparameters *FoldSize=10*, *K=17* (How many k do you want to test), and *varianceThreshold=0.001* at your convenience.


ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC fold 0 (AUC = 0.90)
ROC fold 1 (AUC = 0.86)
ROC fold 2 (AUC = 0.87)
ROC fold 3 (AUC = 0.92)
ROC fold 4 (AUC = 0.85)
ROC fold 5 (AUC = 0.83)
ROC fold 6 (AUC = 0.87)
ROC fold 7 (AUC = 0.94)
ROC fold 8 (AUC = 0.84)
ROC fold 9 (AUC = 0.87)


Precision-Recall Curve
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0
Precision-Recall fold 0 (Ave. Precision = 0.29)
Precision-Recall fold 1 (Ave. Precision = 0.27)
Precision-Recall fold 2 (Ave. Precision = 0.28)
Precision-Recall fold 3 (Ave. Precision = 0.31)
Precision-Recall fold 4 (Ave. Precision = 0.23)
Precision-Recall fold 5 (Ave. Precision = 0.26)
Precision-Recall fold 6 (Ave. Precision = 0.22)
Precision-Recall fold 7 (Ave. Precision = 0.41)
Precision-Recall fold 8 (Ave. Precision = 0.24)
Precision-Recall fold 9 (Ave. Precision = 0.29)