Assignment 2-Cross-validation & performance curves

Description

For GridSearchCV we used knn as the model instance we want to check the hyperparameters on, params_gridknn as the dictionary object, an accuracy scoring metric and a verbose equal to 1 for a detailed print out. The performance metric we used is Accuracy.

Best hyper-parameter setting

The best param is: {n-neighbours' : 2, 'weights': 'uniform' }

	n_neighbors	weights	Accuracy
0	1	uniform	0.898690
1	1	distance	0.898690
2	2	uniform	0.921106
3	2	distance	0.898690
4	3	uniform	0.912082
5	3	distance	0.912664
6	4	uniform	0.914410
7	4	distance	0.914702
8	5	uniform	0.913537
9	5	distance	0.915575
10	6	uniform	0.917322
11	6	distance	0.915575
12	7	uniform	0.914993
13	7	distance	0.916157
14	8	uniform	0.918195
15	8	distance	0.914702
16	9	uniform	0.915284
17	9	distance	0.915866

the best score is: 0.9211062590975254

Collaborations and Online Sources

Github was used for group collaboration

Specifications

Python version:3.8.6
Libraries used: pandas,sys,matplotlib,scikit-learn

Questions

1) **Did you normalize some of the attributes? Why? If yes, which attribute**

Yes we normalized attributes with a numerical value to a common scale between 0 and 1. We did this to bring all of the values to the same range.

2) **Would it make sense to expand your grid (i.e, explore other values for the hyper-parameters)? Why? If yes, which values would you include**

It would make sense to expand the grid because we have multiple best-parameters. The data is split randomly every time, so there is a different best-parameter when run again. We could include: {'n_neighbours': 8, 'weights': 'distance'}.

3) **Did you use stratified cross-validation? Why**

No we did not use stratified cross-validation as the folds are already representative of the data.

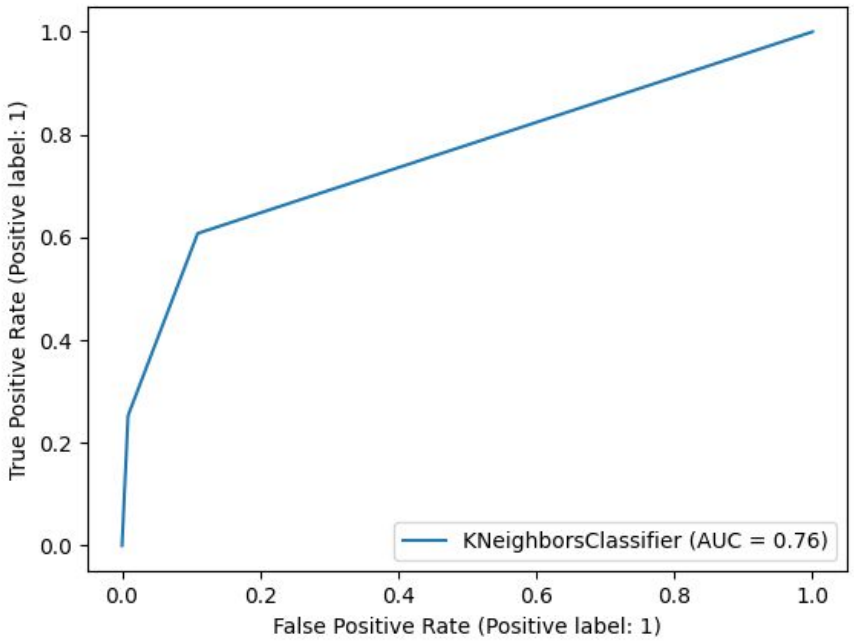
4) **Looking at the ROC and PR curves, where would you recommend to have the threshold for predicting positives (you can indicate the point(s) using coordinations referring to your plots)? Why?**

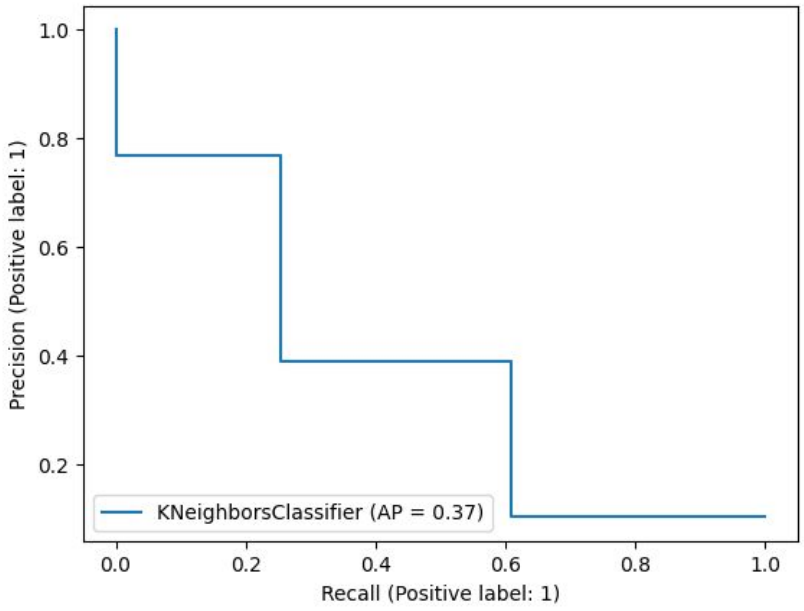
Optimal threshold would be: TPR = 0.60, FPR = 0.11 . This is when the TPR is the highest and the FPR is low on the ROC AUC graph, and Precision and Recall are both high on the PR graph.

PR optimal threshold would be:

5) **Which graphical representation of performance (ROC or PR curve) is more suitable for this task?**

ROC is more suitable because we are interested in the TPR vs FPR. We want to sort our observations and rank the predictions. Also, models rank ROC AUC and accuracy fairly close.


True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)
KNeighborsClassifier (AUC = 0.76)
0.0
0.2
0.4
0.6
0.8
1.0


Precision (Positive label: 1)
Recall (Positive label: 1)
KNeighborsClassifier (AP = 0.37)
0.0
0.2
0.4
0.6
0.8
1.0