**Assignment 3**

Machine Learning, SS2021

|  |  |  |
| --- | --- | --- |
| **Team members** | | |
| Last name | First name | Matriculation Number |
| Rodiga | Luca | 1620972 |
| Binford | Bob | 1600002 |

Task 1)

Dataset 1: {'k': 1}

Test Score: 1.0

Chart, line chart

Description automatically generatedChart, scatter chart

Description automatically generated

Dataset 2: {'k': 1}

Test Score: 0.9953703703703703

Scatter chart

Description automatically generated

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generatedChart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generatedDataset 3: {'k': 3}

Test Score: 0.8867924528301887

1.3)

Choosing a k too small leads to an inefficient and inaccurate clustering performance, where neighbors that matter are not considered. Especially with k = 1, only the nearest neighbor is considered and it’s label is assigned to the test datapoint.

Choosing a k too large leads to inaccurate predictions, because too many data points are considered

Chart, scatter chart

Description automatically generated

Mean score of cross validation: 0.7634585569469291

Test Score for k=1: 0.7222222222222222

Chart, scatter chart

Description automatically generated

Mean score of cross validation: 0.816124031007752

Test Score for k=30: 0.7777777777777778

Chart, scatter chart

Description automatically generated

Mean score of cross validation: 0.7186762075134168

Test Score for k=100: 0.7037037037037037

Chart, line chart

Description automatically generated

The best parameter found by GridSearchCV was {'k': 7} with an accuracy of 0.8269290399522958

Task 3)

1)

We tested a lot of different values for max\_depth and as you can see below, a single best parameter cannot be found. Over more than 5 runs the best max\_depth value varied a lot.

N\_estimators = 1

Chart, scatter chart

Description automatically generatedDataset 1: {'max\_depth': 10}

Test Score: 0.8461538461538461

Dataset 2: {'max\_depth': 100}

Test Score: 0.9212962962962963

Dataset 3: {'max\_depth': 5}

Test Score: 0.8679245283018868

Dataset 1: {'max\_depth': 100}

Test Score: 0.9230769230769231

Dataset 2: {'max\_depth': 100}

Test Score: 0.875

Dataset 3: {'max\_depth': 25}

Chart, scatter chart

Description automatically generatedTest Score: 0.9245283018867925

Dataset 1: {'max\_depth': 5}

Test Score: 0.9230769230769231

Dataset 2: {'max\_depth': 10}

Test Score: 0.8981481481481481

Dataset 3: {'max\_depth': 10}

Test Score: 0.8490566037735849

Dataset 1: {'max\_depth': None}

Test Score: 0.8461538461538461

Dataset 2: {'max\_depth': 25}

Test Score: 0.9074074074074074

Chart, scatter chart

Description automatically generatedDataset 3: {'max\_depth': 10}

Test Score: 0.8679245283018868

Dataset 1: {'max\_depth': 5}

Test Score: 0.8461538461538461

meanCV\_test\_score

[0.79285714 0.89642857 0.86785714 0.86785714 0.84642857 0.89642857]

Dataset 2: {'max\_depth': 25}

Test Score: 0.9583333333333334

meanCV\_test\_score

[0.7464997 0.80993441 0.87790101 0.92892069 0.90573643 0.8887418 ]

Dataset 3: {'max\_depth': None}

Test Score: 0.8301886792452831

meanCV\_test\_score

[0.79758065 0.85383065 0.82883065 0.82862903 0.84112903 0.85403226]

The model performs noticeably better with N\_estimators = 100, but the best max\_depth again varies like in the tests before. As to be seen in the plots of the decision boundaries, the model is less affected by the outliers which leads to the performance increase.

Chart, scatter chart

Description automatically generated

N\_estimators = 100

Dataset 1: {'max\_depth': 2}

Test Score: 0.9230769230769231

Dataset 2: {'max\_depth': 25}

Test Score: 0.9861111111111112

Dataset 3: {'max\_depth': 2}

Test Score: 0.8867924528301887

Scatter chart

Description automatically generatedDataset 1: {'max\_depth': 5}

Test Score: 0.9230769230769231

Dataset 2: {'max\_depth': 25}

Test Score: 0.9861111111111112

Dataset 3: {'max\_depth': 10}

Test Score: 0.8301886792452831

Dataset 1: {'max\_depth': 5}

Test Score: 0.9230769230769231

Dataset 2: {'max\_depth': 25}

Test Score: 0.9861111111111112

Dataset 3: {'max\_depth': 2}

Test Score: 0.8679245283018868

Dataset 1: {'max\_depth': 2}

Chart, scatter chart

Description automatically generatedTest Score: 0.8461538461538461

Dataset 2: {'max\_depth': None}

Test Score: 0.9814814814814815

Dataset 3: {'max\_depth': 10}

Test Score: 0.8867924528301887

Dataset 1: {'max\_depth': 5}

Test Score: 0.9230769230769231

meanCV\_test\_score

[0.975 1. 1. 1. 1. 1. ]

Dataset 2: {'max\_depth': 25}

Test Score: 0.9861111111111112

meanCV\_test\_score

[0.80836017 0.89954681 0.97064997 0.97372689 0.97372689 0.97216458]

Dataset 3: {'max\_depth': 5}

Test Score: 0.8867924528301887

meanCV\_test\_score

[0.86653226 0.91068548 0.89798387 0.89778226 0.89798387 0.89153226]

Task 3.2)

The best parameters found for the RandomForestClassifier were n\_estimators=100, max\_depth=None, with a Test Score: 0.688

Chart, bar chart, histogram

Description automatically generated

The performance of a SVC classifier before the feature reduction with the

Best SVC params: {'C': 10, 'gamma': 0.001, 'kernel': 'rbf'} was a Test Score: 0.7

With the least important features left out the SVC with the same parameter like above, achieved a small improvement in performance with a Test Score: 0.768 and a meanCV\_test\_score: 0.644