CENG499 Assignment3

Uygar Yaşar 2310613

December 2022

1 Part1

```
[]
['Outlook']
['Outlook']
['Outlook', 'Windy']
['Outlook', 'Windy']
['Outlook']
['Outlook', 'Humidity']
['Outlook', 'Humidity']
Training completed
Accuracy : 100.00
```

Figure 1: Decision Tree Output With Used Attributes Printing

As it can be seen from the figure 1, Outlook has most information gain and also when I change the criterion I saw that it has also highest gain ratio. Then, Windy and Humidity separates data into leaves. Since all the dataset separate correctly, accuracy is 100.

2 Part2

I used accuracy as performance metric. For dataset1, I set the C values to 1, 5, 10, and 20 and kernel functions were linear, polynomial, and rbf. So, I get 12 plots. (figure 2-5) For dataset2, I printed all the configurations, their accuracies and confidence intervals (It can be seen from figure 6.)

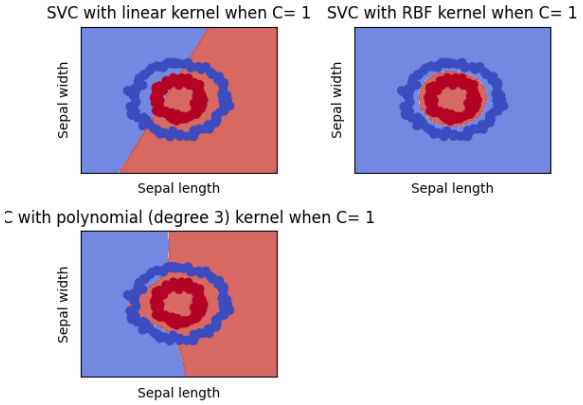

Figure 2: C=1 plots

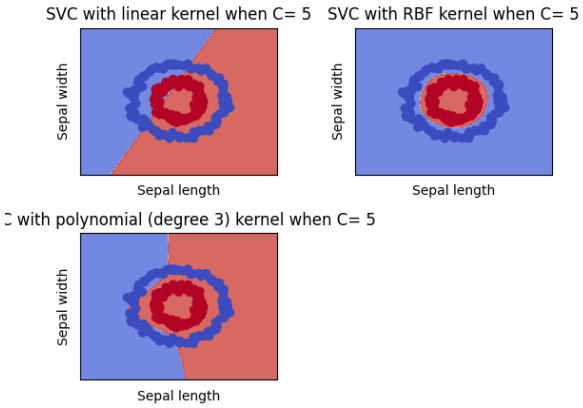

Figure 3: C=5 plots

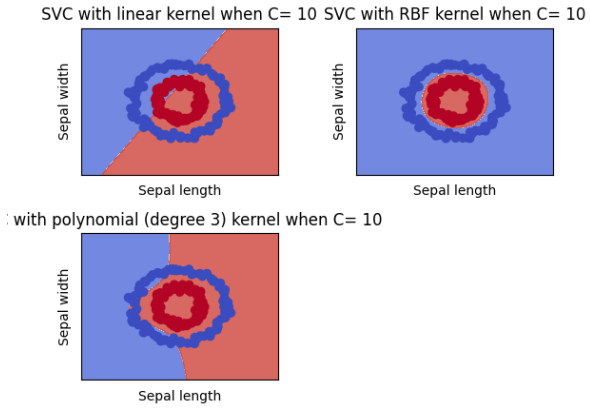

Figure 4: C=10 plots

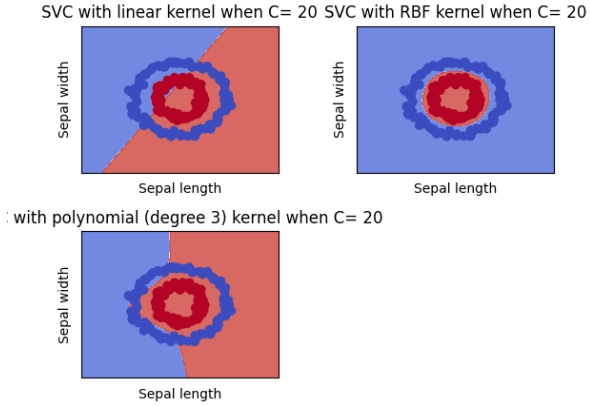

Figure 5: C=20 plots

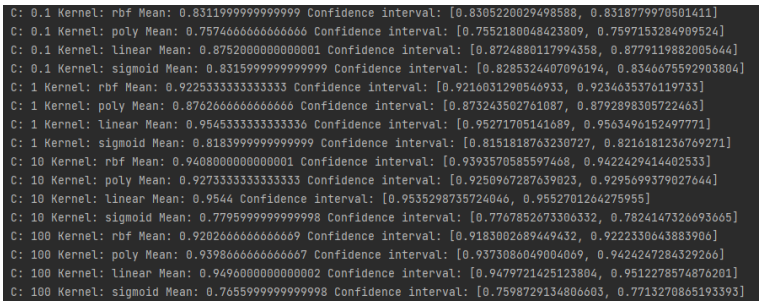

Figure 6: Dataset 2 result with different configurations

3 Part3

Figure 7 shows the configurations that I used.

```
configs = [
    {"kneighborsclassifier__metric": ["euclidean", "manhattan"],
     "kneighborsclassifier__n_neighbors": [4, 6]},

    {"svc__C": [0.1, 1],
     "svc__kernel": ["poly", "rbf", "linear"]},

    {"decisiontreeclassifier__max_depth": [5, 7],
     "decisiontreeclassifier__criterion": ["gini", "entropy"]},


    {"randomforestclassifier__n_estimators": [7, 15],
     "randomforestclassifier__max_depth": [3, 5]}
]
```

Figure 7: Configurations for all methods

My output shows:

The best parameters for each method.

The accuracy scores and their interval for each.

The accuracy scores and their interval for each as well.

For random forest, I appended best_params_ to a list of dictionaries. The best parameter is probably most occured element in the list with sized 5.

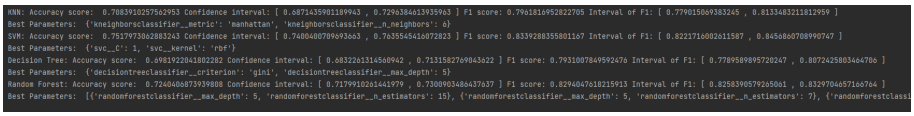

Figure 8: Output of Part3

```
Best Parameters:  [{'randomforestclassifier__max_depth': 5, 'randomforestclassifier__n_estimators': 15},
                   {'randomforestclassifier__max_depth': 5, 'randomforestclassifier__n_estimators': 7},
                   {'randomforestclassifier__max_depth': 5, 'randomforestclassifier__n_estimators': 15},
                   {'randomforestclassifier__max_depth': 5, 'randomforestclassifier__n_estimators': 7},
                   {'randomforestclassifier__max_depth': 5, 'randomforestclassifier__n_estimators': 7}]
```

Figure 9: The random forest best parameters list (Taken from my output)

As it can be seen from the figure 8 and 9,

Best parameters for KNN: manhattan distance with 6 neighbours.
KNN: Accuracy score: 0.7083910257562953
Confidence interval: [0.6871435901189943 , 0.7296384613935963]
F1 score: 0.7961816952822705
Interval of F1: [0.779015069383245 , 0.8133483211812959]

Best parameters for SVM: rbf with C=1.
SVM: Accuracy score: 0.7517973062883243
Confidence interval: [0.7400400709693663,0.7635545416072823]
F1 score: 0.8339288355801167
Interval of F1: [0.8221716002611587 , 0.8456860708990747]

Best parameters for decision tree: criterion gini and max depth 5.
Decision Tree: Accuracy score: 0.6981922041802282
Confidence interval: [0.6832261314560942 , 0.7131582769043622]
F1 score: 0.7931007849592476
Interval of F1: [0.7789589895720247 , 0.8072425803464706]

For random forest the best parameters are depth 5 and estimators 7 or 15.
Random Forest: Accuracy score: 0.7240406873939808
Confidence interval: [0.7179910261441979 , 0.7300903486437637]
F1 score: 0.8294047618215913
Interval of F1: [0.8258390579265061 , 0.8329704657166764]