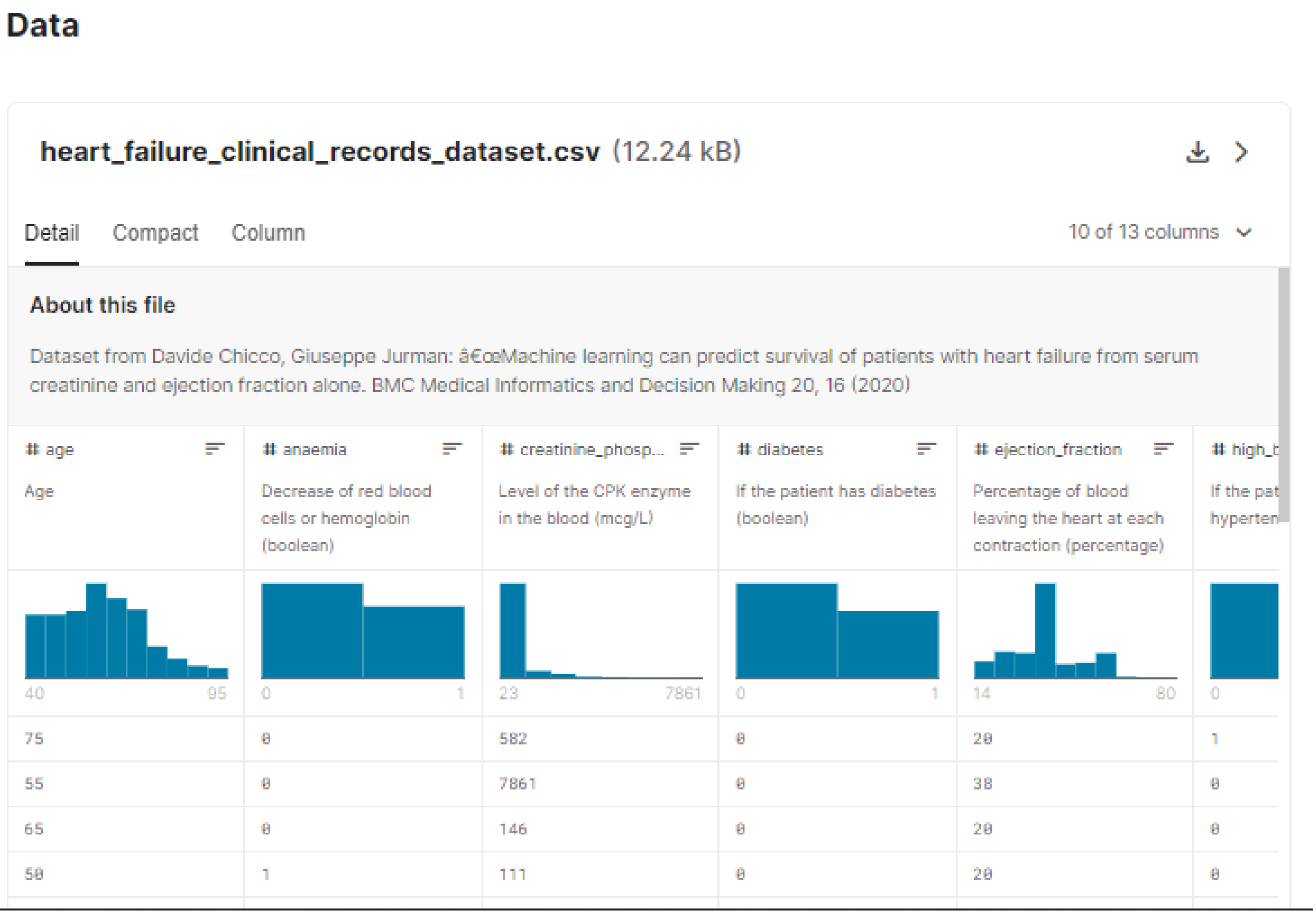
Heart-Failure Prediction with machine learning

Introduction

Heart diseases has increased due to smoking, lack of physical activity and other reasons which put pressure on the heart. Heart failure occurs when the heart cannot pump enough blood to meet the needs of the body. Machine learning can predict patients survival from their data by simulating them and important features can help in survival.



Data

heart_failure_clinical_records_dataset.csv (12.24 kB)

About this file

Dataset from Davide Chicco, Giuseppe Jurman: â€œMachine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Medical Informatics and Decision Making 20, 16 (2020)

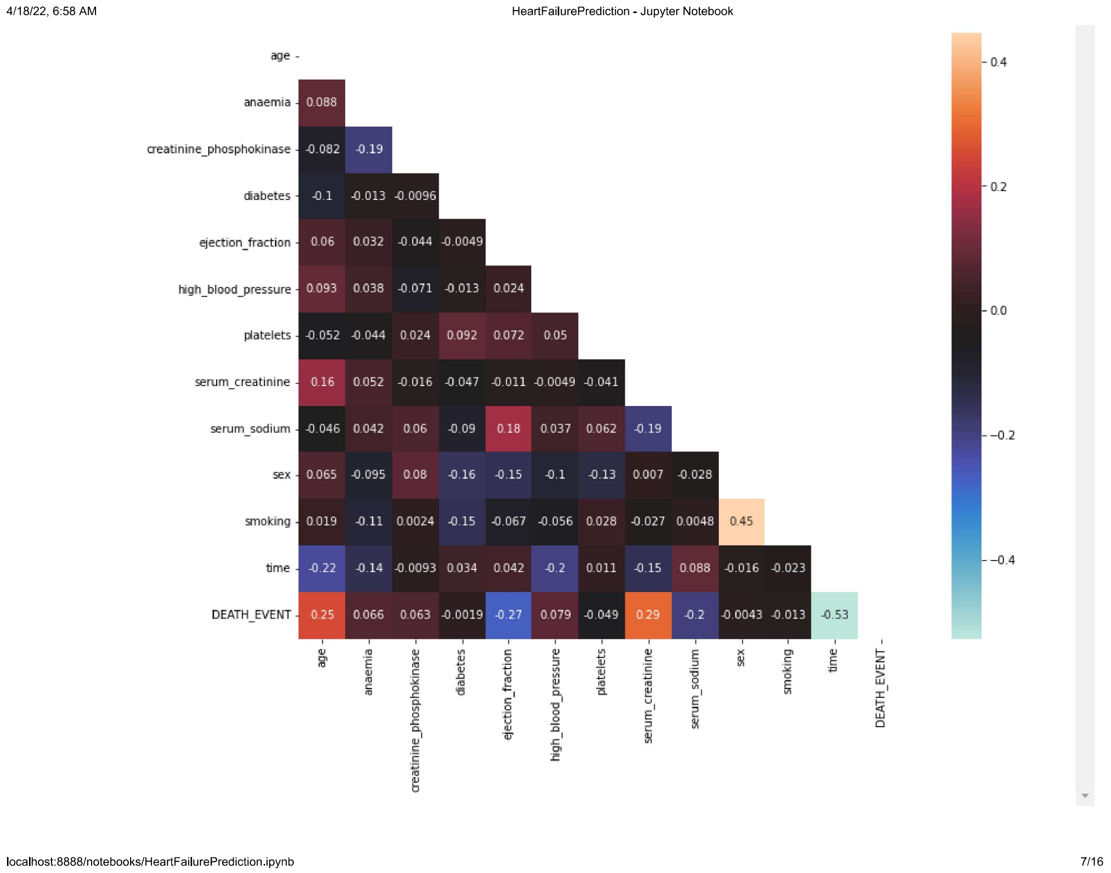
# age	# anaemia	# creatinine_phosp...	# diabetes	# ejection_fraction	# high_b
Age	Decrease of red blood cells or hemoglobin (boolean)	Level of the CPK enzyme in the blood (mcg/L)	If the patient has diabetes (boolean)	Percentage of blood leaving the heart at each contraction (percentage)	If the pat hyperten
40 95	0 1	23 7861	0 1	14 80	0
75	0	582	0	20	1
55	0	7861	0	38	0
65	0	146	0	20	0
50	1	111	0	20	0

Figure 1: Dataset
[1]

This data-set contain the medical records of 299 heart failure patients collected at the Faisalabad Institute of Cardiology, during April–December 2015. The data-set contains 13 features.

Model Development

We analyze a data-set of 299 patients with heart failure which is collected in 2015. We apply KNN, Naive Bayes and Decision Tree to predict the patients survival and also to rank the features corresponding to the most important risk factors. We also compare these results with machine learning algorithms.

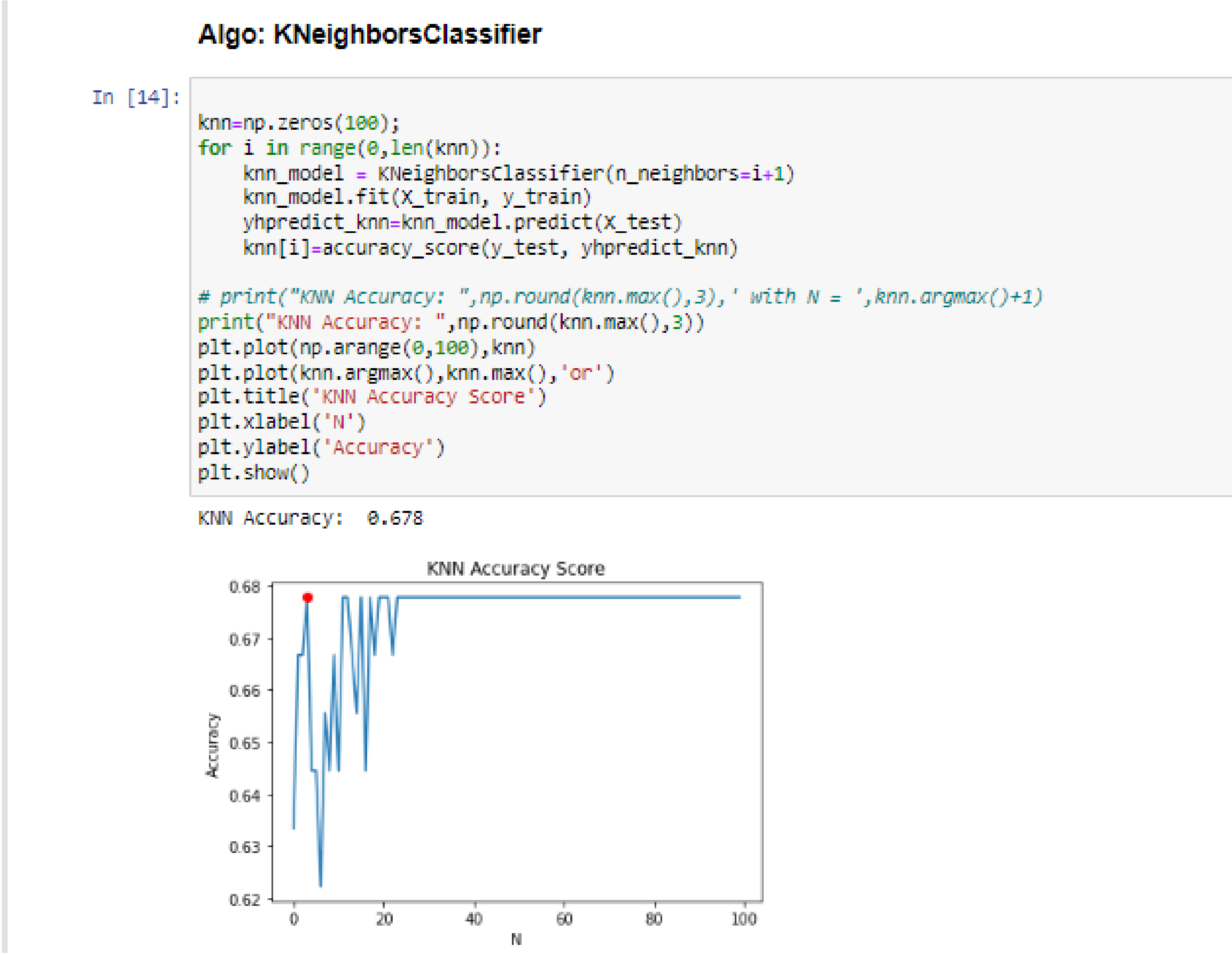


4/18/22, 6:58 AM HeartFailurePrediction - Jupyter Notebook

localhost:8888/notebooks/HeartFailurePrediction.ipynb 7/16

Figure 2: Heatmap

0.1 Algorithm: KNN



Algo: KNeighborsClassifier

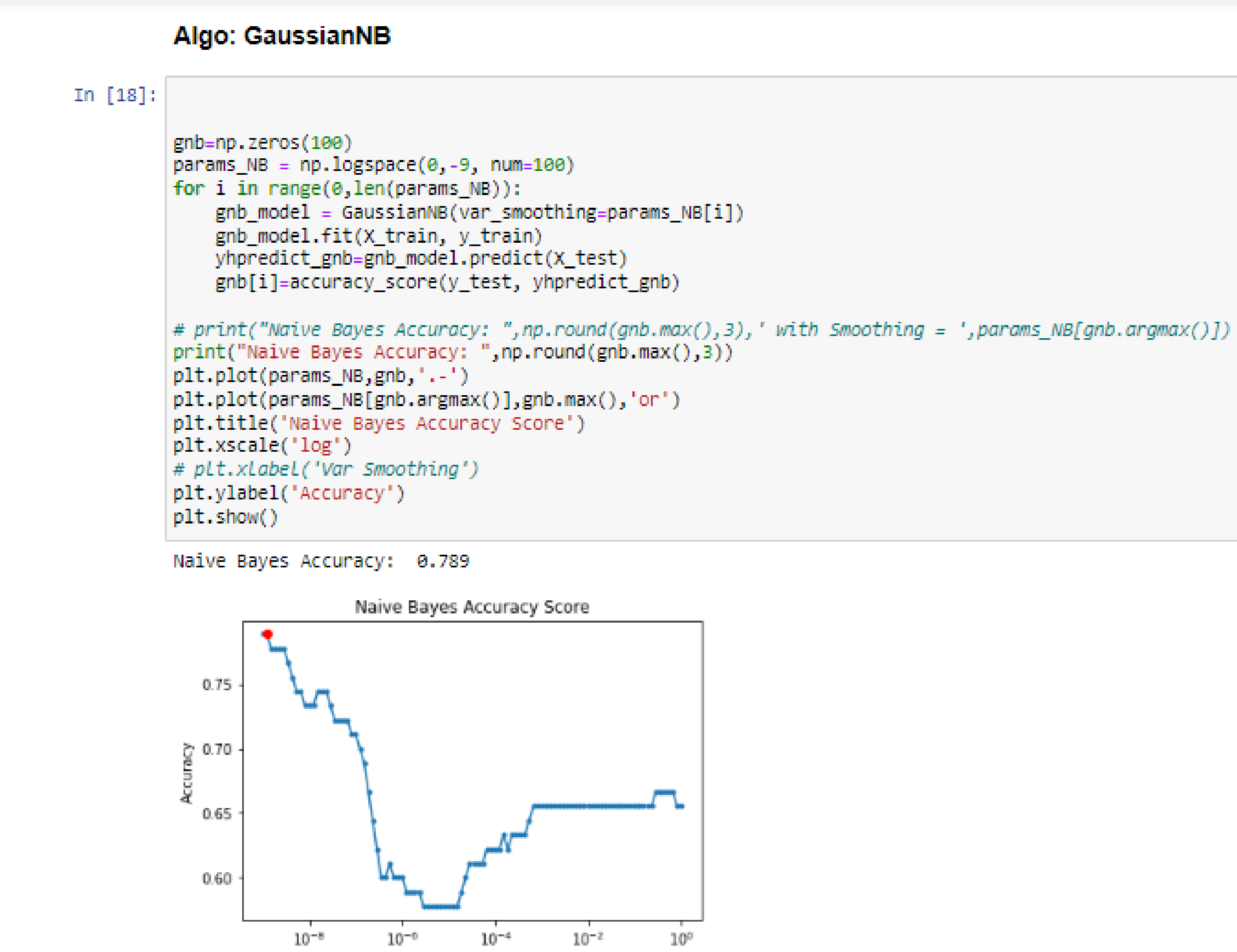
In [14]:

```
knn=np.zeros(100);
for i in range(0,len(knn)):
    knn_model = KNeighborsClassifier(n_neighbors=i+1)
    knn_model.fit(X_train, y_train)
    yhpredict_knn=knn_model.predict(X_test)
    knn[i]=accuracy_score(y_test, yhpredict_knn)

# print("KNN Accuracy: ",np.round(knn.max(),3),' with N = ',knn.argmax()+1)
print("KNN Accuracy: ",np.round(knn.max(),3))
plt.plot(np.arange(0,100),knn)
plt.plot(knn.argmax(),knn.max(),'or')
plt.title('KNN Accuracy Score')
plt.xlabel('N')
plt.ylabel('Accuracy')
plt.show()
```

```
KNN Accuracy:  0.678
```

Figure 3: KNN

0.2 Algorithm: GaussianNB



Algo: GaussianNB

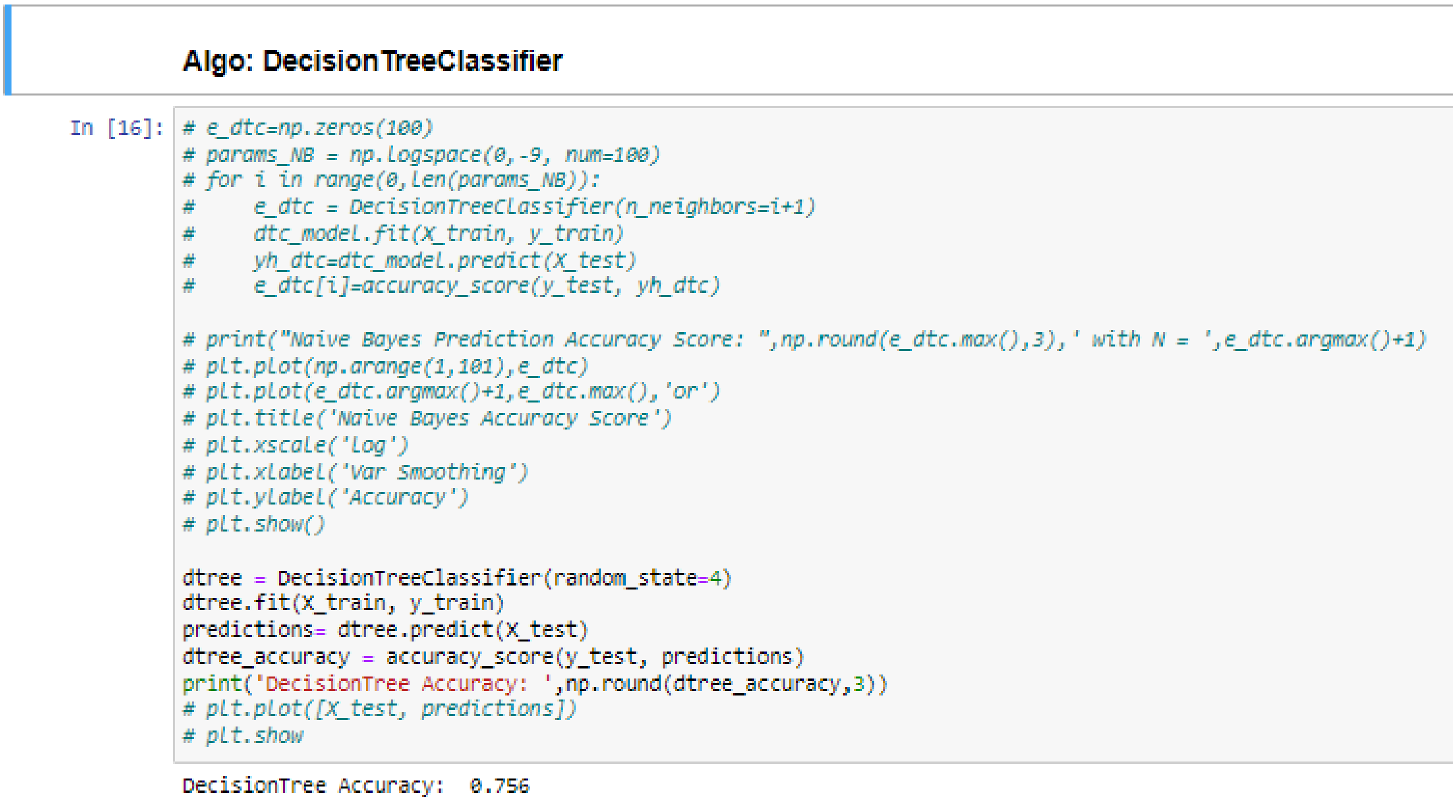
In [18]:

```python
gnb=np.zeros(100)
params_NB = np.logspace(0,-9, num=100)
for i in range(0,len(params_NB)):
    gnb_model = GaussianNB(var_smoothing=params_NB[i])
    gnb_model.fit(X_train, y_train)
    yhpredict_gnb=gnb_model.predict(X_test)
    gnb[i]=accuracy_score(y_test, yhpredict_gnb)

# print("Naive Bayes Accuracy: ",np.round(gnb.max(),3),' with Smoothing = ',params_NB[gnb.argmax()])
print("Naive Bayes Accuracy: ",np.round(gnb.max(),3))
plt.plot(params_NB,gnb,'.-')
plt.plot(params_NB[gnb.argmax()],gnb.max(),'or')
plt.title('Naive Bayes Accuracy Score')
plt.xscale('log')
# plt.xlabel('Var Smoothing')
plt.ylabel('Accuracy')
plt.show()
```

```
Naive Bayes Accuracy:  0.789
```

Figure 4: GaussianNB

0.3 Algorithm: DecisionTree

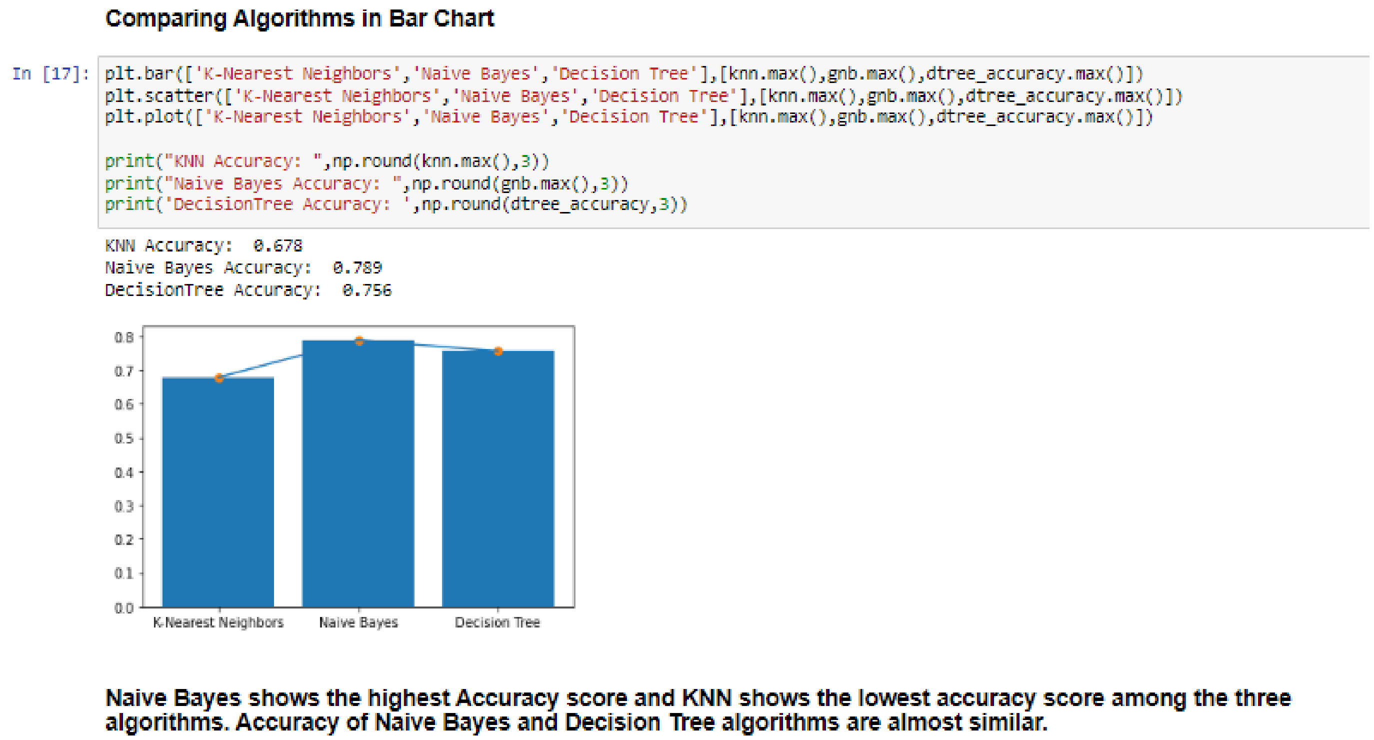


Algo: DecisionTreeClassifier

```
In [16]: # e_dtc=np.zeros(100)
         # params_NB = np.logspace(0,-9, num=100)
         # for i in range(0,len(params_NB)):
         #     e_dtc = DecisionTreeClassifier(n_neighbors=i+1)
         #     dtc_model.fit(X_train, y_train)
         #     yh_dtc=dtc_model.predict(X_test)
         #     e_dtc[i]=accuracy_score(y_test, yh_dtc)

         # print("Naive Bayes Prediction Accuracy Score: ",np.round(e_dtc.max(),3),' with N = ',e_dtc.argmax()+1)
         # plt.plot(np.arange(1,101),e_dtc)
         # plt.plot(e_dtc.argmax()+1,e_dtc.max(),'or')
         # plt.title('Naive Bayes Accuracy Score')
         # plt.xscale('log')
         # plt.xlabel('Var Smoothing')
         # plt.ylabel('Accuracy')
         # plt.show()

         dtree = DecisionTreeClassifier(random_state=4)
         dtree.fit(X_train, y_train)
         predictions= dtree.predict(X_test)
         dtree_accuracy = accuracy_score(y_test, predictions)
         print('DecisionTree Accuracy: ',np.round(dtree_accuracy,3))
         # plt.plot([X_test, predictions])
         # plt.show

         DecisionTree Accuracy:  0.756
```

Figure 5: DecisionTree

Comparison



Comparing Algorithms in Bar Chart

```
In [17]: plt.bar(['K-Nearest Neighbors','Naive Bayes','Decision Tree'],[knn.max(),gnb.max(),dtree_accuracy.max()])
         plt.scatter(['K-Nearest Neighbors','Naive Bayes','Decision Tree'],[knn.max(),gnb.max(),dtree_accuracy.max()])
         plt.plot(['K-Nearest Neighbors','Naive Bayes','Decision Tree'],[knn.max(),gnb.max(),dtree_accuracy.max()])

         print("KNN Accuracy: ",np.round(knn.max(),3))
         print("Naive Bayes Accuracy: ",np.round(gnb.max(),3))
         print('DecisionTree Accuracy: ',np.round(dtree_accuracy,3))

         KNN Accuracy:  0.678
         Naive Bayes Accuracy:  0.789
         DecisionTree Accuracy:  0.756
```

Naive Bayes shows the highest Accuracy score and KNN shows the lowest accuracy score among the three algorithms. Accuracy of Naive Bayes and Decision Tree algorithms are almost similar.

Figure 6: Comparison

Discussion

It might be possible to predict the survival of patients with heart failure from their serum creatinine and ejection fraction, but also that the prediction made on these two features alone can be more accurate than the predictions made on the complete dataset.
KNN Accuracy: 0.678
Naive Bayes Accuracy: 0.789
DecisionTree Accuracy: 0.756
We can see that, between three algorithms, Naive bayes shows the highest accuracy.

Conclusion

In this project, use of machine learning in medical is shown and it has the potential to impact on clinical practice. Now it's the new supporting tool for the physicians to predict if a heart failure patient will survive or not. If a patient survive after heart failure doctors can monitor how should the live to avoid further problems with prediction of machine learning algorithm.

References

[1] D.and Jurman Chicco. "Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone". In: (). DOI: `https://doi.org/10.1186/s12911-020-1023-5`.