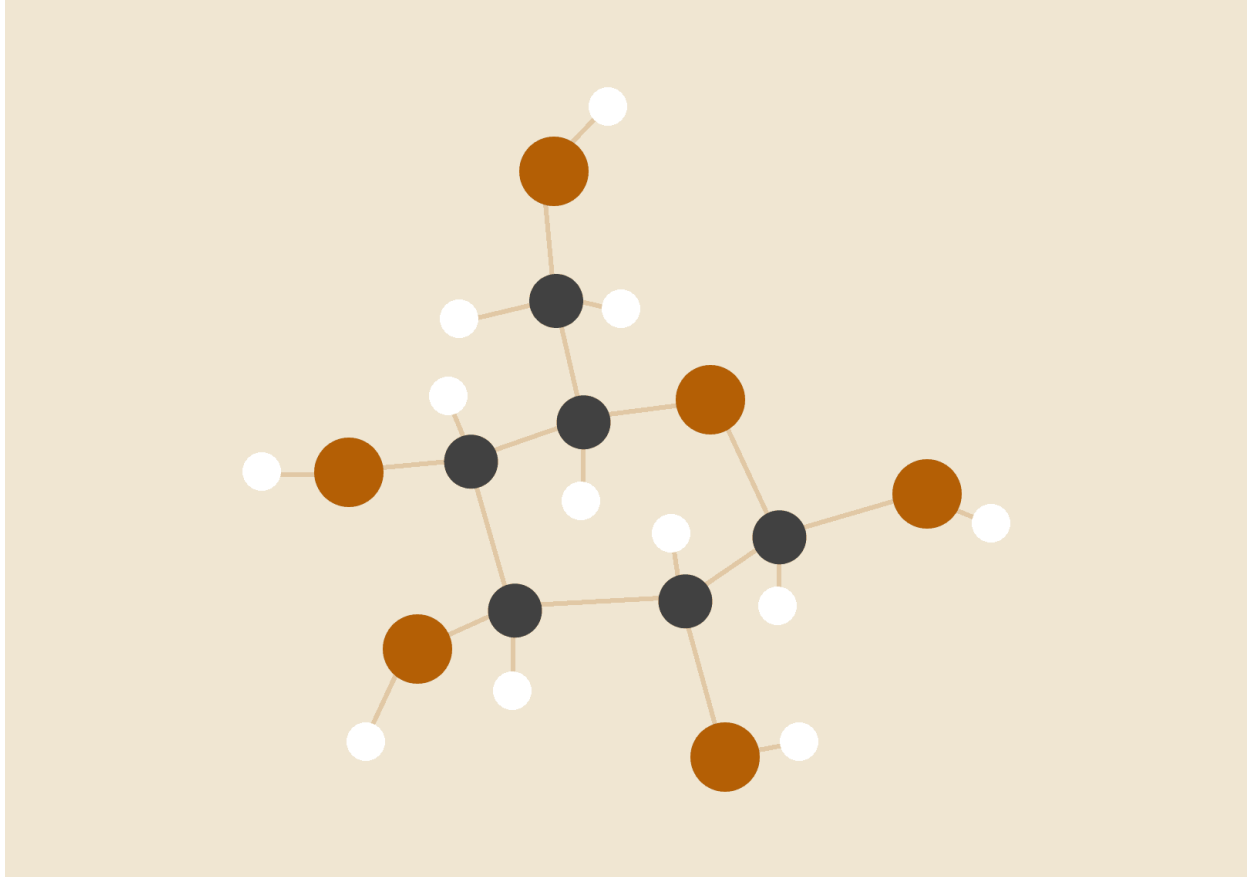


Prediction Output Report

Question 2: Classification Model Problem



Vishruth

Techconative Assignment

Introduction

This document shows the results of the predictive modeling I used to analyze the dataset. Before making predictions, I scaled some features to help the model learn better. **For a detailed implementation, including the code and visualizations that led to these results, please check out the main code file, I have attached in the GitHub Repo link.**

Classification Algorithms Used for Prediction

The following classification algorithms were applied to the dataset:

1. Random Forest

- Score: 81.34%

```
[101] model=KNeighborsClassifier()  
      model.fit(train_X,train_Y)  
      prediction5=model.predict(test_X)  
      print('The accuracy of the KNN is',metrics.accuracy_score(prediction5,test_Y))
```

➡ The accuracy of the KNN is 0.8134328358208955

2. K-Nearest Neighbors (KNN)

- Score: 81.72%

```
[89] model=RandomForestClassifier(n_estimators=100)  
     model.fit(train_X,train_Y)  
     prediction7=model.predict(test_X)  
     print('The accuracy of the Random Forests is',metrics.accuracy_score(prediction7,test_Y))
```

➡ The accuracy of the Random Forests is 0.8171641791044776

3. XGBoost

- Score: 82.46%

```
[103] xgboost = xg.XGBClassifier(n_estimators=900, learning_rate=0.1)
      xgboost.fit(train_X, train_Y)
      Y_pred = xgboost.predict(test_X)

      accuracy = metrics.accuracy_score(test_Y, Y_pred)
      print('The accuracy of the XGBoost Classifier is', accuracy)
```

```
↵ The accuracy of the XGBoost Classifier is 0.8246268656716418
```

Conclusion

In this predictive modeling project, I used three methods to predict if passengers survived. The results from each method help me understand how well each one works with the data. Before making predictions, I scaled the features, which helped improve the models' performance. For more details about the steps I took, please check the code.