CSC 466 Final Project First Draft

Seth Johnson, Noor Dhaliwal, Swayam Chidrawar

11/19/25

Main Goal Pivot

This week we decided that our previous goal to predict the exact number of points a player would score was too dependent on factors that we didn't have data for. Chief among them was the importance of the specific matchup for who would be defending the player in a game. To combat this, we decided to look at season long statistics. We pivoted our primary objective to predicting which players would have a breakout season.

Final Dataset Construction

With this new goal, we needed a new dataset. Using the nba_api data from our EDA, we restructured our data into a specific labelled dataset. We took the season long stats "per game" stats for each player and compared them to the previous season. We then took all players who were one standard deviation above the mean difference in points, assists, rebounds, steals, or blocks per game and defined them to be breakout players. All other seasons for all other players were considered "non-breakout".

Model Construction

From this dataset, the features we selected were points, assists, rebounds, steals, blocks, and minutes per game. The first model that we then trained was a simple KNN classifier with K = 5.

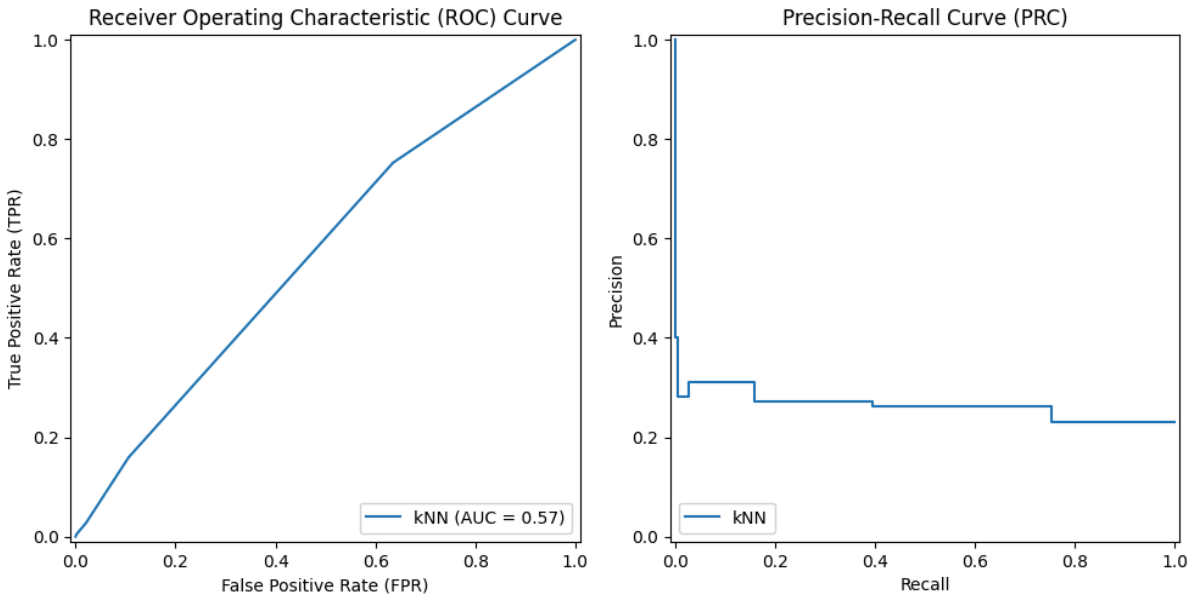
Another model we wanted to try was XGBoost. In addition to this we did try CatBoost because from our research we found that it is a bit better at handling imbalanced datasets, but we achieved essentially the same performance as XGBoost.

Results

KNN Classifier:

```
Classification Report:
              precision    recall  f1-score

           0       0.78      0.89      0.83
           1       0.31      0.16      0.21
```



XGBoost Classifier:

```
Classification Report:
              precision    recall  f1-score   support

           0       0.78      0.98      0.87      1322
           1       0.51      0.05      0.10       396
```

Analysis

The KNN classifier results were quite poor. It worked quite well on predicting non-breakout seasons, but largely failed when it came to breakout seasons. As you can see from the ROC curve, it effectively worked as well as random.

The XGBoost results were also poor. The class imbalance definitely seems to be the main issue at hand. Even attempts with SMOTE did not make much of an impact in increasing the performance metrics that we see.

Likely these issues are due to class imbalance and an imperfect dataset. Fixing these will be our largest priority before our final submission. In addition to this, an idea we have to possibly achieve better performance is to try a deep learning based classifier. This may be able to capture more obscure relationships.