Machine Learning

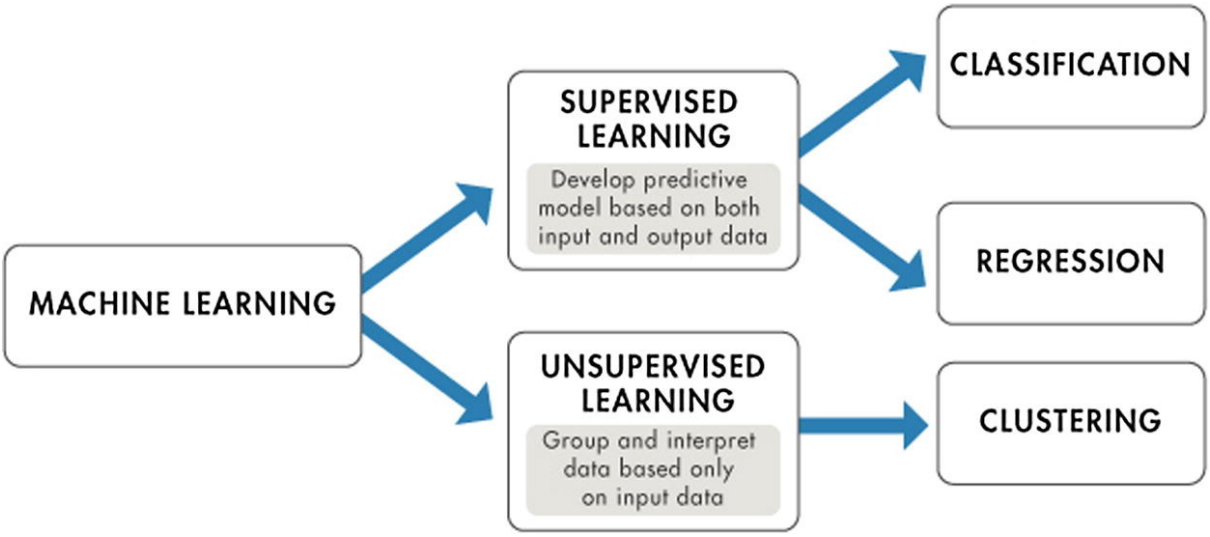
Abstract

Logistic regression algorithm was used as a supervised machine learning model to predict outcomes (win or losses) in home and away matches for 2019, when given in game statistics for NCAA basketball games from 2003-2018.

Ten ingame statistics (features) were averaged from prior games and used as input into models to predict outcomes of future games. This rolling average technique was done to reduce data leakage in the model.The lagging period used for the past average game statistics to predict upcoming matches was two days. The top five ranked features were then used to predict the outcome of matches for home and away matches.

In summary,

- Home court does give an advantage in basketball games played in the NCAA
- The algorithm predicted up to seventy percent accuracy for home and away games outcomes when using all the features for the model.
- When using the ranked features, as determined by the feature selection algorithm, the accuracy of the model stayed the same for the home matches, but declined to forty eight percent for away games.
- It is advantageous for coaches to field players that have a higher tendency to collect offensive rebounds (taller and more athletic players), than players that are smaller and nimble when playing games at home.



Introduction

In light of the outbreak of the coronavirus, the usual benefit of home field advantage presented by the home crowd is now neutralized with the presence of guards, engineers, medical personnels and team mates. The benefit of a model that would assist a coach and also influence the outcome of a match would be very important.

The machine learning model developed in this project would help the coach predict the most important in game statistics that can be used to help win NCAA matches. The results from this study could also help influence how teams plan for the upcoming 2020-21 season, where games are currently planned with no crowds.

The data used in these analyses will have the influence of crowds. However, the goal of this machine learning model will be to predict features an NCAA basketball coach could use to help influence the outcome of games, depending on the location of their matches.

The dataset provided is distributed in the following manner:

Home games: 51,825 (59 percent of games played)
Away games: 26,759 (31 percent of games played)
Neutral games: 8,920 (10 percent of games played)

Data manipulation for machine learning

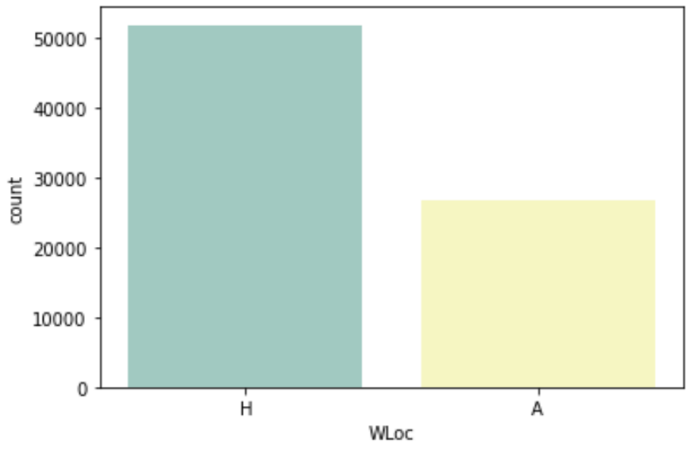
The algorithm used to build the machine learning model is a logistic regression one. The algorithm requires the dependent variable to be part of a binary variable; this is the home and away location of the winning and losing teams. The location of games played in neutral sites will be dropped.

The ratio of matches won at home and away games are displayed with the seaborn count plot in Fig1.

Fig 1: Winning location of matches for home and away games

```
sns.countplot (x="WLoc",data =capstonem,palette="Set3")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc357c18588>
```



Percent of matches won at home location: 0.66
Percent of matches won at away location: 0.34

From the plot we can observe that there were more matches won at home locations than away locations.

Machine Learning Logistic Regression Model Process

In order to predict the final scores for the NCAA games, the original data frame was appended , reshaped and arranged so that it can be used for the machine learning model.

A dataset containing all individual game statistics (such as shooting percentages, rebounds, number of turnovers , steals) was used to predict the outcome of final match results, regardless of whether the team is a home team or an away team. The order of the matches played per individual teams were preserved when manipulating the data .

Since in-game statistics for NCAA games are not usually known until a match has been played, historical in game data (features) are averaged out , and used as features for upcoming matches. Match-related features undergo a separate averaging process before being re-merged with the dependent variable of win and losses (Fig 4) . The dataframe created from this manipulation is then subdivided into two separate dataframes of home and away teams.

The features for the home wins are ranked according to how well they predicted wins and losses of matches. This same step is carried out separately on features for the away team.

Handling Categorical Data

The categorical features from the dataframe were converted to a numerical format.These allowed the machine learning algorithms to handle the features effectively. The conversion to numerical data involved assigning a value of 1 for home games,and a value of 0 for away games. Likewise a value of 1 was assigned for wins and 0 for losses. A sample of this data frame is displayed in Fig 4.

The model performance was evaluated by classifying match results into home and away. The model was then assessed based on the number of matches that was correctly identified, using a standard classification matrix.The description of the features used for the data modeling is in Table 1.

Table 1:Features used for the analysis

Acronym	Description
FGM	Field Goal made
FGA	Field Goal attempts
FTM	Free Throw made
FTA	Free Throw attempts
FGM3	3PT shooting field goal made
FGMA3	3PT shooting field goal attempts
Ast	Assists per team
TO	Turnovers per team
Blk	Blocks per match per team
PF	Personal fouls per team
OR	Offensive Rebounds per team
DR	Defensive Rebounds per match per team
FTMRAT	Field throw percentage
FGMRat	Field goal percentage
FGM3Rat	Three point percentage
Loc	Location of match played
Result	Win or loss

identifier	Unique identifier created from a union of season and DayNum

The initial and the final form of the data frame is shown in Fig 2 and 3 below.

Fig 2: Initial Dataframe

	Season	DayNum	WTeamID	WScore	LTeamID	LScore	WLoc	NumOT	WFGM	WFGA	WFGM3	WFGA3	WFTM	WFTA	WOR	WDR	WAst	WTO	WStl	WBlk	WPF	LFGM	LFGA	LFGM3
0	2003	11	1458	81	1186	55	H	0	26	57	6	12	23	27	12	24	12	9	9	3	18	20	46	3
1	2003	12	1161	80	1236	62	H	0	23	55	2	8	32	39	13	18	14	17	11	1	25	19	41	4
2	2003	12	1458	84	1296	56	H	0	32	67	5	17	15	19	14	22	11	6	12	0	13	23	52	3
3	2003	13	1166	106	1426	50	H	0	41	69	15	25	9	13	15	29	21	11	10	6	16	17	52	4
4	2003	13	1323	76	1125	48	H	0	25	56	10	23	16	23	8	35	18	13	14	19	13	18	64	8

Fig 3: Final Dataframe

	Season	DayNum	TeamID	Score	FGM	FGA	FGM3	FGA3	FTM	FTA	OR	DR	Ast	TO	Stl	Blk	PF	Results	identifier	Loc	FGM3Rat	FGMRat	FTMRat
0	2003	011	1458	81	26	57	6	12	23	27	12	24	12	9	9	3	18	W	2003_011	H	0.50	0.46	0.85
1	2003	012	1161	80	23	55	2	8	32	39	13	18	14	17	11	1	25	W	2003_012	H	0.25	0.42	0.82
2	2003	012	1458	84	32	67	5	17	15	19	14	22	11	6	12	0	13	W	2003_012	H	0.29	0.48	0.79
3	2003	013	1166	106	41	69	15	25	9	13	15	29	21	11	10	6	16	W	2003_013	H	0.60	0.59	0.69
4	2003	013	1323	76	25	56	10	23	16	23	8	35	18	13	14	19	13	W	2003_013	H	0.43	0.45	0.70
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
157163	2019	130	1416	55	19	53	6	20	11	19	8	22	7	16	6	6	21	L	2019_130	A	0.30	0.36	0.58
157164	2019	131	1272	58	16	68	4	23	22	26	21	26	4	9	5	5	21	L	2019_131	H	0.17	0.24	0.85
157165	2019	131	1420	49	15	44	8	26	11	14	4	23	9	13	6	4	19	L	2019_131	A	0.31	0.34	0.79
157166	2019	131	1343	77	28	62	6	24	15	20	9	24	9	10	4	1	17	L	2019_131	A	0.25	0.45	0.75
157167	2019	132	1217	85	28	62	10	32	19	24	12	15	7	9	1	2	22	L	2019_132	A	0.31	0.45	0.79

As can be seen in the final dataframe, the final row count is now 157168, up from the original count of 78584. The original data frame which had the result of the matches (win, loss), and winning location, all in one row, is now separated such that the result of matches and location of matches are its own separate columns.

Splitting of data for Train Test Split

The data was originally split into an x-array (which contained the data that we will use to make predictions), and a y-array (which contained the data that was to be predicted).

The x-array are the in game statistics for each NCAA team, while the y-array is the win and losses for each team.

This data frame was later sub divided into two sets: **training data set**, and the **test dataset**. The training data was chosen to be from **seasons 2003 -2018**, while the test data for the model was chosen to be the **2019 season**. The reason for this is that features for the model would be selected based on the information on the training set, not on the whole data set. The test set was kept separate from the training set so as to evaluate the performance of the feature selection. By also separating the training dataset from the test dataset, it would be possible to show how accurately the models could predict a given test data.

The effect of feature selection was measured with the aid of a classification report. This was done by comparing the effect of removing features from the model and seeing the effect on the classifier accuracy.

Summary of training and test dataset for home games

Training dataset shape: (73389, 10) , Testing dataset shape: (4854, 10)

Summary of training and test dataset for away games

Training dataset shape: (73389, 10) , Testing dataset shape: (4854, 10)

Fig 4: Training data set with first two matches averaged out.

	Season	DayNum	TeamID	Score	OR	DR	Ast	TO	Stl	Blk	PF	identifier	FGM3Rat	FGMRat	FTMRat	Results	Loc
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1	1
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1	1
123	2003.0	11.5	1458.0	82.5	13.0	23.0	11.5	7.5	10.5	1.5	15.5	2003011.5	0.395	0.470	0.820	1	1
360	2003.0	15.5	1458.0	83.5	13.5	26.5	13.0	9.5	7.0	2.5	14.5	2003015.5	0.415	0.465	0.800	1	0
79088	2003.0	22.5	1458.0	76.0	13.0	27.5	12.0	10.0	6.0	5.0	12.5	2003022.5	0.395	0.435	0.790	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
151662	2018.0	101.5	1101.0	72.5	16.0	23.0	11.5	15.0	7.5	3.5	21.0	2018101.5	0.260	0.460	0.695	0	0
151808	2018.0	106.5	1101.0	67.0	16.0	23.5	12.5	16.0	7.0	3.5	19.5	2018106.5	0.325	0.420	0.675	0	1
151920	2018.0	112.0	1101.0	60.5	12.0	22.0	8.0	15.5	7.5	1.5	16.5	2018112.0	0.220	0.385	0.670	0	1
152137	2018.0	115.5	1101.0	69.5	11.0	22.0	9.5	13.0	9.5	3.0	20.0	2018115.5	0.190	0.450	0.565	0	0
73621	2018.0	119.0	1101.0	64.0	11.5	23.0	14.5	16.5	8.5	3.5	19.5	2018119.0	0.300	0.450	0.475	1	0

Feature Selection by Recursive Feature Elimination

The feature selection method used for the project was the Recursive Feature Elimination (RFE). Recursive Feature Elimination (RFE) uses an idea to repeatedly construct a model based on the best or worst performing feature. These features are then set aside with the algorithm repeating the process with the rest of the features. This process is applied until all features in the dataset are exhausted. The goal of RFE is to select features by recursively considering smaller and smaller sets of features.The final features would then be ranked based on the algorithm.
A correlation matrix was also created to check the correlation between ingame statistics.

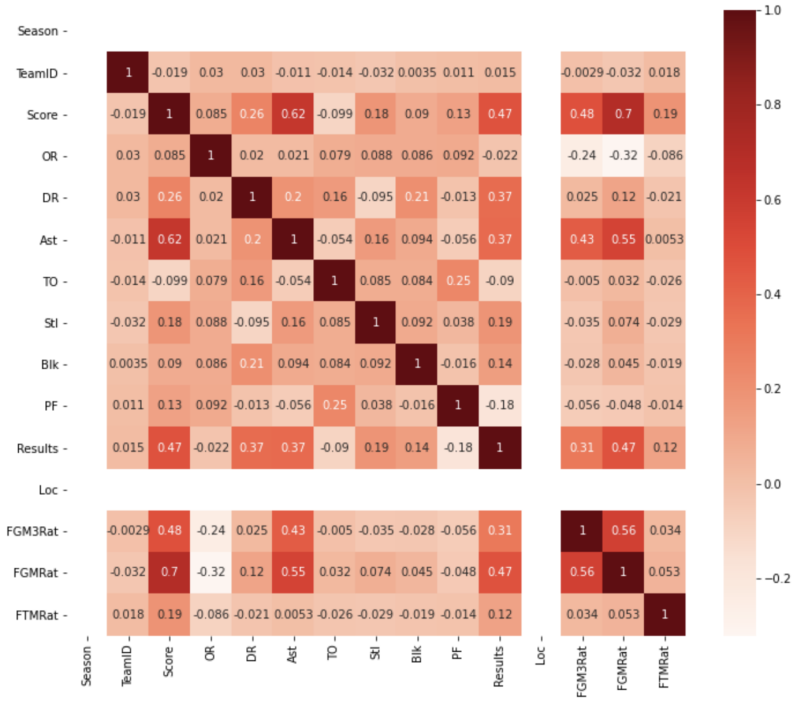
Table: Feature Ranking for Home games

	Feature	Ranking
0	OR	1
5	Blk	1
7	FGM3Rat	1
8	FGMRat	1
9	FTMRat	1
3	TO	2
4	Stl	3
1	DR	4
2	Ast	5
6	PF	6

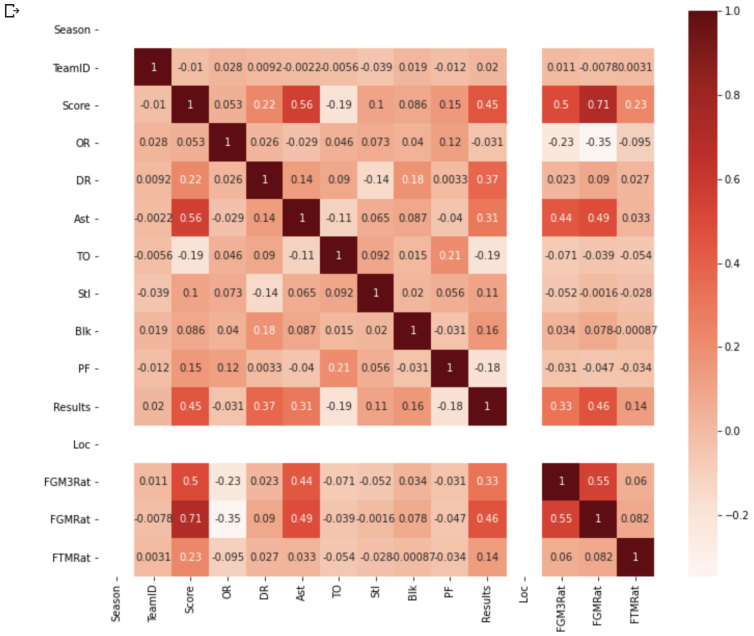
Table: Feature Ranking for Away games

	Feature	Ranking
3	TO	1
5	Blk	1
7	FGM3Rat	1
8	FGMRat	1
9	FTMRat	1
0	OR	2
4	Stl	3
1	DR	4
6	PF	5
2	Ast	6

Correlation matrix for home games



Correlation matrix for away games



Implementing the model

In creating the model for the logistic regression, a comparison of the accuracy scores were made for the model before and after the RFE model was used for feature selection.The result from the feature selection based on the RFE was fed into the model, with the data being transformed before it could be used in the model. This is because the number of features were changed from 10 to 5 for the RFE model. The top 5 features selected by the RFE model were used for the RFE evaluation.

Test Classification Report- **ALL** ten feature selected for home games

```
(4854, 10)


[[ 200 1513]
 [   6 3135]]


[Test Classification Report]
              precision    recall  f1-score   support

           0       0.97      0.12      0.21      1713
           1       0.67      1.00      0.80      3141

    accuracy                           0.69      4854
   macro avg       0.82      0.56      0.51      4854
weighted avg       0.78      0.69      0.59      4854


[Test] Accuracy score (y_predict_test, ytest_home): 0.6870622167284713
```

Comparing performance after feature selection using RFE : home matches

```
[Confusion Matrix for Test Data using RFE]
[[ 250 1463]
 [  44 3097]]


[Test Classification Report]
              precision    recall  f1-score   support

           0       0.85      0.15      0.25      1713
           1       0.68      0.99      0.80      3141

    accuracy                           0.69      4854
   macro avg       0.76      0.57      0.53      4854
weighted avg       0.74      0.69      0.61      4854


[Test] Accuracy score (y_predict_test, ytest_home): 0.6895344046147507
```

Interpretation: Before Feature selection for home games

In total the model made 4854 predictions, out of these predictions, the classifier predicted wins **1513+3135** times for home games, while predicting **200+6** losses for home games. In reality there were **200+1513** losses, and **6 + 3135** wins in home matches

From these confusion matrix, the classifier accuracy is 0.69

Interpretation: After Feature selection for home games

In total the model made 4854 predictions, out of these predictions, the classifier predicted wins **1463+3097** times for home games, while predicting **250+44** losses for home games. In reality there were **250+1463** losses, and **44+ 3097** wins in home matches.From these classification report, the classifier accuracy is 0.69

Based on the performance metrics above, the overall classifier accuracy was similar for when all the features were used to predict the outcome of home games, and when the top five features were used to predict the classifier accuracy. These top five features as denoted by the ranking of 1 in Table 2 are : FGM3Rat, FGMRat, OR, FTMRat and BLK.

The accuracy on both outputs is 0.70, which meant that 70 % of the outcome predicted for wins and losses for the 2019 season were correct. This is a good result for our model,especially if it can predict results for win and loss of home games 70 % of the time based on picking five important features.

Comparing performance before feature selection: Away Matches

```
(4854, 10)
[[3133    8]
 [1484  229]]
              precision    recall  f1-score   support

           0       0.68      1.00      0.81      3141
           1       0.97      0.13      0.23      1713

    accuracy                           0.69      4854
   macro avg       0.82      0.57      0.52      4854
weighted avg       0.78      0.69      0.61      4854

[Test] Accuracy score (y_predict_test, ytest_away): 0.6926246394725999
```

Comparing performance after feature selection: Away Matches

```
[Confusion Matrix for Test Data using RFE]
[[ 632 2509]
 [  28 1685]]


[Test Classification Report]
              precision    recall  f1-score   support

           0       0.96      0.20      0.33      3141
           1       0.40      0.98      0.57      1713

    accuracy                           0.48      4854
   macro avg       0.68      0.59      0.45      4854
weighted avg       0.76      0.48      0.42      4854



[Test] Accuracy score (y_predict_test, ytest_away): 0.4773382777091059
```

Interpretation: Before feature selection for away games

In total the model made 4854 predictions, out of these predictions, the classifier predicted wins **8+2229** times for away games, while predicting **3133+1484** losses for home games. In reality there were **3133+8** losses, and **1484 + 2229** wins in home matches

From these confusion matrix, the classifier accuracy is **0.69**

Interpretation: After Feature selection for away games

In total the model made 4854 predictions, out of these predictions, the classifier predicted wins **2509+1685** times for home games, while predicting **632+28** losses for home games. In reality there were **632+2509** losses, and **28+ 1685** wins in home matches

From the classification matrix, the classifier accuracy i**s 0.48.**

Based on the performance metrics above, the overall accuracy is different for both the before and after selection of the top five features used to predict the outcome of home games. These top five features as denoted by the ranking of 1 in Table 3 are : FGM3Rat, FGMRat, FTMRat, TO and Blk.

Since the accuracy result does not predict as well on the test data , when using the top 5 ranked features, we can say that these features do not do as good a job on predicting the result of away matches. The original 10 features are better predictors of results of away matches.

This is still a good result , especially if the model can predict the outcome of away matches in 2019, seventy percent of the time, when using all the ten features.

The receiver operating characteristics (ROC) curve is another common tool used with binary classifiers. The dotted line represents the ROC curve of a purely random classifier; a good classifier stays as far away from that line as possible (toward the top-left corner).

When comparing the AUC curves both the home and away games curves, the results show similar predictions for home and away games.

Fig 6: AUC/ROC curve for Away games

```
AUC: 0.9775666730100043
AUC scores computed using 5-fold cross-validation: [0.47593976 0.53316678 0.59748753 0.49196562 0.49584118]
AUC scores computed using 5-fold cross-validation: [0.47593976 0.53316678 0.59748753 0.49196562 0.49584118]
```

```
Text(0.5, 1.0, 'ROC Curve')
```

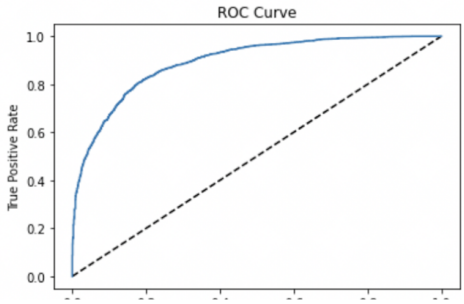
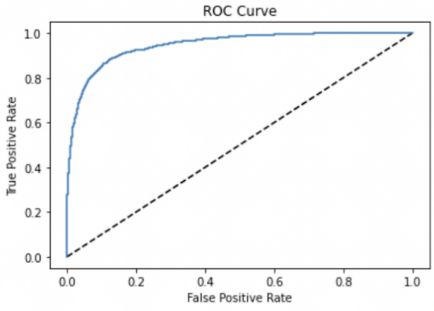


Fig 5: AUC/ROC curve for Home games

```
Text(0.5, 1.0, 'ROC Curve')
```




```
AUC: 0.9484219828692092
AUC scores computed using 5-fold cross-validation: [0.47300468 0.53468432 0.60063523 0.49177316 0.52491914]
```

Conclusions and Future Work

The model correctly identifies the results of 2019 matches a high percentage of the time when a team is playing at home or away.The accuracy of the classifier for the testing data was at seventy percent when all features were selected.

This logistic regression model does a better job of predicting the test data outcomes for home games than away games when using the test data for the 2019 season, and as would be expected, the most important factors contributing to a team winning a match is the percentage of basket scored in a game i.e. (FGMRat, FTMRat, FGM3Rat). These features as with most of the other top five features track alike for teams winning both at home, and away. In addition to these features, the block (blk) feature is also an important feature in predicting a match outcome.

The two features that were different from the top five features for both the away and the home games are the offensive rebound (OR) and the turnovers (TO). Offensive rebounds are a more favored stat when playing at home , while turnovers (TO) are favored while playing away. Basically, saying that home teams have a tendency to secure the ball better than away teams.

The correlation of the results of matches for home and away matches with turnover is a negative one, while the correlation with offensive rebound is a positive one. This correlation shows that high turnovers can cost a team to lose a match, and high offensive rebounds can help a team win a match.

Another observation from the correlation matrix, is the high positive correlation **(0.62)** observed for the assist feature and the results of matches.

Based on the information provided by the model on assist, rebounds and turnovers, a coach can rotate his team in such a way that players that have a higher tendency of providing assists, limiting turnovers and having a greater tendency of collecting offensive rebounds should be played more minutes when playing at home. Likewise players that have less tendency to turnover the ball, providing assists could be played more when playing at away locations.

In general, players that collect a lot of offensive rebounds and blocks are similar types of players (taller than average players , or players with great athletic vertical leap). A coach should think of giving these types of players more minutes when playing at home, since these features are important in determining the outcome of matches.

Finally, since the five features selected from the overall set of features do not greatly decrease the accuracy of the model when used to predict the performance of the model for home matches,

the coach can focus on these top five features when trying to increase his advantage in winning a match when playing at home.

Further information that could help with fine tuning this model, could be plotting the average heights of players and minutes played, when playing matches at home versus when playing matches at away locations. The result of this data with the outcome of matches, can help a coach find out whether specific types of players collecting offensive rebounds, deserve more minutes when playing matches at home or away.