**Milestone 5: Final Paper**

Bryce Kirby

DSC 630

6/1/2024

# Introduction

The NBA is mainly comprised of over 30 teams that have multi-million dollar budgets that they use to build a talented team that will hopefully win enough games to bring home the NBA title. To do this, the teams build their rosters based on talent and budget.  While team payrolls are large, they are not unlimited, and strategic decisions must be made to develop the most talented team while maintaining budget constraints. The assumption of most teams, and players as well, is that the more money a particular player demands, the better the player.  This is not always the case, however.  There are numerous instances where players have been drastically overpaid and dramatically underperformed.Conversely, there have been players who have performed well above their higher-paid counterparts. Ideally, there would be a mechanism where player salary could be determined by consistent player performance. The purpose of this course project is to use predictive analytics to predict what a player’s salary should be by looking at player statistics.

## Method

 During this course project I have used different approaches such as using EDA to help take note of any type of missing data along with max and min values to track any outliers. When it came to collecting the data I used a website called Kaggle which is known for looking at many data sets. Along with that, I had to look at another website to look at certain variables that were needed for this course project as well. This paper will show further details about the models along with the results of what I was able to discover while working on this course project.  The EDA process worked out well since I was able to look at how NBA players would get paid based on certain variables. I used about five models such as Ordinary Least Squares, Ridge Regression, Lasso regression, ElasticNet Regression, and Extreme gradient Bossting Regression which proved very helpful during the course project.

**Model 1: Ordinary Least Squares (OLS)**: The reason for selecting this model is that it is one of the most commonly used models and would serve as a good baseline with which I could compare the subsequent models. The table below shows the results from this model.

OLS Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| OLS | 4398500 | 0.63 | 5024129 | 0.57 |

The R2 values for the models indicate that we are able to account for about 60% of the salary prediction variability with the features we selected.

The root mean square error (RMSE) values will serve as a reference point for the other linear models we are going to use.

**Model 2: Ridge Regression (RR)**: The ridge regression model is an extension of linear regression where the loss function is modified to minimize the complexity of the model. Ridge regression includes an α variable that I can adjust to help improve the fit. The higher the α value, the more likely the model is going to suffer from under-fitting. A lower α value can lead to under-fitting. The table below shows the results from the RR model for two different α values.

RR Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| RR, α=0.01 | 4398504 | 0.63 | 5024400 | 0.57 |
| RR, α=10 | 4608385 | 0.59 | 5203120 | 0.54 |

The R2 values for the models indicate that I was able to account for about 60% of the salary prediction variability with the features I have selected.

The RMSE values for the RR method are slightly higher than the OLS values. This indicates that the OLS predictions are more accurate.

Increasing the α value decreased the model accuracy.

**Model 3: Lasso Regression (LR)**: Lasso regression is another modification of linear regression modeling. In Lasso, the loss function is modified to minimize the complexity of the model by limiting the sum of the absolute values of the model coefficients. I will select different α values to tune the model for the best fit. The table below shows the results from the LR model for two different α values.

LR Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| LR, α=0.01 | 4398500 | 0.63 | 5024129 | 0.57 |
| LR, α=100 | 4398502 | 0.63 | 5023978 | 0.57 |

The R2 values for the models indicate that I was able to account for about 60% of the salary prediction variability with the features selected.

The RMSE values for the LR method are essentially identical to the OLS values. This indicates that the OLS and Lasso predictions are basically the same.

Increasing the α value had little impact on the model accuracy.

**Model 4: ElasticNet Regression (ENR)**: ElasticNet regression combines the properties of both the Ridge and Lasso regression methods. ENR also utilized an α term to help improve fit. The table below shows the results from the ENR model for two different α values.

ENR Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| ENR, α=0.01 | 4432638 | 0.62 | 5070003 | 0.56 |
| ENR, α=10 | 7010747 | 0.06 | 7488845 | 0.05 |

The R2 values for the models were still around 60%, but were slightly lower that what I saw within the other models.

The RMSE values for the ENR method were the highest of all the models. This indicates the least accurate prediction model. This is likely due to the fact that I had removed features that were highly correlated with one another.

Increasing the α value had a negative impact on the prediction accuracy.

**Model 5: Extreme Gradient Boosting Regression (XGBR)**—XGBR is an ensemble modeling technique where new models are added to correct the errors made by existing models. This method uses the gradient boosting decision tree algorithm. Table 6 shows the results from the XGBR model.

XGBR Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| XGBR | 1818766 | 0.92 | 4531190 | 0.65 |

The R2 values for the XGBR method were the best of all models evaluated. Our training set R2had a value of 92%, which is excellent. The results diminished to only 65% when applied to the test set, but this about 10% better than our other models. The reason for the lower R2 value for the test set is likely due to the lower number of data points. Remember that the test set contained only one-third of the original data.

The RMSE values for the XGBR method were also the lowest of all models evaluated. This indicates the most accurate prediction model.

Because the XGBR model provided the best overall fit, this is the model I have chosen as the final model.

The figure below shows the actual versus predicted salaries for the XGBR model test and training sets. The solid red line shows the best fit for each data plot. As one can see, there is good clustering near the best fit line, which indicates a good model result between the actual and predicted salaries. The test set fit is not as good as the training set and this is again due to the reduced number of data points in the test fit dataset.

**A graph of a graph of a graph

Description automatically generated with medium confidence**

Comparison of Xgboost Regression Model Output: Training Set on Left and Test Set on Right

### **The Results**

The initial linear model results were good, but I wanted to include other models to ensure that I was making the most accurate predictions possible.  Of the five models I considered, the Extreme Gradient Boosting Regression (XGBR) proved to be the most accurate.  I was able to obtain an R*2* value of 0.92 for our training subset of data. This was about a 30% improvement over the over four models we used.  Additionally, I reran the XGBR model on the complete dataset (without separating it into test and training subsets) to correlate player names with salary predictions so I could see which athletes were overpaid and which were underpaid. When I ran this model instance, we obtained an R*2* value of 0.93. The final goal of this project was to see what players are currently being overpaid as well as those players who are underpaid. The figure below shows the top 10 overpaid players as predicted by our model.  As you can see, Chandler Parsons is the most overpaid player by over $8M.

A graph of red columns

Description automatically generated

The figure below shows the top 10 underpaid players as predicted by our model.  As you can see, Marcus Morris is the most underpaid player by over $5M.

**A graph of a number of players

Description automatically generated**

While there were some challenges when it came to the course project overall I was successful when it came to solving the problem for this project.

#### Milestone 4: Finalizing Results

# Data Preparation

When it came to this I used Exploratory Data Analysis (EDA) which helped me take note of any missing data along with max and min values to track any type of outliers. The result of these summary statistics showed that there were indeed some missing values that would require further investigation but the max and min numbers appeared to be within the expected ranges of possible values.  The figures below provide a great help for this project along with a good explanation of my use of EDA.

A screenshot of a computer

Description automatically generated

Figure 1. Sample summary statistics for NBA dataset.

Histograms: The output from the histogram plots are shown in Figure 2.

A collage of blue and white graphs

Description automatically generated

Figure 2. Histogram plots of all NBA dataset variables.

The histogram plots do indeed indicate those variables with normal distributions, such as Player\_Efficiency\_Rating. The plots also show that there do not appear to be any outlying data points were need to be concerned about.

The Random Forest Classifier is a method that utilizes the RandomForestClassifier from the sklearn library.  Random forests are one of the most popular machine-learning algorithms. They are so successful because they provide in general a good predictive performance, low overfitting, and easy interpretability. This interpretability is given by the fact that it is straightforward to derive the importance of each variable on the tree decision. In other words, it is easy to compute how much each variable is contributing to the decision. Each of our variables was subjected to this algorithm and then ranked on importance to the model. The output of this process is shown in Figure 3.  The first column is the Data Frame column number, the indexcolumn is the name of the dataset variable, and the RF column is the Random Forest value.  Due to the length of the output, only the first 10 values are shown.

A screenshot of a computer

Description automatically generated

Figure 3. Output from Random Forest variable importance calculation.

The Recursive Feature Elimination (RFE) technique is also from the sklearn library. RFE is a backward selection of the predictors. This technique begins by building a model on the entire set of predictors and computing an importance score for each predictor. The least important predictor(s) are then removed, the model is rebuilt, and importance scores are computed again, hence the recursive nature of the process.The output from this process is shown in Figure 4.  Like before, the first column is the Data Frame column number, the index column is the name of the dataset variable, and the RFE column is the Recursive Feature Elimination result.  The values of the RFE column are shown as True because they have not been eliminated in the RFE selection process.  In other words, these remaining variables are considered important to the overall predictive model.  The output below (in alphabetical order) shows all of the variables that returned a True value.

A screenshot of a computer

Description automatically generated

Figure 4. Output from RFE calculation.

The Extra Trees Classifier technique uses the ExtraTreesClassifier module from the sklearn library.  In concept, the Extra Trees Classifier is very similar to a Random Forest Classifier and only differs from it in the manner of construction of the decision trees in the forest.  Each Decision Tree in the Extra Trees Forest is constructed from the original training sample. Then, at each test node, each tree is provided with a random sample of k features from the feature set. From this, each decision tree must select the best feature to split the data.  This random sample of features leads to the creation of multiple de-correlated decision trees. The output from this process is shown in Figure 5. The first column is the Data Frame column number, the index column is the name of the dataset variable, and the Extratrees column is the algorithm result.  Due to the length of the output, only the first 10 values are shown.

A screenshot of a computer

Description automatically generated

Figure 5. Output from Extra Trees Classifier.

The Chi-Square method is also from the sklearn library.  The Chi-Square Test is used in statistics to test the independence of two variables.  In feature selection, the two variables are the occurrence of the feature and the occurrence of the class.  In other words, I want to test whether the occurrence of a specific feature and the occurrence of a specific class are independent.  When the two events are independent, the observed count is close to the expected count, thus a small chi-square score. So, a high chi square value indicates that the hypothesis of independence is incorrect. In other words, the higher the value of the chi-square score, the more likelihood the feature is correlated with the class, thus it should be selected for the model.The output of the chi-square test is shown in Figure 6.  The first column is the Data Frame column number, the indexcolumn is the name of the dataset variable, and the Chi\_Square column is the calculated chi-square value.  Due to the length of the output, only the first 10 values are shown.

A screenshot of a computer

Description automatically generated

Figure 6. Output from the Chi-Square calculation.

The Lasso Regression (L1)method was also used from the sklearn library.  Regularization consists of adding a penalty to the different parameters of the machine learning model to reduce the freedom of the model and in other words to avoid overfitting. In linear model regularization, the penalty is applied over the coefficients that multiply each of the predictors. From the different types of regularization, Lasso or L1 has the property that can shrink some of the coefficients to zero. Therefore, that feature can be removed from the model.The output from our Lasso Regression is shown in Figure 7.  The first column is the Data Frame column number, the index column is the name of the dataset variable, and the L1 column is the Lasso Regression result. Like with the RFE output, the values of the L1 column are shown as True because they have not been eliminated in the Lasso Regression selection process.  In other words, these remaining variables are considered important to the overall predictive model.  The output below (in alphabetical order) shows all of the variables that returned a True value.

A screenshot of a computer

Description automatically generated

Figure 7. Output from Lasso Regression (L1) calculation.

A scoring table was constructed that tallied all of the different feature selection methods. From this table, I was able to determine which variables should be included in our model. The scoring table, listing only values with a final\_score above 2, is shown in Figure 8.

A screenshot of a computer

Description automatically generated

Figure 8. Scoring table of all feature selection results.

I will have to include Age and Player\_Efficiency\_Rating since they scored the highest. After that, however, we have many other variables that all returned a sum score of 2. I was able to eliminate some of these features by looking at the correlation matrix.

Once the features were scored a correlation matrix was used to test the features correlation, the table in Figure 9 is a heatmap showing the correlation of the features scoring at least a two.

A red and white graph

Description automatically generated with medium confidence

Figure 9. Output from initial Correlation Heatmap

It is clear in this heatmap that there is quite a bit of correlation between features. This isn’t surprising because a good basketball player is good at everything and a bad one is bad at everything. These are all important parts of the game so if someone is great at assists they are also likely great at field goals. That being said, I still want to remove as much unnecessary correlation as I can. First, remove all features marking “attempts”. They were too highly correlated with successes, so they didn’t add much value. The other feature that was removed was Minutes Played. This scored highly in our L1 test but it is very highly correlated with most features. That makes sense because the more time you play the more chances you have at blocking, rebounding, taking shots, etc. Removing these features gave us a heatmap with the 14 features in Figure 10.

A screenshot of a graph

Description automatically generated

Figure 10. Output from final Correlation Heatmap

Removing those features has removed quite a few of the highly correlated points. There is still some high correlation that we have left in for now. For example, Field Goals and Free Throws are highly correlated but field goals do not include free throws so there isn’t that direct relationship like in other features.

## Build/evaluate model

 One of the models that was used for this project was the Ordinary Least Squares (OLS) model.  The reason for selecting this model is that it is one of the most commonly used models and would serve as a good baseline with which we could compare the subsequent models.  The table below shows the results of this model.

OLS Model Fit Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Train RMSE** | **Train R2** | **Test RMSE** | **Test R2** |
| OLS | 4398500 | 0.63 | 5024129 | 0.57 |

The R2 values for the models indicate that we can account for about 60% of the salary prediction variability with the features we selected. The root mean square error (RMSE) values will serve as a reference point for the other linear models we are going to use. But this is just one of the five models that I used for this project.

### **Interpret The Results**

When summarizing the results, the exploratory data analysis was performed on the variables to help identify which ones might not be good predictors of salary. Plotting the variables as histograms helped to visualize each data point and determine which ones had a normal distribution. However, this still wasn't enough evidence to make a solid decision on which variables to use in the analysis. Several feature selection techniques were used to help with this decision, including random forest classifier, RFE, ExtraTreesClassifier, Chi-Square, Lasso regression, and a correlation matrix. These techniques were helpful in feature selection and several variables were removed from the analysis if their correlation value was relatively small when compared to other variables. In the end, these efforts led to a selection of 14 features that will be used to analyze player salaries. Not only that, having one of the models that was mentioned above was proven very helpful for this project. It also helped which players were either being under or overpaid.

**Milestone 3: Preliminary Analysis**

**Will I be able to answer the questions I wanted with the data I have?**

Yes, I have been able to answer the question, but I did have to use different models than what I originally planned when I did my proposal. I did models such as the random forest classifier, extra trees classifier, chi-square, and lasso regression. Using these types of models is what allowed me to look at how NBA players should be paid based on their performance along with answering if players are being overpaid or underpaid.

**What visualizations are especially useful for explaining my data?**

When it came to using visualizations for this semester's project I used different kinds such as histograms along with graphs which were helpful to answer the questions I wanted to explore for this project. It was because of using those to visualize my data that I was able to see the bigger picture for this project.

### **Do I need to adjust the data and/or driving questions?**

           This would be a yes since the original models I wrote in my proposal would not be the best suited for this project. The modes that I mentioned earlier were best suited for me to answer the questions. As for the questions, they were good ones since they are important to help many NBA teams figure out how much the players should be paid.

#### **Do I need to adjust my model/evaluation choices?**

Since I switched to those particular models it has helped me a lot when it comes to looking at the project. For example, I was able to look at how good certain players are within certain positions along with looking at certain factors such as free throws could affect a player's outcome. Using these specific models did help me a lot when it comes to looking at a player's stats along with seeing how much they get paid as well.

##### **Are my original expectations still reasonable?**

Yes, since my expectations were reasonable when it came to this particular project. My finding did make me see how many NBA teams could benefit from using this model to help figure out how much should players be paid based on their skills. But it can also help a player learn where they are the weakest and figure out how to improve their skills as well.

**Milestone 2: Proposal**

**Introduction**

When looking at the NBA is a very well-known sports association along with the MLB and the NFL. Not only that but the people that are within the NBA by being a player or part of the many teams can make a massive amount of money. This is thanks to the many skilled players such as Lebron James who can make many of the NBA team very profitable. But when it comes to these players that is one thing that is important which is how much the players are being paid. While some players within the NBA make more than others. This can cause a problem in some cases in which players are very overpaid while others are underpaid. One could say that a player should be paid based on their performance, which can help determine how much a player should get paid instead of being under or overpaid. This is where my project will come in handy.

**Problem Statement**

When looking at an NBA player should each player be paid based on performance? Are there any players that are being overpaid or any players that are being underpaid? These are the questions that I will be looking at during this semester's course project. By having these types of questions, I hope to find the answers by using predictive analysis.

**Data set**

The following data for this project will be looking at the NBA’s players' performance along with how much they are getting paid based on a data set from a website named Kaggle which is well known for looking up many data sets for many data scientists. To be more specific I will be looking at certain data such as the players along with what their position which I may use other websites in order to look for that type of data. Using these tools will prove very helpful to me find the answers needed for this semester's project.

**The process**

During this project, I would like the CRISP-DM process since I can use this process to not only answer the question for my project but also find a possible solution as well. Not only that it is like the scientific process which many people have used to figure out solutions to their problems. The data set will have a lot of information to look at but I would narrow on which data will be used in order to be successful during the semester project. Yes, it may seem like it is too much, so there will need to be a process to select certain information needed for this project. Examples of this would be what position the player plays along with their performance during the game as well. Having more specific information will not only help me but also be very valuable for the semester project as well.

##### **Model**

I would like to build a prediction model to help how players should be paid based on their performance. This model can have great applications, especially when it comes to recruiting new players into the NBA. The prediction model can be created by using different models such as random forest model, gradient, or Logistic regression. These types of models both have their pros and cons but they can be very helpful for me during this semester project. Creating the prediction model that I want to use will help analyze a player’s performance along with looking at how much a player should get paid. This would help many teams within the NBA figure out not only figure out how a player should get paid but also the teams would be able to have a workable budget needed to play these players as well.

**The plan**

The plan for this project is to keep up with the milestone’s requirements along with the peer review that will be done by my classmates. This will help me keep up with my progress in building the predictive model for this semester's project. Having good feedback on my Milestones will help me try to figure anything new that should be added for my semester project.

**Hope to learn**

           I hope to lean on how players are paid based on their performance during the games of the season along with using variables that will help the project as well. Not only that but I hope that the model that I am building can be used when it comes to paying future NBA players as well. This project can also be used as a tool for teams to build a decent budget for paying their players instead of running out of money due to overpaying players. The whole goal of the project is to see if players are getting paid fairly based on their performance. As the saying goes you want the most bang out of your buck. I hope to have a lot of success working on this project during the semester.

**Risk/ Consequences**

           There may be some issues when it comes to making the models such as having any missing data that may be helpful for the project. In that case, I would focus on certain data that will only help with the development of the model. Then would be other issues of having too much data to work with which can be overwhelming. Like I said before I would focus on certain types of data that can help the project. One last issue that could happen would be if a player does better during the season than the model would have predicted, that is another thing to consider as well. But this all comes down to my progress during the semester project. While data science may not be perfect it can be a very helpful tool specially for this semester project.

References

C., A. (2018, October 20). *NBA player stats 2017-18*. Kaggle. https://www.kaggle.com/datasets/acasalan/nba-player-stats-201718/data

HoopsHype. (n.d.). *These are the 2016/17 salaries of all NBA teams*. HoopsHype. https://hoopshype.com/salaries/2016-2017/