Dong Gwan Lee

Analyzing baseball statistics Using Different Regression Models: Ridge, Lasso, Elastic net, and Linear Regression

**Table of Contents**

[1 Abstract 4](#_Toc29840649)

[2 Introduction 6](#_Toc29840650)

[3 Background Research 7](#_Toc29840651)

[3.1 The formula of the cost function 8](#_Toc29840652)

[4 Methodology 12](#_Toc29840653)

[4.1 Data collection 12](#_Toc29840654)

[4.2 Developing initial model that predicts salary and why it was unsuccessful 14](#_Toc29840655)

[4.3 Different types of regressions using python 15](#_Toc29840656)

[4.3.1 Least square regression 15](#_Toc29840657)

[4.3.2 Lasso algorithm 16](#_Toc29840658)

[4.3.3 Ridge algorithm 18](#_Toc29840659)

[4.3.4 Elastic net algorithm 18](#_Toc29840660)

[4.4 P-test and statistical significance 19](#_Toc29840661)

[4.5 Eliminating dummy variables 19](#_Toc29840662)

[4.6 Model Validation Criteria 19](#_Toc29840663)

[4.7 Comparing Different Models-16 variables. 20](#_Toc29840664)

[4.7.1 Selecting variables by using p-test 20](#_Toc29840665)

[4.7.2 Refine Data 21](#_Toc29840666)

[4.7.3 Excel 23](#_Toc29840667)

[4.7.4 Lasso 24](#_Toc29840668)

[4.7.5 Ridge 25](#_Toc29840669)

[4.7.6 Elasticnet 25](#_Toc29840670)

[4.7.7 Overall comparison 26](#_Toc29840671)

[4.8 Comparing different models- 39 variables 26](#_Toc29840672)

[4.8.1 Selecting variables with using P-test 27](#_Toc29840673)

[4.8.2 Refine Data 29](#_Toc29840675)

[4.8.3 Excel 30](#_Toc29840676)

[4.8.4 Lasso 31](#_Toc29840677)

[4.8.5 Ridge 31](#_Toc29840678)

[4.8.6 Elastic net 32](#_Toc29840679)

[4.8.7 Overall comparison 32](#_Toc29840680)

[5 Results 34](#_Toc29840681)

[6 Lessons learned 35](#_Toc29840682)

[7 Future work 37](#_Toc29840683)

[8 Bibliography 38](#_Toc29840684)

[9 Appendix 40](#_Toc29840685)

# Abstract

In this paper, I compare the performance of four regression models, Ridge, Lasso, Elastic net, and linear regression, in predicting ERA, which is the most commonly accepted statistics for evaluating pitchers. model: ERA is defined as the number of earned runs a pitcher allows per nine innings. In common, ERA is used as an ultimate representation of the overall performance of pitchers. My goal is trying to figure out whether ERA is legitimate statistics to represent pitchers’ performance or not. If the model predicts ERA well with other pitcher statistics, ERA is a good statistic to weigh players’ value. I decided to use other pitcher performance statistics, such as average hits allowed, as predictors. However, not all variables are necessary or relevant to this task. For instance, the variables of game played, and game started are redundant for starting pitchers because starting pitchers always pitch from the first inning. Furthermore, pitchers’ batting average is unlikely to be predictive of their performance as pitchers and using this variable might ultimately undermine the accuracy of the prediction models. Even though I manually took out redundant variables I saw, there might be some variables that doesn’t appear easily. However, different techniques have been introduced to address this problem of variable selection. By comparing the prediction performance of different regression models, I will draw a conclusion about which regression model is most effective under which circumstances.

# Introduction

Regression models are commonly used to make predictions about future data or unknown data. To do this, they build regression models to find potential relationships between the explanatory variables and the response variable. Traditionally, multivariable regression is used when there is more than one explanatory variable, or predictors, that are used to predict the response variable; however, the performance of regression models can be undermined by potential multicollinearity [[1]](#footnote-1)among explanatory variables. Also, sometimes regression model can overfit the data, which is undermines the whole purpose of making the regression model. Fortunately, there are lots of regression method that solves these problems. My goal is to explore different kinds of regression method and apply the different algorithms to predict ERA from the baseball statistics.

# Background Research

Machine learning is a method of data analysis that automates analytical model building. It is a branch of [artificial intelligence](https://www.sas.com/en_us/insights/analytics/what-is-artificial-intelligence.html) based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.[[2]](#footnote-2)

The problem I am trying to solve is that of human misjudgment of players’ value based on inconsequential factors. If I build a model with machine learning, it is going to be more effective because the model will be developed without human intervention which might input some biases.

|  |  |
| --- | --- |
| Figure 1-1[[3]](#footnote-3) | Figure 1-2[[4]](#footnote-4) |

.

Figure 1-1 shows the list of salaries for each starting pitcher. Figure 1-2 shows the list of performance statistics for each player.

I will use multivariable linear regression to build a model that predicts salary. The X players’ performance stats. Y values will be the parameter value that weighs each performance stat. Here is where machine learning comes in.

## The formula of the cost function

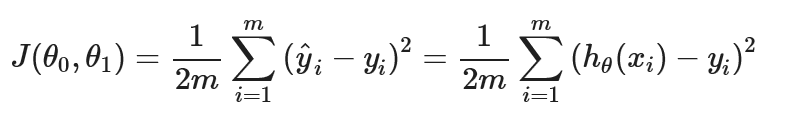


Figure 2

Figure 2: By minimizing the cost function, the model will increase its accuracy by minimizing the cost function. This function adds up the difference between the model’s predicted salary with the real baseball player’s salary for each player.

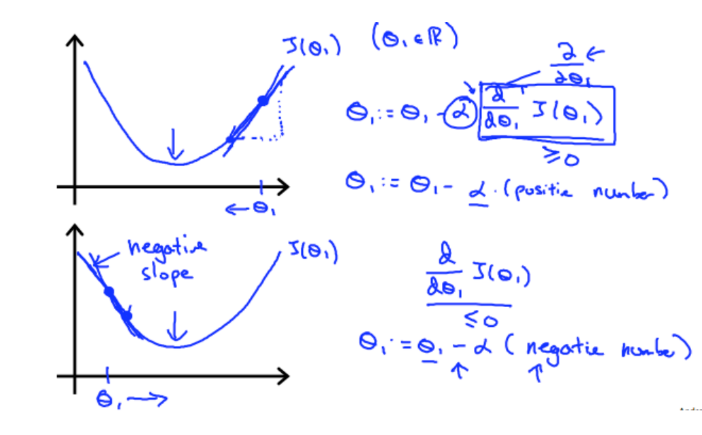


Figure 2-2

Figure 2-2 shows how machine learning attains a parameter value that gives me minimal cost function value. The machine will subtract the slope of that value until it reaches a minimal cost function point. This method is called gradient descent.

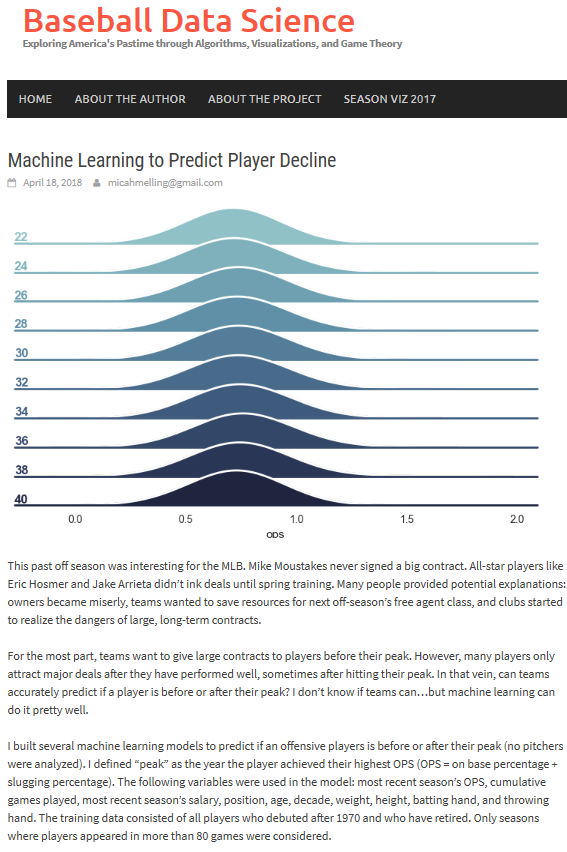


Figure 3-1[[5]](#footnote-5)

Figure 3-1: Fortunately, there was a similar study that predicted a player’s decline using machine learning. However, this study was a little different because it predicted players’ decline whereas my model predicts players’ value. The study only take account for 5 variables, but my model will account for more than 40 variables.

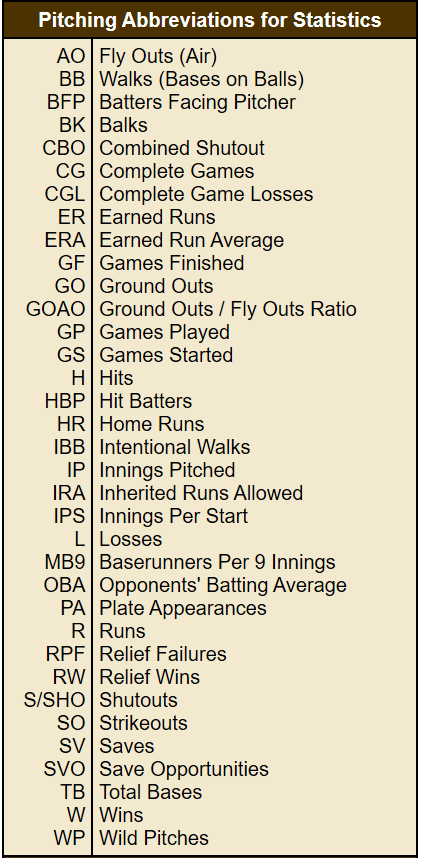


Figure 3-2[[6]](#footnote-6)

Figure 3-2: This is pitching abbreviations I collected from the website. This will help figure out what does each variable mean.

# Methodology

## Data collection

I will use the data of 55 out of 67 MLB starting pitchers who pitched for more than 162 innings during the 2019 season to train my models. I will use every variable that explains the pitchers’ performance (43 variables) as predictors. Originally, I wanted to predict the pitchers’ guaranteed minimum salary for the 2019 season given their performance statistics, both of which are available on the MLB website.



Figure 4

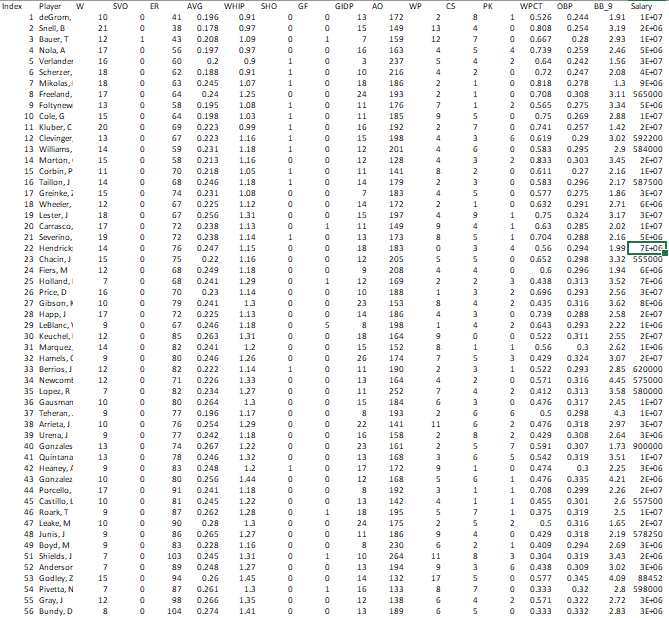


Figure 5

Figure 4: I learned how to code first because I could not perform machine learning without using computer programming. It took 24 hours.

Figure 5: I used a random number generator to pick 16 variables out of 43 variables because excel cannot account for more than 16 variables when I make a regression model though python can.

However, I later realized that in order to find a correlation between salary and stats, I should compare this year’s salary with last year’s stats, instead of using this year’s stats for each player. So, I instead used the players’ 2018-year stats to predict their 2019 salary.

## Developing initial model that predicts salary and why it was unsuccessful

I randomly selected 5 out of 50 pitchers to use as a test data, on which I can evaluate the model’s accuracy. I trained the models on the data of remaining 45 pitchers. With those players, I compared the model’s prediction salary and salary that they received.

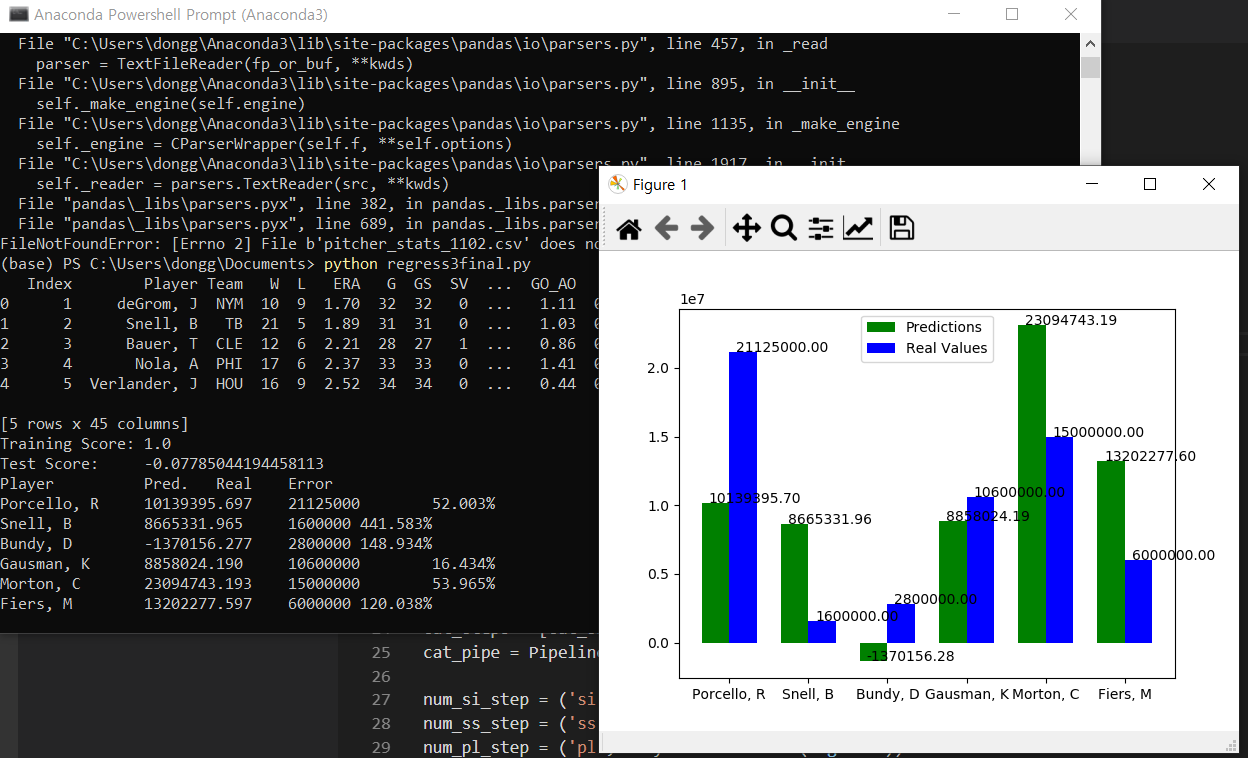


Figure 6

However, I found out that the model is not good at predicting the player’s salary at all (Figure 6). Initially, if the error rate was less than 10 %, I was going to label each player by overvalued player and undervalued player. However, the error was so large that the model does not predict close to real salary values at all. Also, the test score (R^2 score) is -0.07 percent, which means R^2 tells us that this model is not legitimate at all.

I pondered about it, and I realized that this happened because every player gets a different type of contract regarding salary. For example, a player who is in FA (free agents) can sign a guaranteed 4/3-year contract despite being a poor player, while there are international players who get paid less despite being some of the best pitchers on the MLB. When it comes to salary, every player is in different situations, so performance stats were insufficient for the models to predict well.

## Different types of regressions using python

Although I carried out my initial analysis on Excel, I chose to use Python because it allows me to build different models other than multivariate linear regression.

However, using python to build the model had some advantages over using excel.

I can build models using all variables in python while I can only use 16 predictors in excel.

I can use different types of algorithms other than linear regression.

Taking advantage of the fact that I can use python, I will use different models for the new task: Lasso, Ridge, and Elastic net. After I build the model using those algorithms, I am going to compare accuracy with the multivariable linear regression model built on excel. The 4 types of cost function are used to determine the formula of the regression.

### Least square regression

Least square regression is the most commonly used regression method. The Excel multivariable linear regression uses this regression.

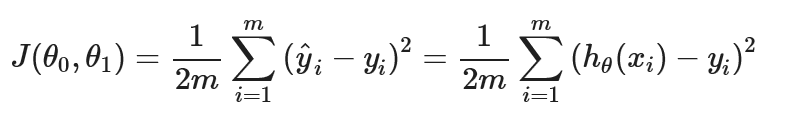


Figure 7

Figure 7: This is the cost function of the Least square regression. The goal of the this Least square regression is to minimize the cost function. This function is sum of the square of the difference between the predicted y value and real y value. The lower the cost function is , the more accurate the function is.

### Lasso algorithm

Linear regression uses every variable as a predictor, but lasso selects a subset of the input predictors, reducing the coefficients of others to zero.

The benefits of the lasso algorithm: Lasso algorithm is used to prevent overfitting on the training data and to resolve the multicollinearity problem. Lasso regression can also eliminate variables that are irrelevant to the task.

It uses a different of cost function to determine the weight for each data value.

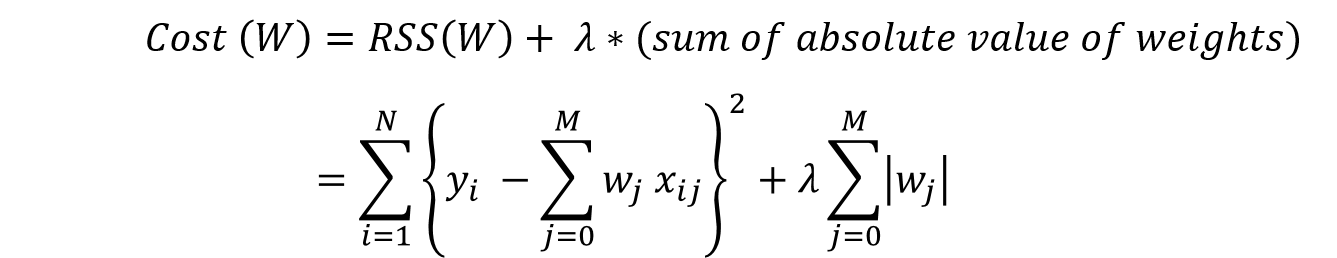


Figure 8

Figure 8 shows that the left side of the equation is the same as finding least square regression in linear regression. The goal for lasso algorithm is also the same: minimizing the cost function. The important part is that the cost function is now defined as the residual sum of squares and a penalty term, λ \* absolute value of weights. If some features are not useful enough in minimizing the RSS, their weight will be set to zero in order to lower the overall cost function. λ is a parameter I can set. If I select a large value for λ, the model will set weights of more variables to 0 because I am giving more penalty for weights of variables. If I lower the λ value, then I give less penalty for giving variable weights, therefore there is going to be more variables that will give weights.

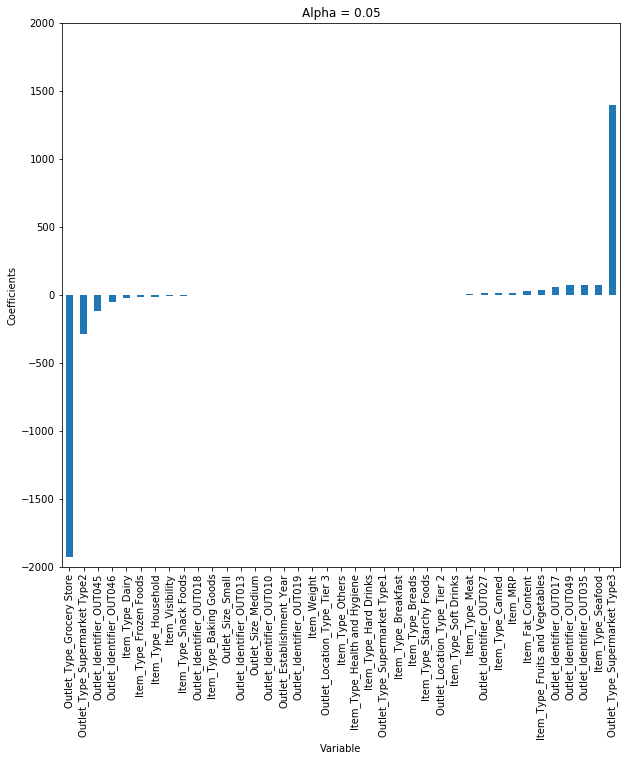


Figure 9-1

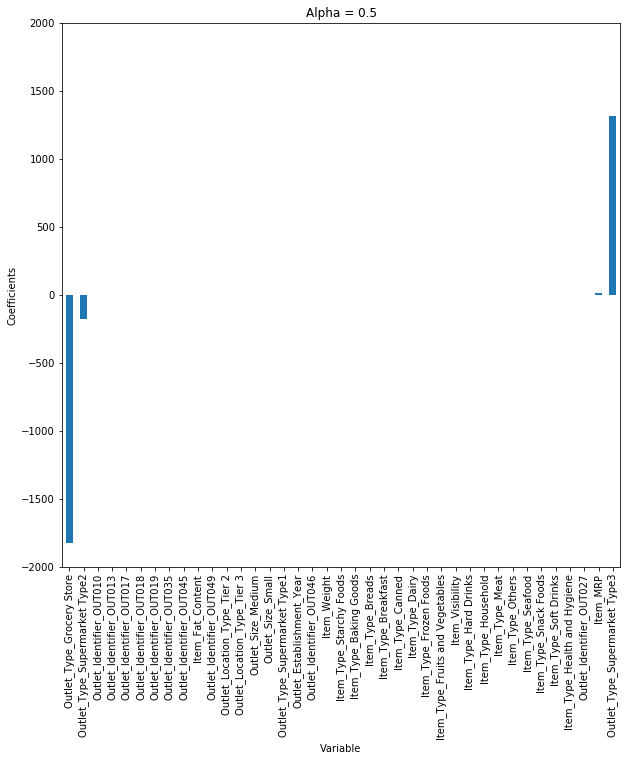


Figure 9-2

Figure 9-1, 9-2: these figures supports my explanation regarding Lasso’s variable selection according to lambda value because figures justifies that the bigger lambda value gets, the more the lasso penalizes variables, resulting in selecting less variables.

### Ridge algorithm

Ridge algorithm differs from the simple linear regression in that ridge shrinks coefficients by penalizing the feature weights values. It also differs from the Lasso algorithm because it only minimizes the coefficients instead of eliminating variables. Ridge algorithm is RSS (least square) + λ \* sum of squares of weights, whereas lasso is RSS + λ\* sum of absolute value of weights (Figure 10).

Benefits of Ridge algorithm is it reduces variance[[7]](#footnote-7), and it prevents overfitting.

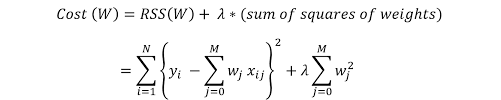


Figure 10

### Elastic net algorithm

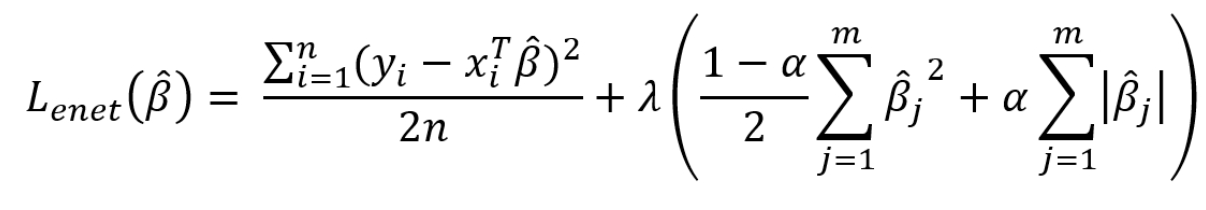


Figure 11

Elastic net algorithm is an algorithm that combines the lasso algorithm and the ridge algorithm. Since it shrinks and eliminates variables, using this algorithm confers me the advantages of both Ridge and the Lasso(Figure 11).

## P-test and statistical significance

The p-test is testing whether independent variables can occurrence by an accident. The P-test can provide the evidence whether I should accept or reject the variable. The smaller the p-value, the higher the credibility is. We commonly say under 0.05(under 5%) is statistically significant, which means that the variable would not occur by an accident. So, I am only going to use variables that are only under 0.05 p-value.

## Eliminating dummy variables

There were two variables that would not help predicting ERA. One was GS (game started). This variable isn’t important because since all the starting pitchers starts from the beginning of the game. So, it overlaps with G (games played). Second was HLD (hold). This variable is only for relief pitcher, because this count when relief pitcher keeps the game into winning. So, I eliminated those variables in the dataset.

## Model Validation Criteria

I used two Criteria to evaluate the accuracy of the model, R2 score and Average Error. R2 score is a statistic that will give some information about how well the model fits. To compare more, I also introduce the concept of average error. Average error is the percentage error between the predicted ERA and Actual ERA. I picked 6 players to determine the average ERA error rate, which will **not** be trained on the datasets. The formula of the average error rate is (actual ERA- predicted ERA) / (predicted ERA). The players I chose are Mikolas, Berrios, Fiers, Price, Castillo, Cole. With the excel regression, I am going to take those 6 players out from the dataset, and later I will calculate their predicted ERA using the regression formula and find an error. For python, I am going to use the code below to separate 6 players from the dataset and compare the ERA error rate.

Train, X\_test, y\_train, y\_test = train\_test\_split (X\_all, y\_all, test\_size=0.1, random\_state=321)

## Comparing Different Models-16 variables.

Since Excel model can only account for 16 variables at a time, I started with using 16 variables. I used random number generator to pick 16 out of 39 variables. The variables are W(win) SVO (save opportunity), ER(earned run), AVG(batting average with that pitcher), WHIP (walks plus hits per inning pitched), SHO(shutout), GF(games finished), GIDP(ground into double play), AO(air outs), WP(wild pitch), CS(Caught stealing), PK(pickoff),WPCT(winning percentage) OBP(On Base Percentage), BB\_9(Walks per 9 innings), and Salary.

### Selecting variables by using p-test

A close up of text on a white background

Description automatically generated

Figure 12

Figure 12: I regressed using 16 variables that I randomly selected. There were 3 variables that pass though the P-test. The variables that passed are GIDP, AO, and OBP. I am going to only use those variables to compare the models (excel, ridge, lasso, elastic net).

### Refine Data

A close up of a piece of paper

Description automatically generated

Figure 13

Figure 13: I deleted all the variables except those three variables to input the dataset in the python. I saved the excel document as CSV (comma separated value) in order to make excel analyze the data.

A screenshot of a cell phone

Description automatically generated

Figure 14

Figure 14: I intentionally separated the 6 players out of the dataset. After I built the regression formula that I determined without those 6 players, I substituted their data into the regression formula to get a predicted ERA.

### Excel

These figures below are the output for the model using 3 variables (out of 16 variables).

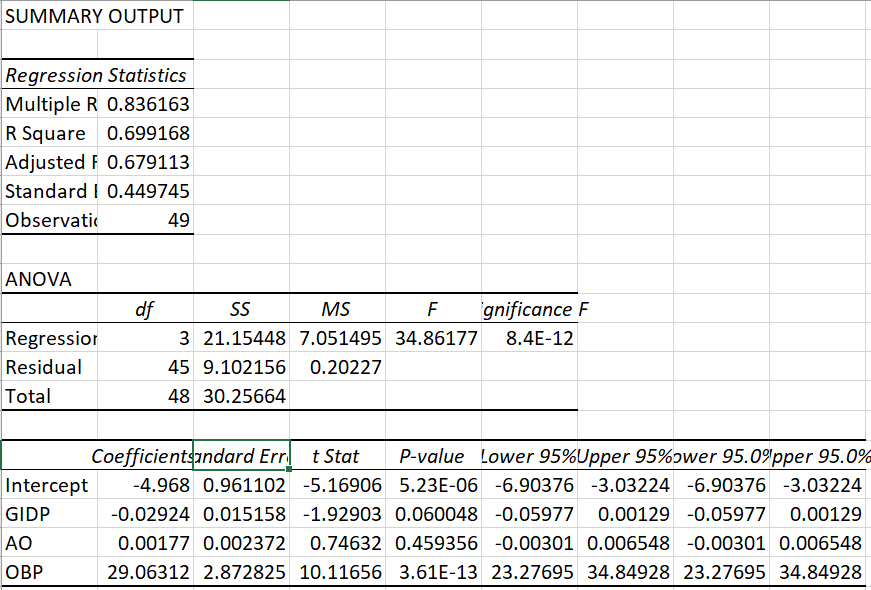


Figure 15

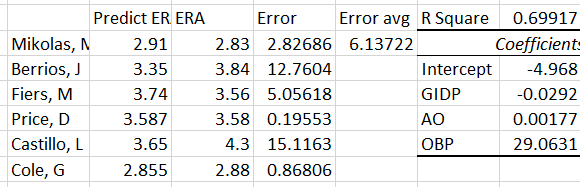


Figure 16

Figure 15, 16: These are the outputs for the excel regression with using 3 variables. It had 0.69 R2 value, and the error average was 6.13%.

### Lasso

Even though I could make a Least square regression model using Excel, I had to use python to use regularization regression model such as Lasso, Ridge, and Elastic Net algorithm.

With the code I wrote in the appendix, I added the additional line of code to make the model with using lasso algorithm.

regress = linear\_model. Lasso(alpha=0.01)

I inputted the alpha value equal to 0.01.

GIDP: -0.12372702182569675  
AO: 0.042111192615565875  
OBP: 0.6579057207343265  
Total 3 non-zero feature(s)  
R2 score: 0.6675342759075515

Player      Pred(ERA). Real(ERA) Error(%)  
Mikolas, M 2.937 2.83 3.768%   
Berrios, J 3.555 3.84 7.416%   
Fiers, M 3.721 3.56 4.516%   
Price, D 3.579 3.58 0.032%   
Castillo, L 3.657 4.3 14.946%   
Cole, G 2.866 2.88 0.502%

This is the result for using Lasso regression. It used all 3 variables since it had not enough variables. The average Error for the era with 6 pitchers were 5.19%. R2 score was 0.667

### Ridge

With the code I wrote in the appendix, I added the additional line of code to make the model with using Ridge algorithm.

regress = linear\_model. Ridge(alpha=0.01)

Same as Lasso, I used alpha= 0.01

GIDP: -0.13588875874839543  
AO: 0.04961351126935333  
OBP: 0.6725347999266862  
Total 3 non-zero feature(s)  
R2 score: 0.6634338574742956

Player      Pred(ERA). Real Error  
Mikolas, M 2.915 2.83 2.995%   
Berrios, J 3.562 3.84 7.234%   
Fiers, M 3.740 3.56 5.048%   
Price, D 3.588 3.58 0.221%   
Castillo, L 3.651 4.3 15.087%   
Cole, G 2.856 2.88 0.832%

R2 score of ridge model was 0.663. and average error rate was 5.236%

### Elasticnet

With the code I wrote in the appendix, I added the additional line of code to make the model with using ElasticNet algorithm.

regress = linear\_model. ElasticNet(alpha=0.01)

For every algorithm, I used identical alpha value in order to make it fair.

GIDP: -0.1279238626178823  
AO: 0.045736310122311434  
OBP: 0.6615105102373245  
Total 3 non-zero feature(s)  
R2 score: 0.6649305589773236

Player     Pred(ERA). Real Error  
Mikolas, M 2.931 2.83 3.556%   
Berrios, J 3.558 3.84 7.332%   
Fiers, M 3.729 3.56 4.735%   
Price, D 3.583 3.58 0.075%   
Castillo, L 3.654 4.3 15.029%   
Cole, G 2.864 2.88 0.543%

Average error rate was 5.211% and The R2 score was 0.664.

### Overall comparison

R2 score comparison: Excel linear regression > Lasso > Elastic net > Ridge

Error avg comparison: Excel linear regression > Ridge > Elastic net > Lasso

Both Excel linear regression had both the best R2 score and Error avg value.

## Comparing different models- 39 variables

I did another experiment that uses all 39 variables. I wondered if the model had ample number of explanatory variables, Lasso could be more accurate excluding dummy variables.

### **Selecting variables with using P-test**

A screenshot of a cell phone

Description automatically generated

Figure 17-1

A screenshot of text

Description automatically generated

**Figure 17-2**

A screenshot of text

Description automatically generated

Figure 17-3

Figure 17-1, 17-2, 17-3: I selected variables that are statistically significant using p-test. Since excel regression cannot only take 16 variables at a time, I did regress 3 times to find p values for every variable. The variables that had under 0.05 p-values are GIDP (ground into double play), WPCT (winning percentage), SV (save opportunity), IP (inning pitched). I am going to use those 4 variables in Excel, Ridge, Lass, and Elastic Net to compare their accuracy.

### Refine Data

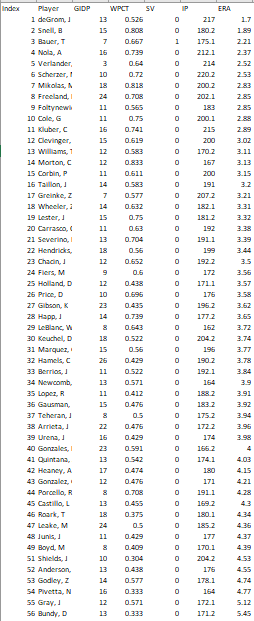


Figure 18

Figure 18: I refine data by putting index numbers and took out all the variables except for those four variables. I am going to input this into the python to build a model that uses Lasso, Ridge, Elastic Net regression.

A screenshot of a cell phone

Description automatically generated

Figure 19

Figure 19: I intentionally separated the 6 players out of the dataset. After I built the regression formula that I got without those 6 players, I substituted their data into the regression formula to get a predicted ERA.

### Excel

These are the output of the Least square regression model using 4 variables(out of 39 variables).

A screenshot of text

Description automatically generated

Figure 20

Figure 20: This is my result for my Excel regression using 4 variables. The R2 score for the excel regression is 0.617(there are no units for R2). The average error rate for 6 players for the Excel is 5.469%.

### Lasso

I used the line of code below to make a model that uses Lasso regression model.

regress = linear\_model. Lasso(alpha=0.01)

WPCT: -0.39975576933012946  
SV: -0.1713388401305614  
IP: -0.32261375211487975  
Total 3 non-zero feature(s)

R2 Score: 0.8641396907665297  
Player      Pred (ERA) Real Error  
Mikolas, M 2.557 2.83 9.637%   
Berrios, J 3.643 3.84 5.131%   
Fiers, M 3.831 3.56 7.601%   
Price, D 3.449 3.58 3.653%   
Castillo, L 4.338 4.3 0.874%   
Cole, G 2.769 2.88 3.848%

The R2 score of the Lasso regression was 0.864. Lasso used 3 variables out of 4 variables. The average error rate of the Lasso algorithm is 5.124%.

### Ridge

I used this line of code to make a regression model using Ridge algorithm

regress = linear\_model. Ridge(alpha=0.01)

GIDP : -0.006540900637083309  
WPCT : -0.40737654824809727  
SV : -0.18158608340892726  
IP : -0.3315660051726509  
Total 4 non-zero feature(s)  
R2 Score: 0.8478898127266364  
Player      Pred. Real Error  
Mikolas, M 2.530 2.83 10.617%   
Berrios, J 3.647 3.84 5.018%   
Fiers, M 3.845 3.56 8.006%   
Price, D 3.454 3.58 3.513%   
Castillo, L 4.357 4.3 1.315%   
Cole, G 2.755 2.88 4.328%

Though Ridge algorithm used 4 variables, the coefficient variables are smaller. The average error rate was 5.466%. The R2 score of the Ridge algorithm is 0.847.

### Elastic net

This line of code was used to make a regression model using Elastic Net regression:

regress = linear\_model. ElasticNet(alpha=0.01)

WPCT: -0.4015446268116681  
SV: -0.17513697182299512  
IP: -0.3258519371454889  
Total 3 non-zero feature(s)  
  
R2 Score: 0.8601178452843824  
Player      Pred (ERA). Real Error  
Mikolas, M 2.551 2.83 9.850%   
Berrios, J 3.643 3.84 5.136%   
Fiers, M 3.834 3.56 7.686%   
Price, D 3.450 3.58 3.630%   
Castillo, L 4.343 4.3 1.004%   
Cole, G 2.764 2.88 4.024%

The Elastic Net algorithm used 3 variables out of 4 variables. The R2 score was 0.86. The average error rate was 5.221%.

### Overall comparison

R2 score: Lasso > Elastic Net > Ridge > Excel linear regression

Average error rate: Excel linear regression > Lasso > Elastic Net > Ridge

# Results

These are the results of the R2 score and the average error for using 3 variables (out of 16 variables) and 4 variables(out of 39 variables).

Using 3 variables (out of 16 variables)

Excel: 0.69 R2 score and 6.13% average error rate

Lasso: 0.667 R2 score and 5.19% average error rate

Ridge: 0.663 R2 score and 5.236% average error rate

Elastic Net: 0.662 R2 score and 5.211% average error rate

R2 score comparison: Excel linear regression > Lasso > Elastic net > Ridge

Error avg comparison: Excel linear regression > Ridge > Elastic net > Lasso

Using 4 variables (out of 39 variables)

Excel: 0.61 R2 score and 5.469 % average error rate

Lasso: 0.864 R2 score and 5.124% average error rate

Ridge: 0.847 R2 score and 5.466% average error rate

Elastic Net: 0.86 R2 score and 5.221% average error rate

R2 score: Lasso > Elastic Net > Ridge > Excel linear regression

Average error rate: Lasso > Elastic Net > Ridge> Excel linear regression

# Lessons learned

In my project, I learned how multivariable linear regression model fits the coefficients for the variables. Furthermore, I explored various types of algorithms (Ridge, Lasso, Elastic net) by comparing their performance in predicting the ERA given the performance statistics. Now I understand the role and benefits of each algorithm.

I started with the 16 variables for the first experiment. However, I left out with 3 variables after I finished the p-test. When I compared models using 3 variables, in terms of R2 score, Excel algorithm worked best, Lasso as a second, Elastic Net as the third, and Ridge regression as 4th. In terms of Error avg, Excel linear regression was best, Ridge worked as a second, Elastic net was third, and Lasso was the last. Every regression model used all 3 variables since they don’t have enough variables.

For the second experiment, I used all 39 variables. When I conducted p-test using excel, there were only 4 variables that passed p-test. In terms of the variables, Lasso and Elastic Net both eliminated same variable (GIDP) to make the model better. Because there were more variables now(slightly), Lasso which eliminates ineffective variable worked best in terms of R2 score. Elastic net was second, Ridge was third, and Excel linear regression was forth. Also, with the average error rate, Lasso worked best, Elastic Net worked as a second, and Ridge worked as a third, and Excel linear regression worked the last. Regression model worked better than the excel least square regression because there were more variables.

I learned that there are lots of multicollinearity within the baseball variables since there were only few variables that passed the p-test. I also learned that regularization algorithms (Ridge, Lasso, Elastic Net) do not work well when there are not enough variable. However, regularization algorithm works well when there were more variables. I infer that when there are more variables, regularization algorithm works better. In modern society, since we use “big data” that have tons of variables to analyze, I think the frequency of using regularization algorithm will rise as well.

Throughout the project, there were lots of handwork to make a model. I can’t imagine how much effort they put in in their job. I should seriously think about the work effort if I later want to be a data scientist.

# Future work

This data doesn’t add as the season continues. Next time, I want to build a model using two or more seasons to make a model better. Also, my initial goal was to find overvalued players and undervalued players using a model that predicts salary. However, the model was not accurate because every player gets a different type of contract regarding salary. To solve this problem, I will later explore ways to account categorical variables such as their ethnicity, type of contract they have, or whether they are in the FA contract or not, health status. Besides, to improve accuracy, I will account for more variables than merely pitcher performance stats when predicting salary. I will add variables like age, height, or weight that are not included on the pitcher performance statistics.

# Bibliography

* Machine learning. (n.d.). Retrieved September 6, 2019, from <https://www.coursera>.org/learn/machine-learning/home/welcome
* [MLB salary]. (n.d.). Retrieved September 20, 2019, from http://mlb.mlb.com/stats/sortable.jsp#elem=%5Bobject+Object%5D&tab\_level=child&click\_text=Sortable+Player+pitching&game\_type='R'&season=2019&season\_type=ANY&league\_code='MLB'&sectionType=sp&statType=pitching&page=1&ts=1568941807909
* MONEYBALL. (n.d.). Retrieved from https://www.sonypictures.com/movies/moneyball
* Statistics. (n.d.). Retrieved from <http://mlb.mlb.com/stats/sortable.jsp#elem=%5Bobject+Object%5D&tab_level=child&click_text=Sortable+Player+hitting&game_type='R'&season=2019&season_type=ANY&league_code='MLB'&sectionType=sp&statType=hitting&page=1&ts=1567731093937>
* Micahmelling@gmail.com. *Baseball Data Science*, 18 Apr. 2018, <http://www.baseballdatascience.com/machine-learning-to-predict-player-decline/>.
* *Multiple Régression*, <https://www.stapplet.com/multreg.html>.
* “Full Catalog.” *Codecademy*, <https://www.codecademy.com/learn>.
* Baseball Almanac, Inc. *Baseball Abbreviations*, <https://www.baseball-almanac.com/stats4.shtml>.
* Jain, Aarshay, et al. “A Complete Tutorial on Ridge and Lasso Regression in Python.” *Analytics Vidhya*, 3 Sept. 2019, <https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-ridge-lasso-regression-python/>.
* “An Introduction to Ridge, Lasso, and Elastic Net Regression.” *By Gurkamal Deol*, <https://hackernoon.com/an-introduction-to-ridge-lasso-and-elastic-net-regression-cca60b4b934f>.
* Ganti, Akhilesh. “P-Test Definition.” *Investopedia*, Investopedia, 18 Nov. 2019, https://www.investopedia.com/terms/p/p-test.asp.

# Appendix

These are the code that I used for the project.

import pandas as pd

import numpy as np

from matplotlib import pyplot as plt

from sklearn. preprocessing import OneHotEncoder

from sklearn. impute import SimpleImputer

from sklearn. compose import ColumnTransformer

from sklearn. pipeline import Pipeline

from sklearn. preprocessing import StandardScaler

from sklearn import linear\_model

from sklearn. preprocessing import PolynomialFeatures

from sklearn. model\_selection import train\_test\_split

# data\_all = pd. read\_csv('pitcher\_stats\_43variables.csv')

data\_all = pd. read\_csv('pitcher\_stats\_42variables.csv')

# y\_all = data\_all.pop('Salary'). values

print (data\_all. head ())

y\_all = data\_all.pop('ERA'). values

cat\_si\_step = ('si', SimpleImputer (strategy='constant',

fill\_value='MISSING'))

cat\_ohe\_step = ('ohe', OneHotEncoder (sparse=False,

handle\_unknown='ignore'))

cat\_steps = [cat\_si\_step, cat\_ohe\_step]

cat\_pipe = Pipeline(cat\_steps)

num\_si\_step = ('si', SimpleImputer(strategy='median'))

num\_ss\_step = ('ss', StandardScaler ())

num\_pl\_step = ('pl', PolynomialFeatures (degree=1, include\_bias=False))

num\_steps = [num\_si\_step, num\_ss\_step, num\_pl\_step]

num\_pipe = Pipeline(num\_steps)

all\_cols = data\_all. columns. values. tolist ()

not\_num\_cols = ['Index', 'Player', 'Team']

num\_cols = [col for col in all\_cols if col not in not\_num\_cols]

# cat\_cols = ['Team']

cat\_cols = []

# transformers = [('cat', cat\_pipe, cat\_cols),

# ('num', num\_pipe, num\_cols)]

transformers = [('num', num\_pipe, num\_cols)]

ct = ColumnTransformer(transformers=transformers)

X\_all = ct.fit\_transform(data\_all)

index\_col = data\_all['Index']. values. reshape (-1, 1)

X\_all = np. hstack ((index\_col, X\_all))

X\_train, X\_test, y\_train, y\_test = train\_test\_split (

X\_all, y\_all, test\_size=0.1, random\_state=321)

# 1381 0.83

index\_test = X\_test [: 0]

X\_train = np. delete (X\_train, 0, 1)

X\_test = np. delete (X\_test, 0, 1)

#regress = linear\_model. Lasso(alpha=0.01)

#regress = linear\_model. Ridge(alpha=0.01)

regress = linear\_model. ElasticNet(alpha=0.01)

# regress = linear\_model. LassoCV(cv=5)

# regress = linear\_model. RidgeCV(cv=5)

# regress = linear\_model. LinearRegression ()

# regress = linear\_model. ElasticNetCV(cv=5)

regress.fit(X\_train, y\_train)

predictions = regress. predict(X\_test)

print('Coefficients:')

print (regress. coef\_. shape)

print(len(num\_cols))

n\_valid\_feature = 0

for name, value in zip (num\_cols, regress. coef\_):

if value! = 0:

n\_valid\_feature = n\_valid\_feature + 1

print (name, ':', value)

print (f'Total {n\_valid\_feature} non-zero feature(s)')

print ('R^2 score:\t' + str (regress. score (X\_test, y\_test)))

index\_test = [int(i-1) for i in index\_test]

name\_lst = data\_all.loc [index\_test, 'Player']

print ('Player \tPred.\tReal\tError')

for i, v in enumerate(name\_lst):

pred = predictions[i]

real = y\_test[i]

err = abs ((pred - real) / real \* 100)

print('{name}\t{pred}\t{real}\t{err}% ‘. format (

name=v, pred='{:.3f}’. format(pred), real=real,

err='{:.3f}'.format(err)))

n\_groups = len(index\_test)

index = np. arange(n\_groups)

bar\_width = 0.35

rects1 = plt.bar (index, predictions, bar\_width,

color='g',

label='Predictions')

for i, v in enumerate(predictions):

plt.text(index[i]-bar\_width/4, v+0.1, '{:.2f}’. format(v))

rects2 = plt.bar (index + bar\_width, y\_test, bar\_width,

color='b',

label='Real Values')

for i, v in enumerate(y\_test):

plt.text(index[i]+3\*bar\_width/4, v+0.1, '{:.2f}’. format(v))

plt. xticks (index + bar\_width/2, name\_lst)

plt. legend ()

plt. show ()

1. **multicollinearity** (also collinearity) is a phenomenon in which one predictor variable in a multiple regression model can be linearly predicted from the others with a substantial degree of accuracy (Wikipedia). [↑](#footnote-ref-1)
2. “Machine Learning: What It Is and Why It Matters.” *SAS*, https://www.sas.com/en\_us/insights/analytics/machine-learning.html. [↑](#footnote-ref-2)
3. “Sortable Player Stats.” Major League Baseball, http://mlb.mlb.com/stats/ [↑](#footnote-ref-3)
4. “Machine Learning: What It Is and Why It Matters.” SAS, https://www.sas.com/en\_us/insights/analytics/machine-learning.html. [↑](#footnote-ref-4)
5. Micahmelling@gmail.com. Baseball Data Science, 18 Apr. 2018, http://www.baseballdatascience.com/machine-learning-to-predict-player-decline/. [↑](#footnote-ref-5)
6. Baseball Almanac, Inc. *Baseball Abbreviations*, https://www.baseball-almanac.com/stats4.shtml. [↑](#footnote-ref-6)
7. The variance is an error from sensitivity to small fluctuations in the training set. It is a measure of spread or variations in our predictions. High variance can cause an algorithm to model the random noise in the training data, rather than the intended outputs (overfitting) (from Data Science Central). [↑](#footnote-ref-7)