The Prediction of Salary Based on Baseball Statistics
Multiple Linear Regression using R and Python

Ryan Fox, rfox2@bellarmine.edu
Caitlin Sizemore, csizemore@bellarmine.edu

ABSTRACT
This exploratory data analysis will attempt to find a connection between certain baseball statistics and player salaries. We conducted this data analysis using Anaconda Python and R Studio. Using these languages, we were able to perform linear regression and multiple linear regression tests. Our data was split into training, testing, and validation sets. This allowed us to find our R-Squared and P-values. It also allowed us to create predictions based on our analysis. We compared our predictions to actual data values to determine whether our model was a good predictor.

I. INTRODUCTION
The data set was obtained from Kaggle.com and contains information concerning batting data from each player's 1986 and 1987 seasons along with career totals. It also contains information regarding league and salary. Using Python and R Studio, we predicted the salaries of each player based upon their statistics.

II. BACKGROUND
A. Data Set Description
The dataset was found on Kaggle.com and chosen due to an interest in baseball. It consists of data from multiple sources concerning the 1986 and 1987 seasons of Major League Baseball. Specifically, the StatLib library of Carnegie Mellon University, Sports Illustrated issue April 20, 1987, and the 1987 Baseball Encyclopedia Update. This dataset consists of batting data particular to the 1986 and 1987 seasons, but also includes data for each player's career. It also includes information about each player's league in 1986 and 1987, errors from the 1986 season, and each player's salary.

B. Machine Learning Model
The Multiple Linear Regression model uses mathematical figures to make predictions. Specifically, it uses independent variables to predict the chosen dependent variable. Using the data provided, it generates coefficients and intercepts, creating a formula that will predict the dependent with the given dependents. In addition, Mean Squared Error and the R-Squared value analyze the accuracy of the regression. Mean Squared Error estimates the average squared difference between each prediction and the actual value. The R-Squared value shows how much variance within the dependent variable that can be explained by the independent variables. The closer this value is to one, the more accurate the model is.

III. EXPLORATORY ANALYSIS
This dataset contains 322 samples with 20 columns of various data types. The Salary column contained 59 missing values. When plotting, it was discovered that this column was right skewed. Using median, the missing values were filled in. No other parts of the dataset contained missing values and did not require plotting for investigation.

Table 1: Data Types

Variable Name	*Data Type*
AtBat	int64, continuous
Hits	int64, continuous
HmRun	int64, continuous
Runs	int64, continuous
RBI	int64, continuous
Walks	int64, continuous

Years	int64, continuous
CAtBat	int64, continuous
CHits	int64, continuous
CHmRun	int64, continuous
CRuns	int64, continuous
CRBI	int64, continuous
CWalks	int64, continuous
League	object, discrete
Division	object, discrete
PutOuts	int64, continuous
Assists	int64, continuous
Errors	int64, continuous
Salary	float64, continuous
NewLeague	object, discrete

IV. METHODS

A. Data Preparation

Before analyzing and testing our data, we prepared it. We did this by first exploring the data, checking for missing values, and splitting our data. When exploring the data, we looked at the summary of the data, the first few data entries, the length of the dataset, and the column names otherwise known as our variables. While exploring the data we found that the variable "Salary" had 59 missing values. Using the median method, we input these missing values as the median salary for players. Once all the missing values were fixed, we were able to split the data into training and testing sets which allowed us to test the data.

B. Experimental Design

It is important to note that the R Studio Validation set was created by creating a new file consisting of 5 entries from the original data.

Table X: Experiment Parameters

Experiment Number	Parameters
1-P	80/10/10 split Train, Test, and Validation
2-P	75/12.5/12.5 split Train, Test, and Validation
3-P	70/15/15 split Train, Test, and Validation
1-R	Training and Test sets were split 80/20 and used 5 values for validation
2-R	Training and Test sets were split 90/10 and used 5 values for validation
3-R	Training and Test sets were split 60/40 and used 5 values for validation

C. Tools Used

The following tools were used for the analysis:

- Python: Pandas, sklearn, numpy, seaborn
  - Pandas: Used to import the csv file and convert categorical variables into numeric data
  - Sklearn: Used to split the model into Train, Test, and Validation sets, calculate the linear regression model, and calculate the Mean Squared Error and R-Squared value
  - Numpy: Used to convert and stack the validation results as an array
  - Seaborn: Used to plot the Salary data as a histogram
- R Studio: Tidyverse, CaTools
  - Tidyverse Library: This library helped us plot our data as well as organize and view the data. This contains some of the basic functions for analyzing data in R Studio.

- CaTools Library: This library was used to split and test our data. It allowed us to do our linear regressions and predictions.

V. RESULTS

A. Mean square Error and R-Square calculation

Mean Squared Error measures the average squared distance between the predicted value and the actual value. It is calculated by subtracting the predicted value from the actual value and squaring the difference. This is repeated for each value. Finally, sum the squared values and divide them by the total number to get the average of the squared differences. The smaller this number is, the more accurate the model. R-Squared is used to explain the variance. It represents the amount of variance in the independent variable that can be explained by the dependent variable. The closer this value is to 1, or 100%, the better. Generally, 0.7 or higher is considered a good model.

B. Discussion of Results

When using python, all our models resulted in the same R-squared value of 0.47. When using R Studio, we got values of 0.4859 for our first model, 0.4638 for the second, and 0.4806 for the third. We believe this is due to how the data was split in each of our models. Our first and second models yielded similar MSEs, while the third model yielded a high MSE. This means that our first model would be the best predictor because it has the highest R-Squared value and close to the lowest MSE. While this was our best model, it is still not a good predictor for salary.

C. Problems Encountered

Team members were required to create their own validation set for R-Studio as none were provided with Kaggle.com. A provided validation set would have eased the analysis. In addition, 19 dependent variables are a lot to work with. At times, this made analysis slightly overwhelming. However, both problems were easy to work through during the analysis.

D. Limitations of Implementation

Our model is limited due to not having enough data. The data in our dataset does not include all players in Major League Baseball. It is also limited because 59 of the salary values were null. We had to replace these values with the median of all salaries given instead of their actual values.

E. Improvements/Future Work

Our model could have been improved if we had removed some of our variables. Focusing on several key variables instead of 19 would've helped narrow our predictions. It could have also been improved by using a larger dataset that contained current data for the MLB.

VI. CONCLUSION

Overall, our model was not very successful. With a low R-Squared value of 0.47 and a high Mean Squared Error, the model is not very accurate at predicting salary based upon seasonal and career batting statistics. The dataset required preparation and the large number of variables created a lot of work. For both Python and R Studio multiple experiments were conducted, however results were all similar, showing the poor prediction of Salary.

REFERENCES

Hitters Baseball Data | Kaggle