Project 3 158: Multiple Linear Regression

Tesfa Asmara and Kevin Loun

4/09/2022

Contents

1 Introduction

The dataset for this project contains 10,000 League of Legends ranked matches from the North American region with 775 variables offered through the Riot Games API, provided on Kaggle (Games 2021a)(James 2020). Each match is pulled from players who rank Gold in the League system, a ranking system that matches players of a similar skill level to play with and against each other. Amongst North American players, the Gold skill level was the second most common tier, achieved by 27.7 percent of players, or approximately 49.86 million players when considered against Riot Games' player base of 180 million ("Distribution of League of Legends (Lol) Summoners in North America as of October 2021, by Tier" 2021)(Games 2021b). This dataset will be referred to as `lol10`.

For this project, the following variables are of interest: time spent crowd controlling others, map side, longest time spent living, kills, gold earned, and total damage dealt. A figure including all the relevant variables and their description is attached at the end.

2 Hypothesis

We consider the following research question: Are one or more of the independent variables, `b_gold_earned`, `b_kills`, `b_longest_time_spent_living`, or `b_time_c_cing_others` in the model useful in predicting the future values of `b_total_damage_dealt`?

3 Feature Engineering

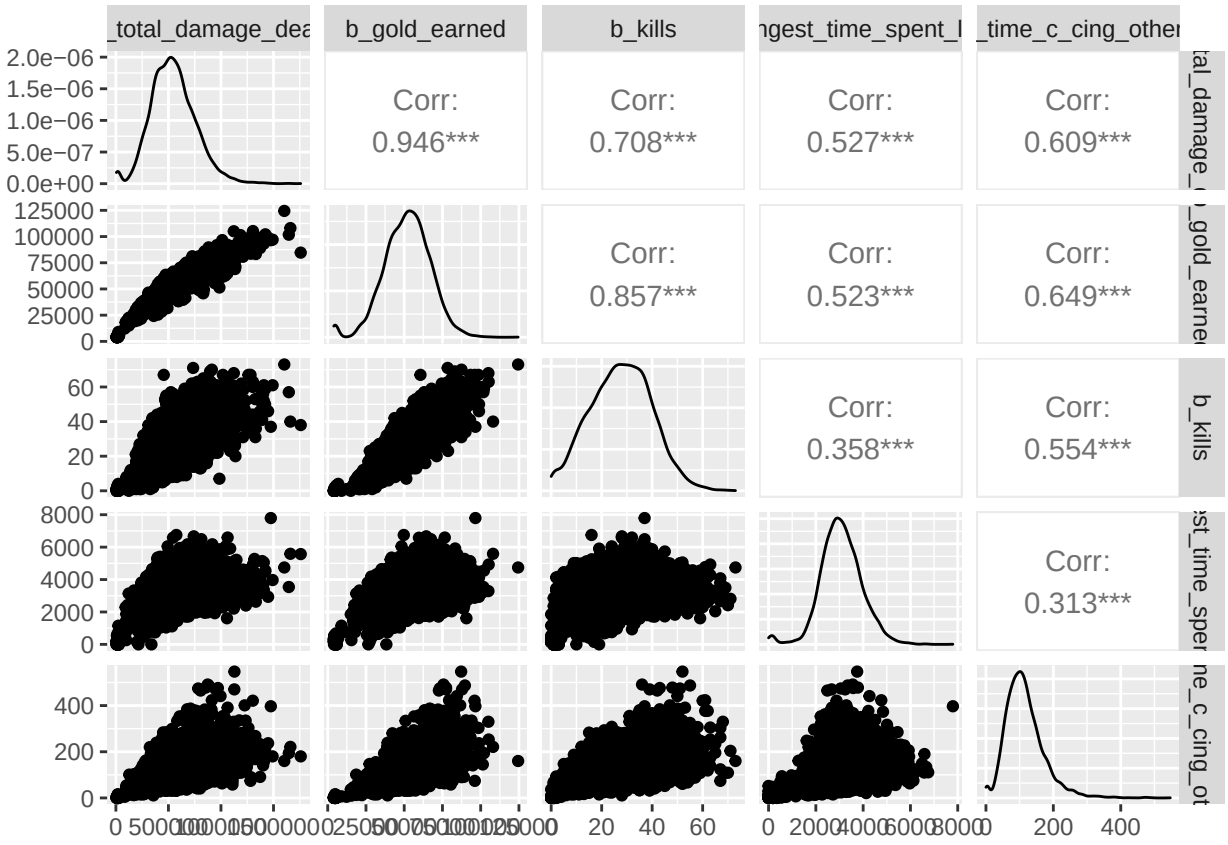


Figure 1: Correlation Matrix of Variables

For $n = 5$ observations, Table B.6 in ALSM is employed to assess whether or not the magnitude of the correlation coefficient supports the reasonableness of the normality assumption. The feature engineering we conducted was minimal.

4 Interaction Variables

We wish to test formally in the `lol10` dataset whether interaction terms between the four explanatory variables should be included in the regression model. We therefore need to consider the following regression model: ,

$$\begin{aligned}\text{b\_total\_damage\_dealt} = \beta_0 + \beta_1(\text{b\_gold\_earned}) + \\ \beta_2(\text{b\_kills}) + \beta_3(\text{b\_longest\_time\_spent\_living}) + \\ \beta_4(\text{b\_time\_c\_cing\_others}) + \beta_5(\text{b\_gold\_earned} \times \text{b\_kills}) + \\ \beta_6(\text{b\_gold\_earned} \times \text{b\_longest\_time\_spent\_living}) + \beta_7(\text{b\_gold\_earned} \times \text{b\_time\_c\_cing\_ot} \\ \beta_8(\text{b\_kills} \times \text{b\_longest\_time\_spent\_living}) + \beta_9(\text{b\_kills} \times \text{b\_time\_c\_cing\_others}) + \\ \beta_{10}(\text{b\_longest\_time\_spent\_living} \times \text{b\_time\_c\_cing\_others}) + \epsilon \end{aligned} \tag{1}$$

,

.

We wish to test whether any interaction terms are needed. We do so by performing a partial F-test by fitting both the reduced and full models separately and thereafter comparing them using the `anova()` function.

Since $F \approx 297.3748455$ (p-value ≈ 0), we reject the null hypothesis $H_0 : \beta_5 = \beta_6 = \beta_7 = \beta_8 = \beta_9 = \beta_{10} = 0$ at the $\alpha = 0.05$ level of significance. This means that the interaction terms do not contribute significant information to the `b_total_damage_dealt` once the explanatory variables `b_gold_earned, b_kills, b_longest_time_spent_living, b_time_c_cing_others` have been taken into consideration.

5 Computational Model

From our domain experience, we consider the following parsiminous models: ,

$$\begin{aligned}\text{b\_total\_damage\_dealt} = \beta_0 + \beta_1(\text{b\_gold\_earned}) + \\ \beta_2(\text{b\_kills}) + \epsilon \end{aligned} \tag{2}$$

,

and ,

$$\begin{aligned}\text{b\_total\_damage\_dealt} = \beta_0 + \beta_1(\text{b\_gold\_earned}) + \\ \beta_2(\text{b\_kills}) + \beta_3(\text{b\_longest\_time\_spent\_living}) +, \\ \epsilon \end{aligned} \tag{3}$$

,

.

We compare the two models using cross-validation prediction error.

In comparing which model is better, the CV RMSE provides information on how well the model did predicting each $1/v$, where $v =$ the number of folds, hold out sample. We can compare the model RMSE to the original variability seen in the `b_total_damage_dealt` variable. The original variability (measured by standard deviation) of `b_total_damage_dealt` was 2.1223789×10^5. After running Model 1, the remaining variability (measured by RMSE averaged over the folds) is 2661.4238251; after running Model 2, the remaining variability (measured by RMSE averaged over the folds) is 2677.1891693.

Hence, the better computational model is ,

$$\begin{aligned}\text{b\_total\_damage\_dealt} = \beta_0 + \beta_1(\text{b\_gold\_earned}) + \\ \beta_2(\text{b\_kills}) + \epsilon \end{aligned} \tag{4}$$

,

.

6 Statistical Model

For the four predictors in the `lol10` data, we know there are $2^4 = 16$ possible models. The four-parameter model ,

$$\text{b\_total\_damage\_dealt} = \beta_0 + \beta_1(\text{b\_gold\_earned}) + \beta_2(\text{b\_kills}) + \beta_3(\text{b\_longest\_time\_spent\_living}) + \beta_4(\text{b\_time\_c\_cing\_others}) + \epsilon \quad (5)$$

is identified as best by the $R_{a,p}$ criterion; it has $\texttt{max}(R_{a,p}) = 4$ and will serve as the selected model.

For the test data, the linear model we selected has $R^2 = 0.9321065$ and a $R^2_{adj} = 0.9319978$. Therefore, 0.0093211% of the variability in `b_total_damage_dealt` for players who rank Gold in the North American region is explained by the variables `b_gold_earned`, `b_kills`, `b_longest_time_spent_living`, `b_time_c_cing_others`. However, this R^2 is not a guarantee that our model will accurately describe the population. We have a relatively high R^2 value but this simply shows that the variables explain the variability but R^2 is easilly influenced by bias and can also be simply affected by the number of predictor variables in a model. So while R^2 is a good starting point it is not a guarantee that our model will be accurate to the population.

6.1 Beta interpretation and Mean Value Confidence Interval

When looking closer at the β coefficients of our model we can evaluate our coefficients based on their p-calue to determine if they are significant. Upon closer inspection it appears that `b_longest_time_spent_living` and `b_time_c_cing_others` are not significant in our final model. They have p-values of 0.208 and .608 respectively. We set our α to be 0.05 and as such these coefficients are not significant, however, `b_gold_earned` and `b_kills` are significant as they have p-values lower than our threshold.

```
## # A tibble: 5 x 5
##   term                          estimate std.error statistic   p.value
##   <chr>                            <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)                  -206876.    4744.      -43.6   2.17e-309
## 2 b_gold_earned                     18.2      0.171   107.    0
## 3 b_kills                        -6751.     178.      -37.9   2.20e-248
## 4 b_longest_time_spent_living       -1.89     1.50     -1.26  2.08e-  1
## 5 b_time_c_cing_others             -13.6     25.9      -0.525 6.00e-  1
```

The 95% Confidence interval for total damage dealt for summoners on the blue side is (546714,555966) meaning that we are 95% confident that the true mean value of total damage dealt is between(546714,555966). A future predicted value for a combination of X's can be seen using the following combination of X's:

```
##   b_kills b_gold_earned b_longest_time_spent_living b_time_c_cing_others
## 1      20         55000                          60                   15
```

And Once we apply the model we see that the new predicted value for future data containing similar values is:

```
##        1
## 660054.6
```

6.2 Model Interpretation

The final model we selected indicates that `b_total_damage_dealt` can be predicted by `b_gold_earned`, `b_kills`, `b_longest_time_spent_living`, `b_time_c_cing_others`. Initially we began with a model that included interactions between these variables but they were insignificant and were dropped from the model. These were most likely non-significant because changing one variable did not have a strong affect on any other variable, that is an increase or decrease in one variable did not result in an increase or decrease in any other variabilty. There seems to be a high correlation between `b_gold_earned`and `b_kills` which can indicate multicollinearity or that one of these variables can take the place of each other. This issue of multicollinearity will be addressed through a Variance Inflation Factor test further in this report.

7 Checking for Outliers

We first apply an outlier test on our data to see if there are any significant Bonferroni P values that could indicate potential outliers in our dataset.

```
##       rstudent unadjusted p-value Bonferroni p
## 1700 11.241725         1.2239e-28   3.0646e-25
## 2495  7.024868         2.7520e-12   6.8910e-09
## 1259  6.870322         8.0580e-12   2.0177e-08
## 1972  6.279619         3.9899e-10   9.9908e-07
## 413   5.727456         1.1417e-08   2.8588e-05
## 335   5.684848         1.4615e-08   3.6596e-05
## 1857  5.284712         1.3679e-07   3.4253e-04
## 1063  5.258432         1.5761e-07   3.9465e-04
## 1666  5.097518         3.6990e-07   9.2624e-04
## 1656  4.559145         5.3838e-06   1.3481e-02
```

The test output shows potential outliers at observations 1700,2495,1259,1972,413,335,1857,1063,1666,and 1665. While this test is important for identifying a potentially significant outlying observation, it is not the ultimate indicator of outliers and checking for patterns in our outliers so we can proceed to generate a plot of Dfbetas to see if these points reappear as outliers.

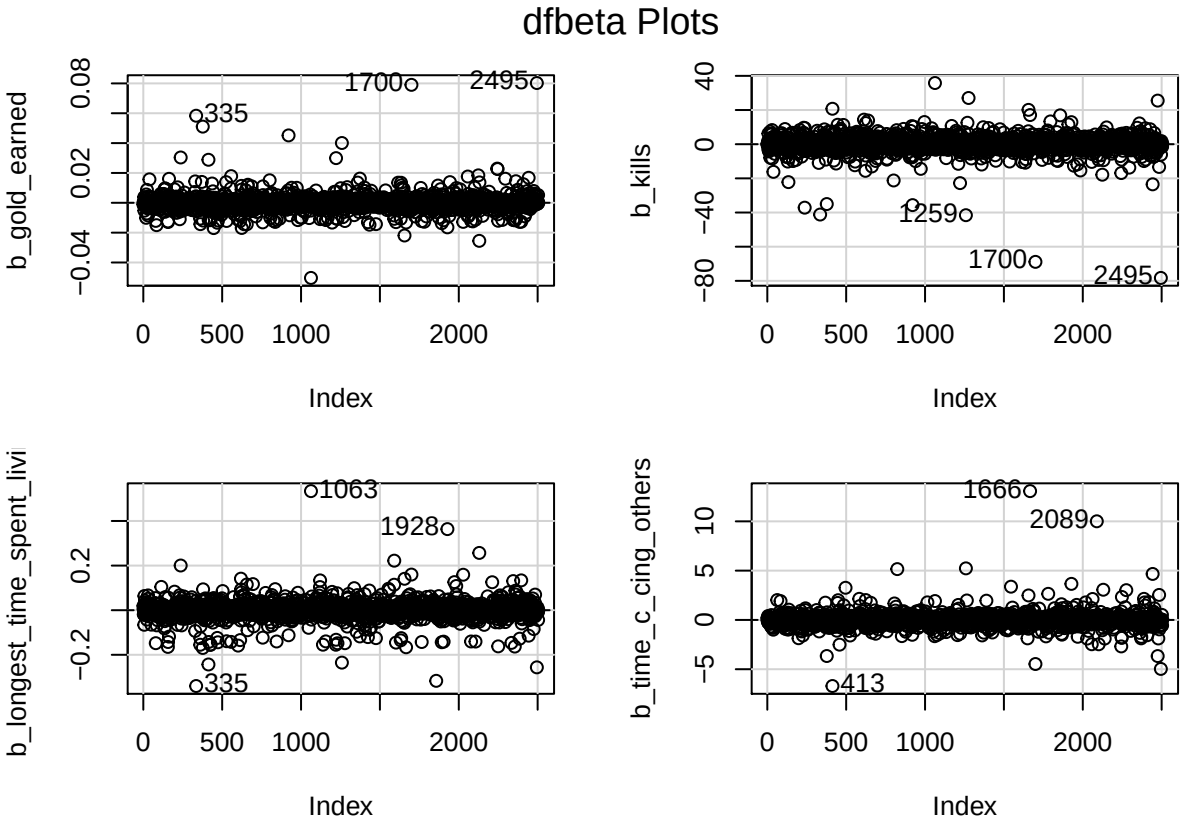


Figure 2: Plots for DfBetas

```
## NULL
```

We notice that the same outlying points appear on the dfbeta plots further indicating that these are outliers that could affect our model. In the leverage plot for each predictor we see a similar outcome; outliers at observation 1700 and 2495 for each predictor variable.

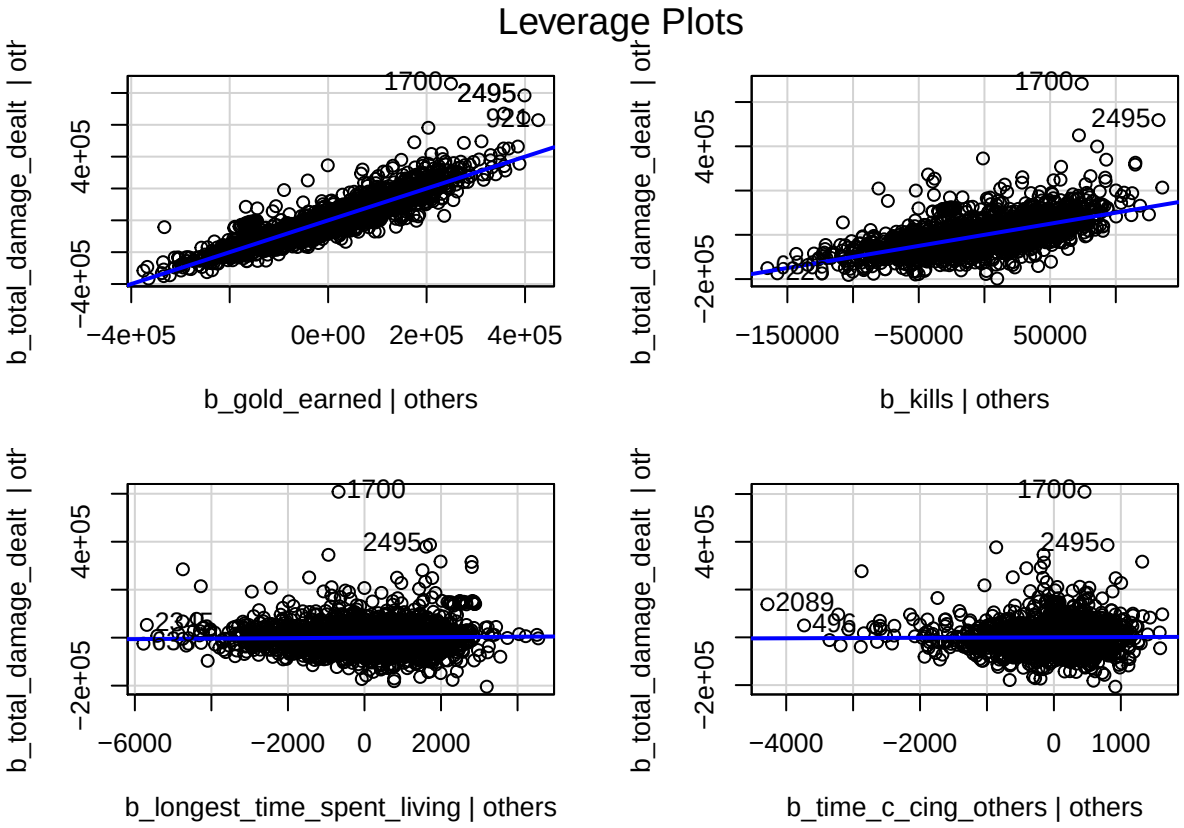


Figure 3: Leverage Plots for Outliers

To ensure that the outliers aren't an underlying cause of high correlation between our variables, we ploted our data in an added variable plot to visualize our outliers in plots where opur predictors are independent of one another.

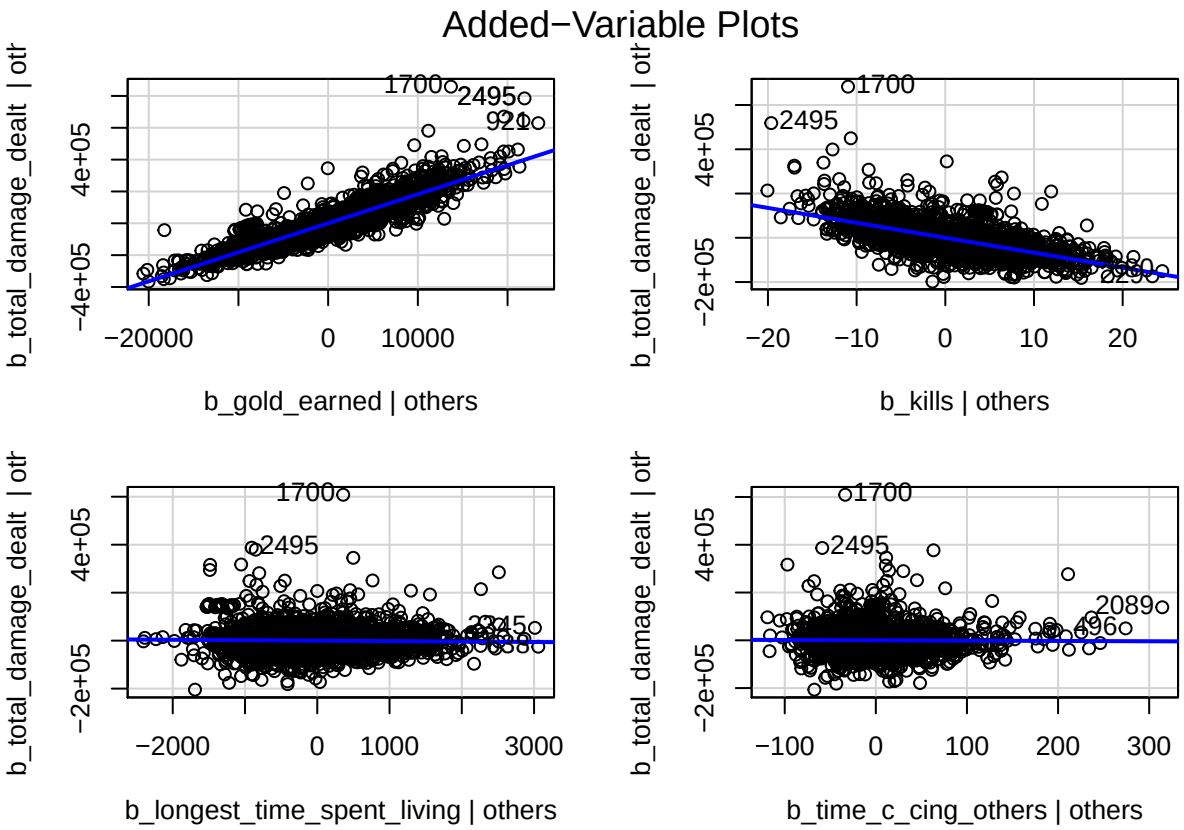


Figure 4: Added Variable Plot for each Predictor

We then chose to utilzes Cook's Distance as a metric to evaluate our outlier points. We set our cutoff value as $>\frac{4}{(n-k-1)}$, or 0.001601922. Any observation that yields a Cook's Distance value greater than our cutoff will be considered an outlier.

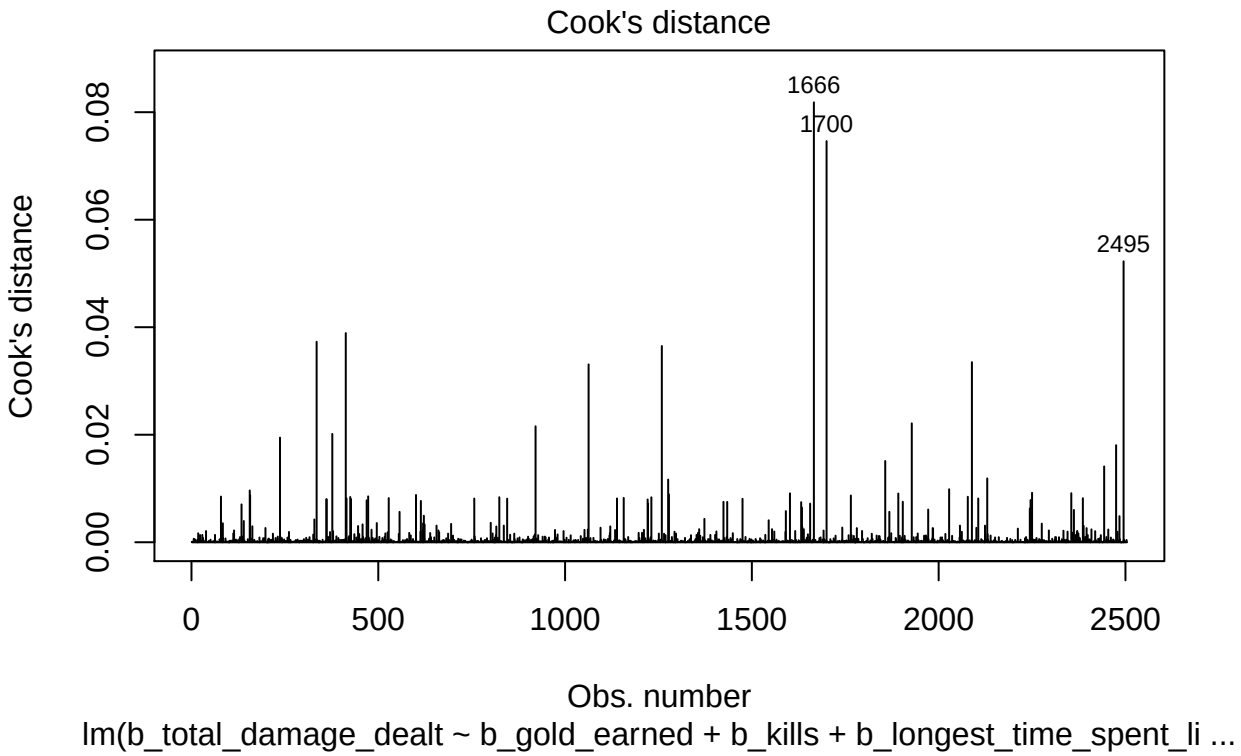


Figure 5: Cook's Distance per Observation

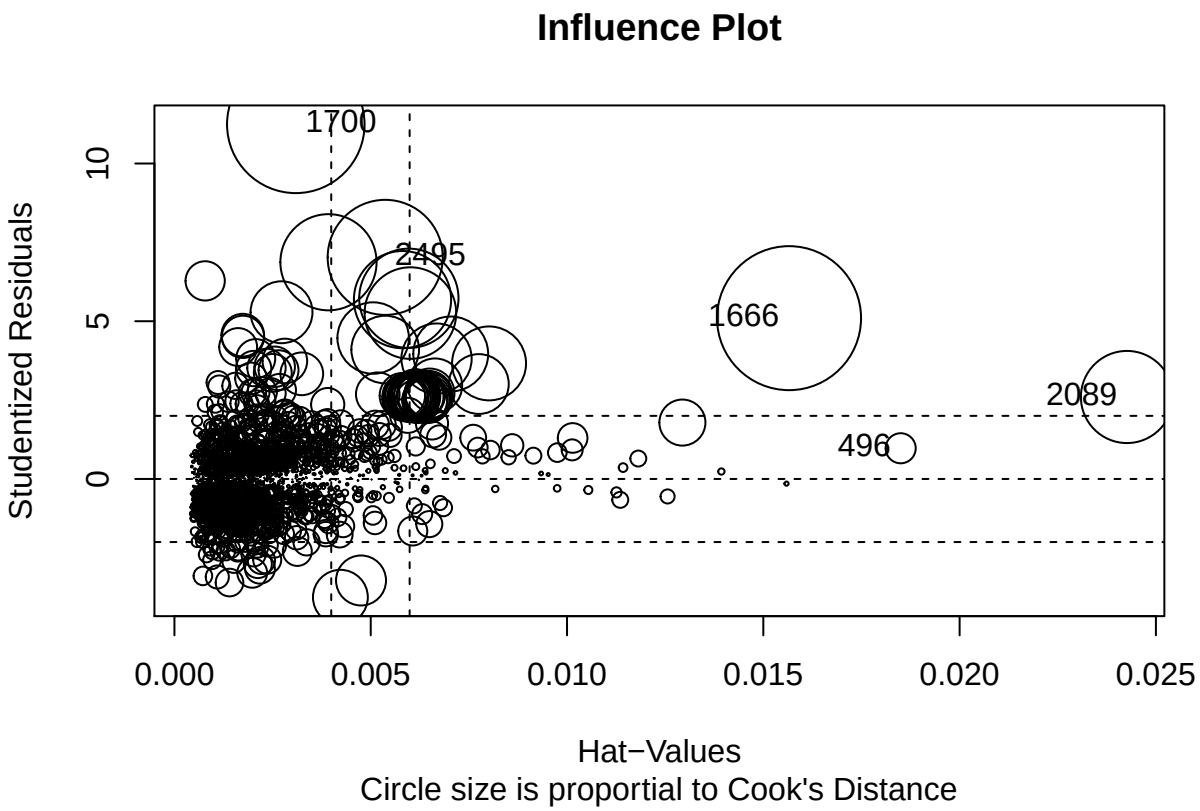


Figure 6: Influence plot for Cook's Distance

```
##         StudRes         Hat       CookD
## 496   0.9728662 0.018498547 0.003567732
## 1666  5.0975175 0.015654895 0.081833242
## 1700 11.2417253 0.003090876 0.074621180
## 2089  2.5985059 0.024262284 0.033502519
## 2495  7.0248676 0.005368642 0.052262053
```

Based on the influence plot and our cutoff value, we can identify observations 496, 1666,1700,2089,and 2495 as outliers, and after checking to ensure that these observations being outliers were not a result of data errors or misformulated regression, we have decided to remove these outliers from our data.

After removing the outliers from the data we can plot our studentized residuals in the form of a QQ plot and a histogram for datasets with and without the outliers. We noticed that without the outlier points our QQ plot appears more normal, though not perfect and that the shape of our histogram appears more normal than before droping the outliers.

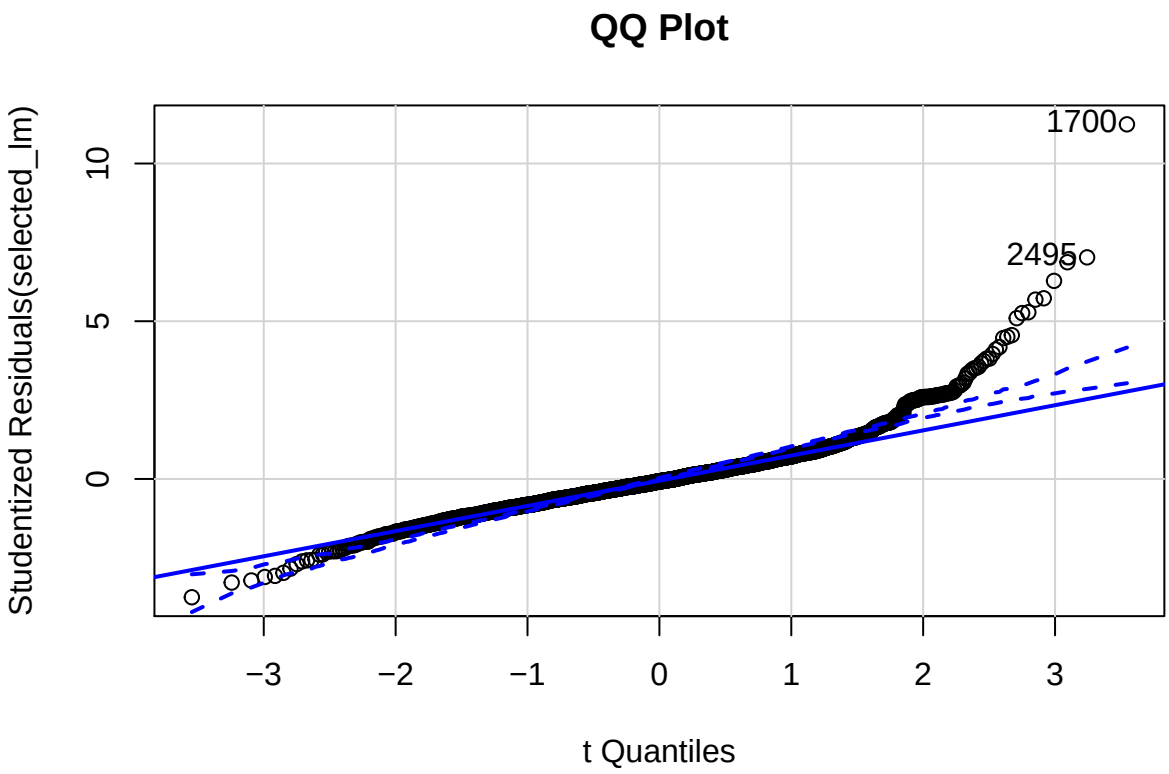


Figure 7: QQ Plot and Distribution of Studentized Residuals

```
## [1] 1700 2495
```

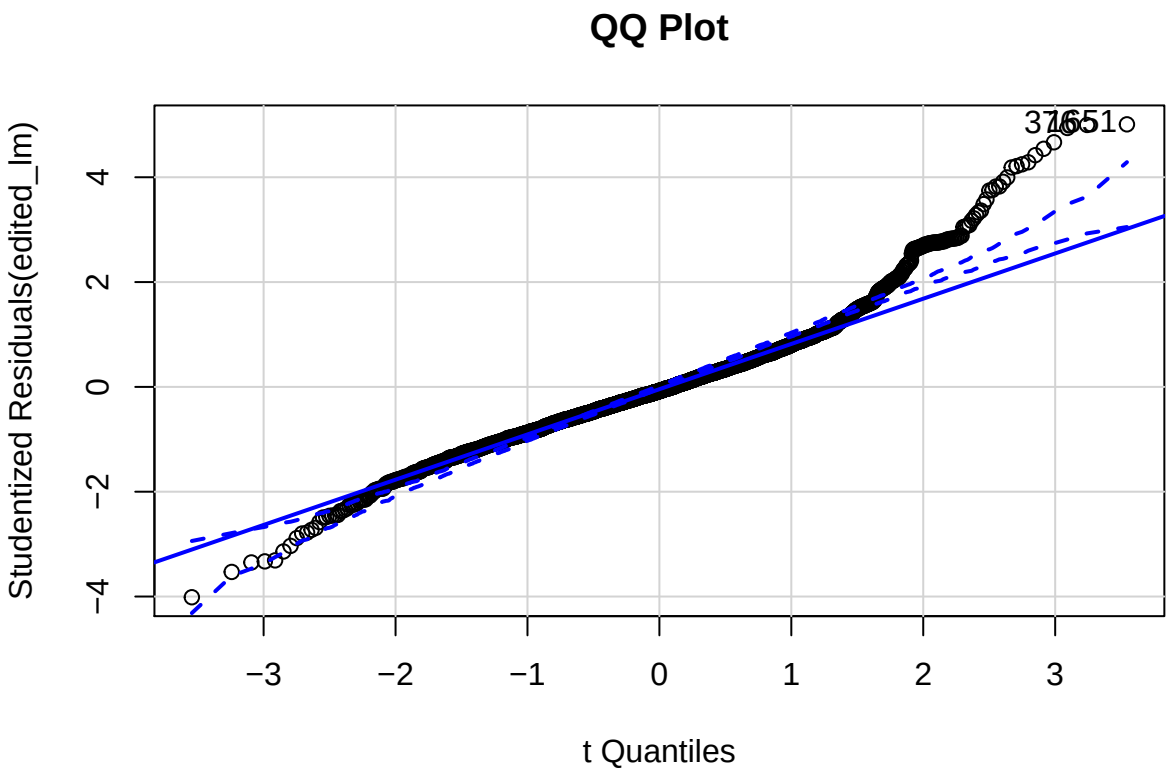


Figure 8: QQ Plot and Distribution of Studentized Residuals

```
## [1]  376 1651
```

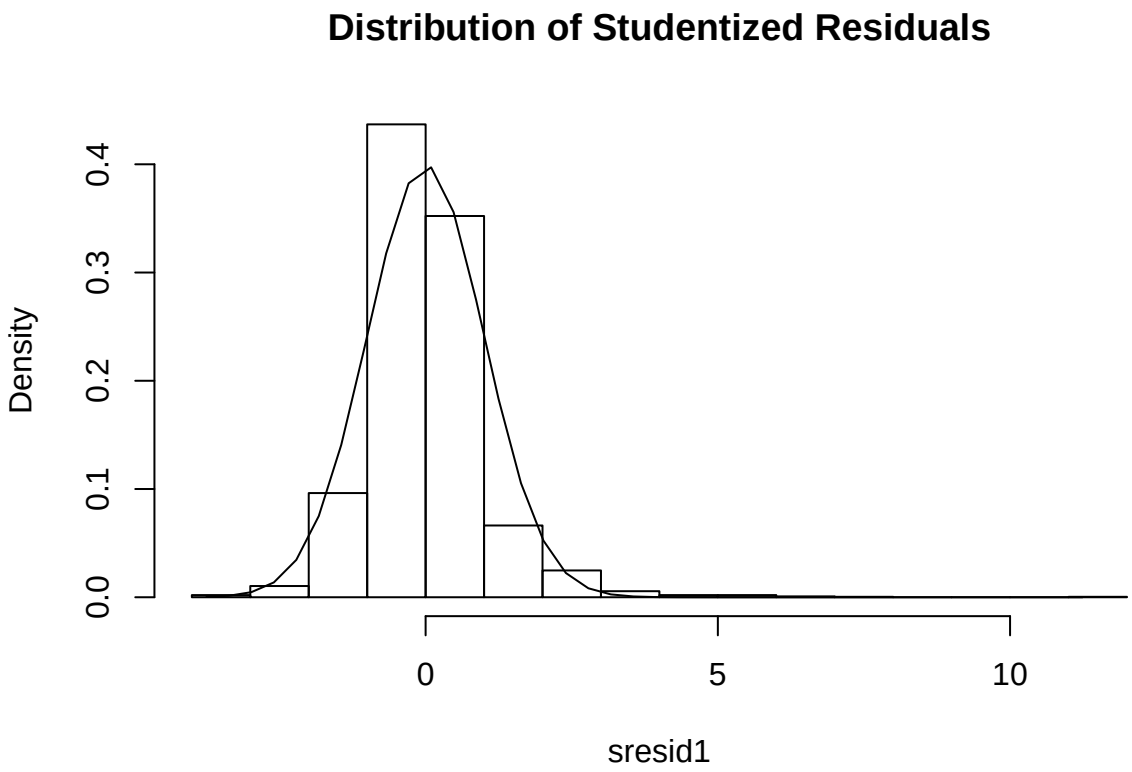


Figure 9: QQ Plot and Distribution of Studentized Residuals

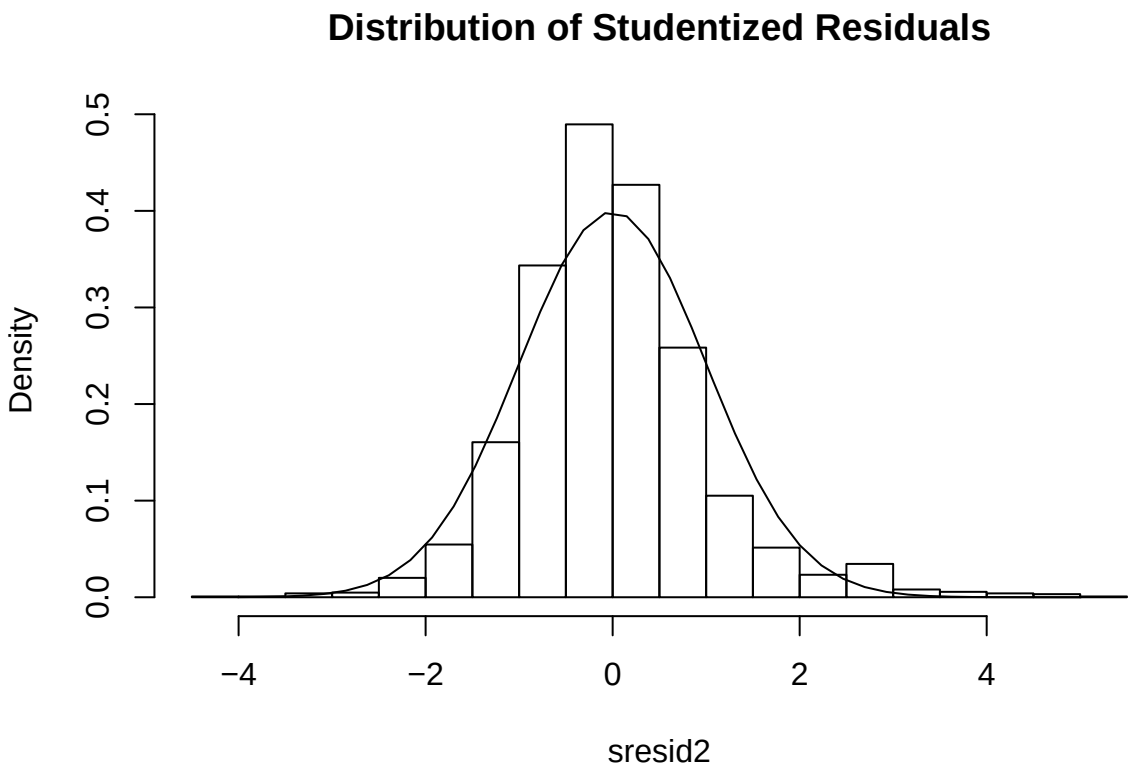
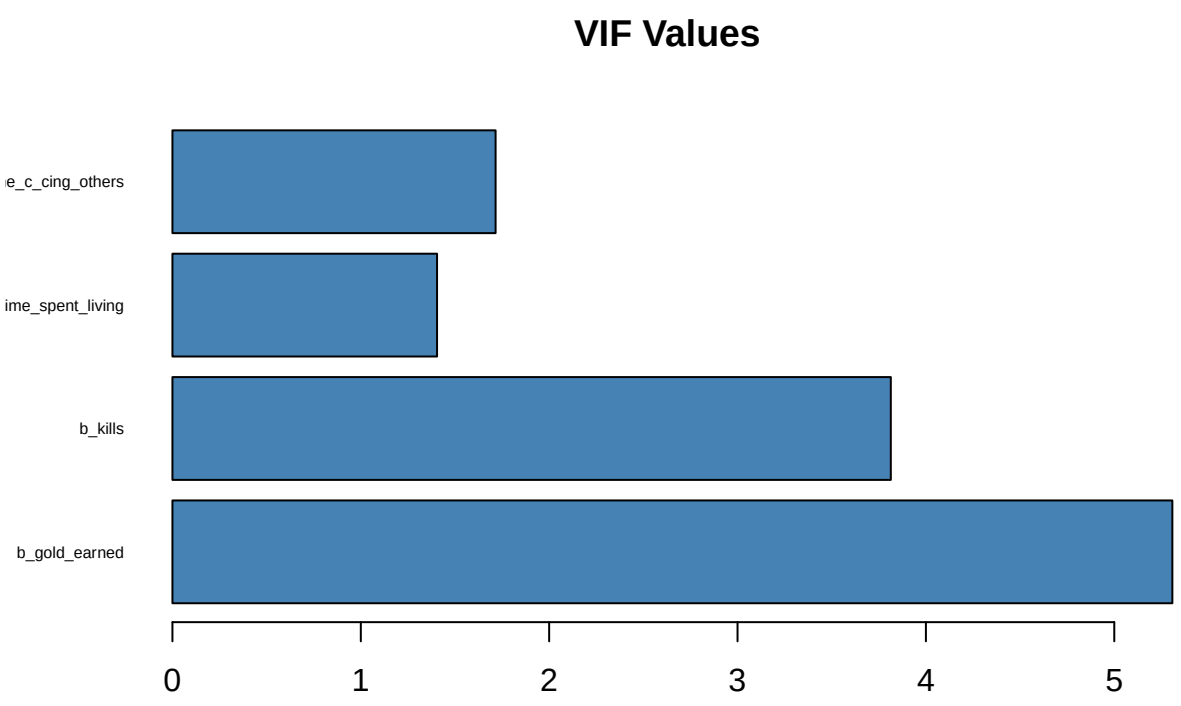


Figure 10: QQ Plot and Distribution of Studentized Residuals

7.1 Checking For Multicollinerity

In the correlation matrix generated earlier their appeared to be high correlation between some of our predictor variables; high correlation between predictor variables implies that there may be multicollinearity present in our model. To asses whether there really is multicollinearity we use the Variance Inflation factor for each variable.

```
##               b_gold_earned                     b_kills
##                    5.308366                    3.813736
## b_longest_time_spent_living       b_time_c_cing_others
##                    1.404775                    1.715489
```



We set our cutoff for VIF values at $VIF => 10$ and for our predictor variables the highest VIF was 5.308366 meaning that there is likely no multicollinearity occuring in our model.

8 Summary

We initially began by considering the following research question: Are one or more of the independent variables, `b_gold_earned`, `b_kills`, `b_longest_time_spent_living`, or `b_time_c_cing_others` in the model useful in predicting the future values of `b_total_damage_dealt`? We first began asnwering this question by assesing whether any feature engineering or interaction variables were neccessary in our model. We found that based on Table B.6 in ALSM that no feature engineering was needed. Through an F-test we found that non of the interaction variables were significant enough to add to our model. We then chose to approach the question through the use of a computational and statistical model. After comparing both models we concluded that the statistical model was the best choice for answering our question. We then proceeded to asses our data for any outliers that could affect our model; we accomplished this by using metrics such as: studentizeed residuals, cook's distance, dfbetas, and leverage. The outliers identified were not a result of data errors or misformulated regression so they were dropped from the data which in the end made the data more normal. Finally, a test for multicollinearity was applied due to a high correlation between variables. It was found that based on our cutoff value, no multicollinearity was present in our data.

Bibliography

"Distribution of League of Legends (Lol) Summoners in North America as of October 2021, by Tier." 2021. *Statista*. https://www.statista.com/statistics/807298/league-of-legends-player-tier/.

Games, Riot. 2021a. "Riot Games Api." *Riot Developer Portal*. https://developer.riotgames.com/apis.

———. 2021b. *Twitter*. Twitter. https://twitter.com/riotgames/status/1455172784938651649?s=20&t=AQmQGrTa1ijf6u3cEDPZcg.

James. 2020. "League of Legends Ranked Match Data from Na." *Kaggle*. https://www.kaggle.com/jamesbting/league-of-legends-ranked-match-data-from-na.

Variable	Description
b _kills	The number of kills obtained by summoners on the blue side of the map.
b _gold_earned	The gold obtained by summoner 1 on the blue side of the map.
b _longest_time_spent_living	Sum of the longest time spent alive by summoners on the blue side of the map.
b_time_c_cing_others	The total time spend crowd controlling enemy players by summoners of the blue side of the map.
b_total_damage_dealt	Total damage done by summoners on the blue side of the map.

Figure 11: Variables and their descriptions