Title: League of Legends 2021 World Championship Play-In Groups Statistics

Introduction:

Last October, 220 professional gamers from across the world participated in one of the most exciting esports event in the world: the 2021 League of Legends World Championship.

During such events, data analysis/machine learning research are conducted to predict the outcome of each game. While different papers have tried to predict the outcome (See Bibliography), we decided to focus more towards in-game statistics. Precisely, the goal of this research is to understand the extent to which the predictors affect the gold earned by creating the best model. The result of this research is important because it could be beneficial to players of all levels, as it helps them prioritize some aspects of the game over others to maximize gold earned.

Methods:

i) Variable selection

First, we removed the following predictors: Player, Opponent, Champion, Ward Interactions, Dragons For, Dragons Against, Barons For, Barons Against, and Kill Participation. The main reason is that these predictors are simply unrelated to Gold Earned, just from my knowledge of the game. For example, the predictor Opponent is just the name of the team that the player is facing, which has nothing to do with Gold Earned. A similar argument can be made for the other removed predictors.

Now, we focus our attention to the following kept predictors: Position, Kills, Deaths, Assists, Creep Score, Champion Damage Share, Wards Placed, Wards Destroyed, Result.

When it came down choosing our variables, we've used extensively the result of the F-test and the T-test to guide us into selecting which predictors were linearly related to the response in the presence of other predictors. We primarily relied on the summaries of the models, which immediately provided us with the respective p-values for each test. Moreover, we've also decided to perform a partial F-test as we noticed multiple predictors weren't linearly related to the response, so we wanted to test a subset of the original model. Through the summary of the models, we were also looking for a high adjusted R-score, relatively low Residual Standard Error, and relatively low Standard Errors for individual coefficients.

ii) Model validation

For model validation, we first divided our 220 observations into 2 sets: a training set and a test set using a 50/50 split. Then, to get a general sense of whether the two sets were similar we created a table of the means/standard deviations of each set.

At the end, after we've selected two models using the data in the training set, we've constructed the same models using the test set. For each model, we've rechecked the summary and plotted the residuals plots to check for assumptions violations and calculated the VIF of each predictor.

The final model was chosen to be the one that performed similarly well in the training set and in the test set, meaning that it had relatively low Residual Standard Error, high adjusted R-squared, linearly significant according to the T-test and the F-test, little to no assumption violations, no problematic points (leverage, influential, outlier), and also relatively lower VIFs.

iii) Model violations and Diagnostics

First, we plotted histograms of each variable to get a general sense of the means, spread, and shape. This step was also useful in checking the possibility of the normality violation for the response.

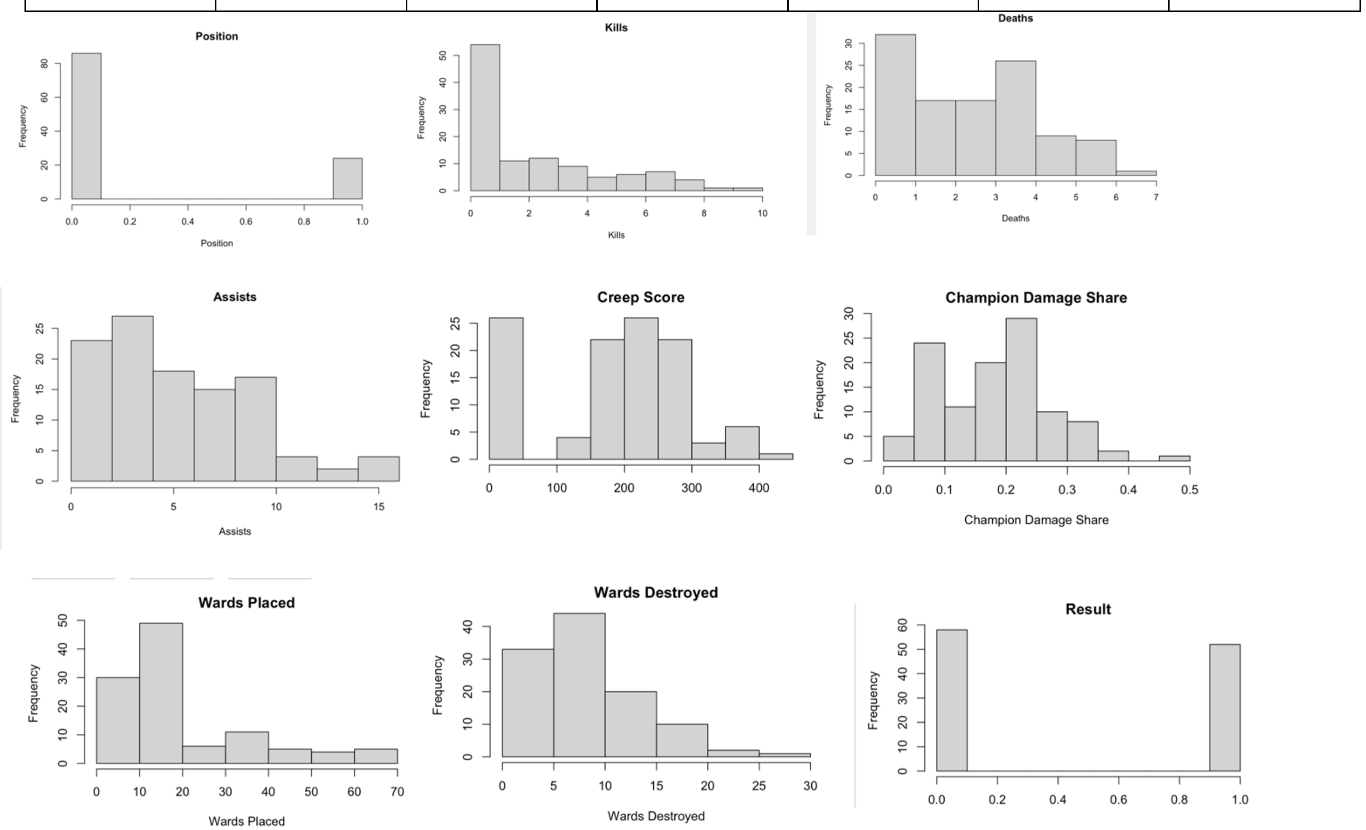
Next, we had to check conditions 1 and 2 before checking violations assumptions. We pair-wise plotted the predictors against each other to check whether a linear relationship exists between pairs of predictors for condition 2, and we plotted Gold Earned (our response) vs fitted values to check whether the conditional mean is a single function of the model. When the conditions look like they are satisfied, we then proceeded to plot the residuals vs fitted values, residuals vs predictors, and QQ plots to assess assumptions violations. We were on the lookout for any sort of clustering and fanning pattern in the residual plots, as those might indicate a violation of the constant variance assumption. On the other hand, the QQ plots were useful to assess the violation of the normality assumption. If we noticed any of those behaviors, we would consider using power transformations on either the response or the predictors to correct them.

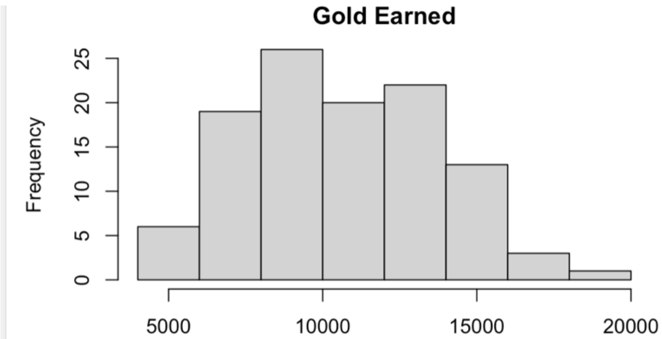
Results:

i) Description of Data

Table 1: the numerical/visual summaries of each variable

	Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
Position	0.000	0.000	0.000	0.2182	0.000	1.000
Kills	0.0	1.0	2.0	2.6	4.0	10.0
Deaths	0..000	1.000	3.000	2.827	4.000	7.000
Assists	0.000	3.000	5.000	5.645	8.000	16.000
Creep Score	14.0	136.8	295.5	189.4	258.0	414.0
Champion Damage Share	0.0400	0.1000	0.1900	0.1837	0.2400	0.4600
Wards Placed	5.00	10.00	13.00	19.97	21.75	67.00
Wards Destroyed	1.000	4.250	7.500	8.845	12.000	30.000
Result	0.0000	0.0000	0.0000	0.4727	1.0000	1.0000
Gold Earned	4714	8286	10139	10640	13130	19588





Notice that one of the key insights from this summary is that the distributions of some predictors seemed to be skewed. In particular, the predictors Position, Kills, Deaths, Assists, Wards Placed, and Wards Destroyed all exhibit some right skewness. Furthermore, the distribution of Gold Earned seems to be normal, so we should expect that later on in the assumption violation verification step, the QQ line in the QQ plot traverses through most of the data points.

ii) Process of Obtaining the Final Model

After dividing our full dataset into the training set and the test set, we compared the means/standard deviations of each set to make sure they were similar (See Appendix A).

Now, we fit the full model with the training data. The summary is displayed below:

Figure 1: the summary of the full model. Estimates of the coefficients for each predictor is displayed, as well as the p-values of different hypothesis tests are shown.

lm(formula = Gold. Earned~., data = train)					
	Estimate	Std. Error	T value	Pr(>|t|)	Significance
(Intercept)	1743.016	424.122	4.110	8.12*10^-5	(***)
Position	-187.909	212.785	-0.883	0.3793	
Kills	326.544	41.643	7.841	4.98x10^-12	(***)
Deaths	101.775	54.494	1.868	0.06741	(.)

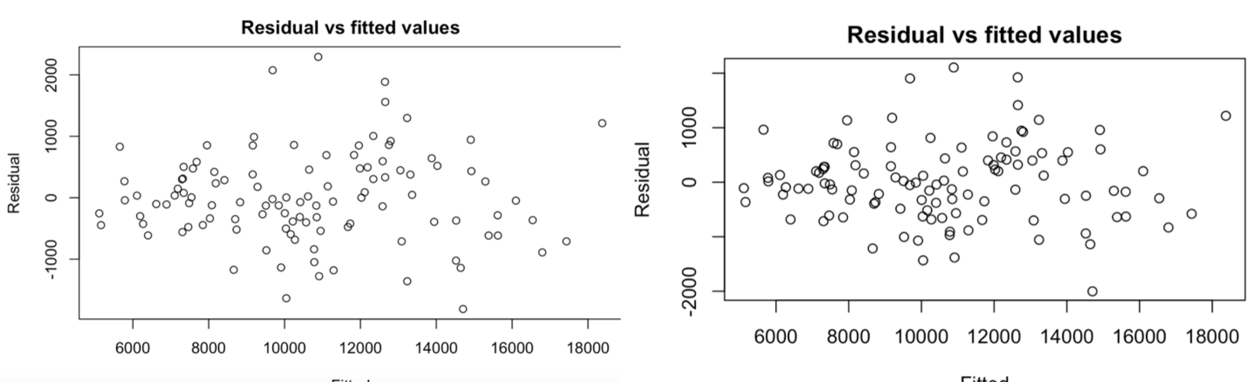
Assists	114.354	26.450	4.323	3.64x10^-5	(***)
Creep Score	27.866	1.327	20.999	<2x10^-16	(***)
Champion Damage Share	1664.936	1346.568	1.236	0.219195	
Wards Placed	42.366	7.407	5.720	1.11x10^-7	(***)
Wards Destroyed	55.252	15.384	3.592	0.000512	(***)
Result	503.632	245.909	2.048	0.043176	(*)
Residual standard error: 738.6 on 100 df Adjusted R-squared: 0.9421 F-statistic: 198.2 on 9 and 100 DF, p-value <2.2x10^-16					

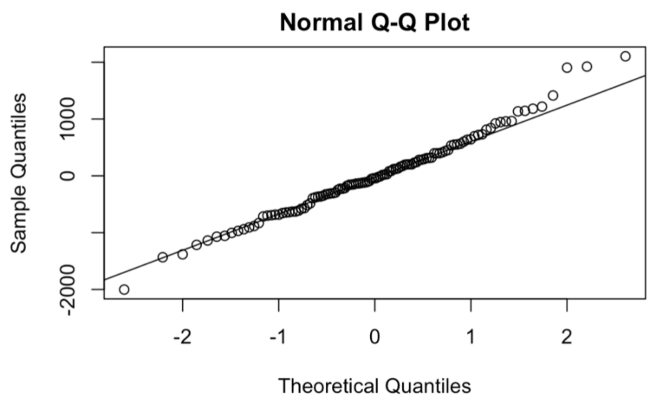
First off, the F-test indicate that there's an overall linear relationship between the predictors and the response. However, individual T-tests indicate that at the 0.05 significance level, the predictors Position, and Champion Damage Share are not linearly related to the response in the presence of other predictors. Finally, we obtained a strong adjusted R-squared of 0.9421, but a big Residual standard error, which suggests that the line is not fitting the data points well.

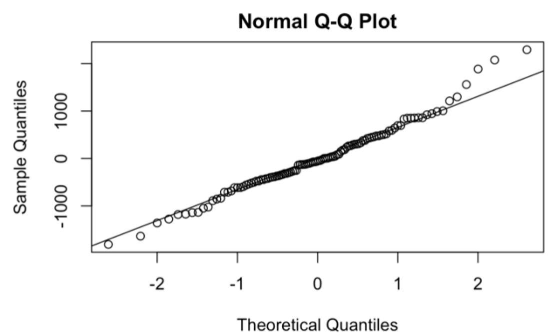
We then proceeded to create two different models. The first one is obtained by removing Position and Champion Damage Share, while the other one is obtained by further removing Deaths. This decision is supported by the Analysis of Variance Table in Appendix B.

After verifying conditions 1 and 2, we decided to move on to verifying assumptions. We plotted the corresponding plots.

Figure 2: residuals vs fitted values and QQ-plots for model 1 (on the left) and model 2 (on the right). Data points in both figures seem to be randomly spread, and no clustering is observed. QQ-lines seem to fit most data points for both models







However, we still decided to proceed with a squared root transformation on the response for both models because the Profile Likelihood gave us a 95% confidence interval centered at around 0.7.

Figure 3: summaries of the two transformed models via a square root transformation

lm(formula = I(sqrt(Gold. Earned)) ~ Kills + Deaths +Assists + Creep Score +Wards Placed+Wards Destroyed + Result, data = train)					
	Estimate	Std. Error	T value	Pr(>|t|)	Significance
(Intercept)	60.501966	1.898439	31.869	< 2x10^-16	(***)
Kills	1.609049	0.177069	9.087	8.51x10^-15	(***)
Deaths	0.594088	0.271830	2.186	0.031140	(*)
Assists	0.609159	0.131298	4.640	1.04x10^-5	(***)
Creep Score	0.135226	0.005283	25.597	<2x10^-16	(***)
Wards Placed	0.155750	0.036185	4.304	3.85x10^-5	(***)
Wards Destroyed	0.264837	0.069906	3.788	0.000257	(***)
Result	2.535753	1.200455	2.112	0.037100	(*)
Residual standard error: 3.708 on 102 df Adjusted R-squared: 0.939 F-statistic: 240.5 on 7 and 102 df, p-value < 2.2x10^-16					

lm(formula = I(sqrt(Gold. Earned)) ~ Kills + Assists + Creep Score + Wards Placed +Wards Destroyed, data = train)					
	Estimate	Std.Error	T value	Pr(>|t|)	Significance
(Intercept)	62.69404	1.58214	39.689	<2x10^-16	(***)
Kills	1.69952	0.17091	9.944	<2x10^-16	(***)
Assists	0.72621	0.11105	6.539	2.36x10^-9	(***)
Creep Score	0.13407	0.00535	25.061	<2x10^-16	(***)
Wards Placed	0.15373	0.03676	4.183	6.03x10^-5	(***)
Wards Destroyed	0.25923	0.07076	3.663	0.000393	(***)
Residual standard error: 3.774 on 104 df Adjusted R-squared: 0.9368 F-statistic: 324 on 5 and 104 df, p-value: <2.2x10^-16					

The results of the T-tests and F-tests are preserved. In addition, we notice something phenomenal: the Residual standard error dropped from 748.7 (full model) to around 3.7 in both transformed models! These two models seem much superior to the full model. Now, let's determine a final model based on multicollinearity.

Model 2's predictors seem to have lower VIFs compared to those in Model 1 (See Appendix C), which suggests that predictors in Model 2 are less linearly dependent with each other as opposed to those in Model 1, so we'll go with Model 2.

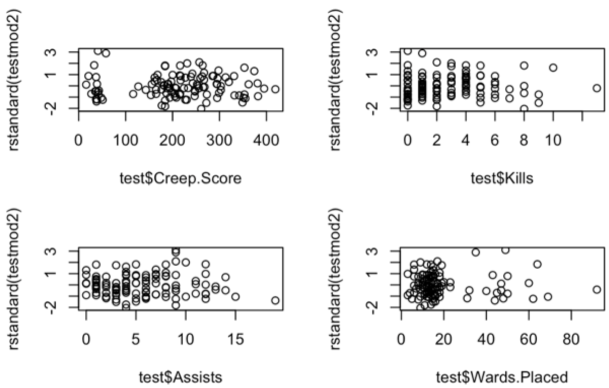
ii) Goodness of Final Model

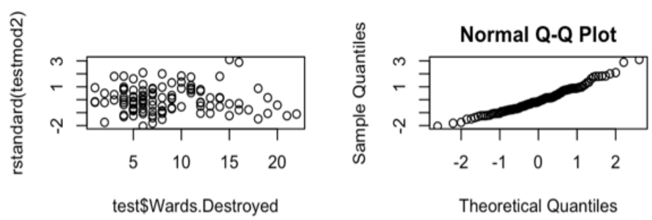
Now, for the validation step, we refit the transformed Model 2 but using the test set instead.

Figure 3: summary of the validated model +residual/QQ plots to check for assumptions violations

lm(formula = I(sqrt(Gold. Earned)) ~ Kills + Assists + Creep Score + Wards Placed +Wards Destroyed, data = test)					
	Estimate	Std.Error	T value	Pr(>|t|)	Significance
(Intercept)	62.138203	1.5331189	40.529	<2x10^-16	(***)
Kills	2.172589	0.156977	13.849	<2x10^-16	(***)
Assists	0.736824	0.095736	7.696	8.42x10^-12	(***)
Creep Score	0.134475	0.005062	26.564	<2x10^-16	(***)

Wards Placed	0.182942	0.033140	5.520	2.5x10^-7	(***)
Wards Destroyed	0.141066	0.086727	1.627	0.107	
Residual standard error: 3.667 on 104 df Adjusted R-squared: 0.9437 F-statistic: 366.3 on 5 and 104 df, p-value: <2.2x10^-16					





In the summary, we see that Wards Destroyed is no longer linearly significant to the response. Furthermore, we see that in one of our residual plots, namely, residual vs Wards Placed, there seems to be clustering at around 0-20, which wasn't really present in the training set. As for Residual standard error, the adjusted R-squared, and the other residual/QQ-plots, they seem to be fine. Finally, the estimates of the coefficients seem relatively similar to those in the training set.

Discussion:

i) Final Model Interpretation and Importance

Our final model is the transformed Model 2 using the test set, in particular, we have,

$$\sqrt{Gold\ Earned} = 62.14 + 2.17Kills + 0.74Assists + 0.13Creep\ Score + 0.18Wards\ Placed + 0.14Wards\ Destroyed$$

The interpretation of this is that if we keep the other predictors fixed, on average, as the number of Kills/Assists/Creep Score/Wards Placed/Wards Destroyed by 1, the square root of Gold Earned increases by the corresponding coefficients. Now, coming back to the goal of the study, we can confidently deduce that the number of kills has the most positive impact on the square root of the amount of gold earned one receives in the game. Therefore, players of all level should focus more on the number of kills to maximize their gold earned in the game.

ii) Limitations of Analysis

One of the lingering problems with the final model is that one of the residuals plots had significant clustering. This would heavily impact the values of the VIF. The reason why we weren't able to correct this assumption violation could be because of influential points. In the training set, there were a couple of data points that were flagged as influential points but they were relatively close to the rest of the data points (which is why I didn't discuss them in this paper). However, the issue might be that in the test set, there were more problematic outlier/influential/leverage points, which ultimately lead to this assumption violation.

Appendix

Appendix A: mean and standard deviations of the training set versus the test set

Variable	Mean (s.d.) in training	Mean(s.d.) in test
Position	0.218 (0.415)	0.182 (0.387)
Kills	2.6 (2.542)	2.818 (2.624)
Deaths	2.827 (1.771)	2.609 (1.725)
Assists	5.645 (3.883)	5.691 (3.96)
Creep Score	189.373 (103.116)	211.309 (98.759)
Champion Damage Share	1.06x10^4 (3070.567)	1.31x10^4 (3295.027)
Wards Placed	0.184 (0.089)	0.216 (0.098)
Wards Destroyed	19.973 (15.848)	18.936 (15.633)
Result	8.845 (5.456)	8.564 (4.742)
Gold Earned	0.473 (0.502)	0.527 (0.502)

Appendix B: the Analysis of Variance Table for both reduced models. Both reduced models appeared to be linearly significant using the partial F-test

```
Analysis of Variance Table

Model 1: Gold.Earned ~ Kills + Deaths + Assists + Creep.Score +
Wards.Placed +
    Wards.Destroyed + Result
Model 2: Gold.Earned ~ Position + Kills + Deaths + Assists + Creep.Score +
    Champion.Damage.Share + Wards.Placed + Wards.Destroyed +
    Result
  Res.Df      RSS Df Sum of Sq      F Pr(>F)
1    102 55819955
2    100 54556664  2   1263291 1.1578 0.3184
```

```
Analysis of Variance Table

Model 1: Gold.Earned ~ Kills + +Assists + Creep.Score + Wards.Placed +
    Wards.Destroyed
Model 2: Gold.Earned ~ Position + Kills + Deaths + Assists + Creep.Score +
    Champion.Damage.Share + Wards.Placed + Wards.Destroyed +
    Result
  Res.Df      RSS Df Sum of Sq      F Pr(>F)
1    104 58052189
2    100 54556664  4   3495524 1.6018 0.1798
```

Appendix C: the VIF of each model, rounded to 3 decimal places

	Kills	Deaths	Assists	Creep Score	Wards Placed	Wards Destroyed	Result
Model 1	1.606	1.836	2.007	2.352	2.607	1.1529	2.873
Model 2	1.445	N/A	1.386	2.329	2.597	1.1405	N/A

Bibliography

Gaina, R., & Nordmoen, C. (2018, June 9). *League legends study - github pages*. League of Legends: A Study of Early Game Impact. Retrieved October 22, 2021, from https://rdgain.github.io/assets/pdf/league-legends-study.pdf.

Lin, L. (2016). *League of legends match outcome ... - stanford university*. League of Legends Match Outcome Prediction. Retrieved October 22, 2021, from http://cs229.stanford.edu/proj2016/report/Lin-LeagueOfLegendsMatchOutcomePrediction-report.pdf.

Hsu, S. (2018). *Classifying and analyzing playstyle in League of Legends*. Student Association for Applied Statistics. Retrieved October 22, 2021, from https://saas.berkeley.edu/rp/league-of-legends.