Predicting Baseball

Short Summary

Ben Rogers, Jason Lwin, Kevin Finity, Spencer Marusco

The premise of baseball is simple: the team that scores the most runs in a game wins. Repeat this 162 times, for one team's regular season. Using data found on Baseball Reference dating back from 1876 to today, our goal is to create a model to predict the number of games a team will win with a linear model, and their likelihood of winning a single game with a logistic model.

For our project, we looked at over 100 possible variables, including data on what league the team is in, batting statistics, and pitching statistics. The first problem we came across was there was a high correlation between many of the variables. For example, if a team scores more runs, their hitters will have more plate appearances. Our goal was to minimize the correlation between variables and to limit the number of variables within the regression using step functions to help determine fit. Looking at the correlation between variables, we used the VIF function from the "car" library within R to observer correlation. Any VIF higher than 10 indicated a high enough correlation to cause concern. We used the "car" library as the VIF function from the "faraway" library would return significantly lower VIF values for logistic regressions, with a max of around 2. This was a problem - using the same variables in the linear function, we discovered that the variables could have VIF values well exceeding our cutoff of 10. Since the VIF function is looking at the correlation between variables, we knew that a change in the type of regression should not decrease the VIF values. In reality, the values should be the same, as long as the regressors are the same. Reaching out to a coworker of Spencer's, he discovered a bug in the VIF function for logistic regression, so we used an alternative VIF function. Another interesting finding is that when using the forward and backward stepwise functions in R, linear regressions are more likely to keep a variable compared to logistic regressions, assuming the same approach for deriving the models were used. We could not figure out why, but it did seem notable.

Our final logistic regression model is win_odds = exp(0.12 + 0.0025*earned_run_avg_plus_pit + 0.00243*onbase_plus_slugging_plus_off - 0.3678*R_pit + 0.3749*runs_per_game_off - 0.0185*AB_off.

The linear model we selected is wins = 65.65 + 0.31*earned_run_avg_plus_pit + 0.30*onbase_plus_slugging_plus_off + 12.03*RBI_off - 0.82*AB_off - 2.22*X2B_off + 1.39*SH_off + 0.82*leagueNL + 4.50*LOB_pit - 3.66*fip_pit - 0.10*batters_used_off - 5.92*BB_pit - 7.34*H_pit + 2.18*SB_off - 3.26*WP_pit - 1.20*strikeouts_per_base_on_balls_pit + 0.31*age_bat_off. The linear model gives an Adjusted R-squared value of 0.87.

We chose to use a larger model because while it is harder to pinpoint a specific variable, the reduction of the number of model parameters would be less helpful. At its core, since baseball games are decided by who scores more runs, a two variable model including runs allowed and runs scored would likely be predictive; however, it would not offer any insight into how to improve a team's chances of winning a game, or maximizing wins in a season. Teams cannot magically score more and stop more runs at the drop of a hat. With this large model, teams have several variables they can focus on improving as a team. Since our logistic model is meant to show win percentage for one game, a smaller set of variables provides a better indicator for the success of the team as we can pick broader variables that are more useful in a general sense.

In conclusion, our two models will enable teams to predict their success over the course of the season. The linear model provides Baseball executives knowledge of where their team stands going forward into the season and allows for long term projections. The logistic model provides teams a model to use in the playoffs and near the trade deadline to determine if they should be buyers or sellers or their effective odds of making the playoffs and winning the World Series.

Full Report

Our search for a proper linear and logistic regression model began with deciding what variable we were trying to predict. Armed with decades of baseball data, including records by team and by season, the logical choice was to try to predict a variable related to a team's record. From our vantage point, we had two options: we could try to predict the expected number of wins for a team in a given season, or we could try to predict the win probability of a team, in a specific game. At the outset, we favored the logistic regression model approach because we could view our calculated probabilities as the likelihood of

winning a single game and then scale up to multiple games or even an entire season. This added flexibility, relative to predicting the number of wins over the course of a season, is an obvious benefit.

With two objectives in hand, the next step was data cleaning. Our dataset was scraped from the popular website www.baseball-reference.com. Though the data is extremely thorough, there were still a variety of issues present. We began by dropping duplicate columns, like "Hits - defense" is identical to "Hits - pitching". Next, we settled on replacing missing values by imputing the mean for the column. To the extent that the values for these columns are non-stationary, this method runs the risk of creating potential issues with our dataset. However, we regressed each of the columns containing missing data (with the missing values removed) against both number of wins and win percentage. We found the fields with missing values did not appear to be strong predictors of number of wins or win percentage. This gave us comfort that our crude approach would suffice. Next, we turned to the problem of unscaled variables. Since seasons have varied by length over time, ranging from 92 to 165 games, many variables were not to scale across time. Some examples included number of home runs and stolen bases. To correct this problem, we converted all variables to per-game averages to ensure that season length would have no impact on our model.

Once we completed our data cleaning steps, we began to examine our regressors through plots and correlation matrices. It became obvious that even though we had over 100 potential regressors in our dataset, many of the regressors were highly correlated. Also, our per-game scaling revealed more duplicate columns (e.g. "Hits per game" vs. "Hits per nine innings"). Consequently, we likely had far fewer than 100 regressors with relatively low collinearity. In an effort to further cleanse our data, we identified and removed all highly correlated regressors which were less effective at explaining our response variables, number of wins, and win probability.

With our dataset whittled down to below 30 potential regressors, we implemented backward stepwise regression in an attempt to identify an optimal logistic and linear regression model. The resultant models were low quality, and we decided to re-insert the regressors we had deleted, let the stepwise search method identify optimal models, and then examine the variance inflation factors to eliminate

highly correlated regressors while attempting to maintain a high R-Squared value. This would ensure we had not removed any potentially useful regressors. With our dataset back up to approximately 50 regressors we implemented both forward and backward steps. We also split our dataset into a train and test set, assigning 99%, or 2,397 individual seasons, to train and 1%, or 25 seasons, to test. This would provide us with an additional tool to examine our out-of-sample model performance.

One caveat about the automated search methods is that they are more of a blunt instrument than a scalpel. Therefore, once we completed each of the four stepwise regressions (forward linear, backward linear, forward logistic, and backward logistic) we had to devise a way to eliminate any unnecessary regressors. To do this, we examined the current model p-values. Starting with the output from each of the stepwise regressions, we would identify the regressor with the highest p-value. If that regressor's p-value was greater than 0.05, we would drop that regressor from the current model, re-fit the model, and perform the test again. Our reduction was considered done once all the regressors in our current model had a p-value less than or equal to 0.05.

```
Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                       68.20945    9.46003   7.210 7.47e-13 ***
earned_run_avg_plus_pit            0.30814    0.01957  15.744  < 2e-16 ***
onbase_plus_slugging_plus_off      0.29578    0.02629  11.250  < 2e-16 ***
RBI_off                           11.96363    0.44725  26.749  < 2e-16 ***
AB_off                            -0.91303    0.27294  -3.345 0.000835 ***
batting_avg_off                    9.64278   18.65769   0.517 0.605326
LOB_off                           -0.06861    0.31468  -0.218 0.827422
XZB_off                           -2.25627    0.60905  -3.705 0.000217 ***
SH_off                             1.12718    0.60103   1.875 0.060860 .
leagueNL                           0.86872    0.21172   4.103 4.21e-05 ***
LOB_pit                            4.53489    0.53276   8.512  < 2e-16 ***
fip_pit                           -3.65450    0.46048  -7.936 3.18e-15 ***
batters_used_off                  -0.09723    0.02085  -4.663 3.29e-06 ***
BB_pit                            -5.86157    0.53264 -11.005  < 2e-16 ***
H_pit                             -7.40852    0.42875 -17.279  < 2e-16 ***
BK_pit                            -3.04800    2.95825  -1.030 0.302956
SB_off                             2.25868    0.41221   5.479 4.72e-08 ***
WP_pit                            -3.06597    1.40273  -2.186 0.028934 *
strikeouts_per_base_on_balls_pit -1.06240    0.46705  -2.275 0.023012 *
SO_off                            -0.06149    0.17042  -0.361 0.718281
age_bat_off                        0.30424    0.07746   3.928 8.83e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.36 on 2376 degrees of freedom
Multiple R-squared:  0.8729,    Adjusted R-squared:  0.8718
F-statistic:   816 on 20 and 2376 DF,  p-value: < 2.2e-16
```

We decided to begin with forward stepwise regression, since it is a somewhat less thorough method than backward stepwise. We say that it is less thorough because with forward selection, we may not look at all of the regressors and how they interact together. This means there is a chance some potentially beneficial relationships were not examined. However, even though the forward stepwise approach isn't perfect, it would give us a feel for our data. The results of our model are shown to the right:

```
Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)
(Intercept)                     0.1359108  0.3023632   0.449 0.653074
earned_run_avg_plus_pit         0.0028073  0.0008062   3.482 0.000497 ***
onbase_plus_slugging_plus_off   0.0027391  0.0009645   2.840 0.004514 **
R_pit                          -0.3591572  0.0168655 -21.295  < 2e-16 ***
runs_per_game_off               0.3624882  0.0156463  23.168  < 2e-16 ***
AB_off                         -0.0278894  0.0095299  -2.927 0.003428 **
batting_avg_off                 0.8524303  0.5730414   1.488 0.136868
LOB_off                         0.0102797  0.0105354   0.976 0.329200
XZB_off                        -0.0250097  0.0197397  -1.267 0.205165
SH_off                         -0.0121422  0.0195283  -0.622 0.534090
leagueNL                        0.0107900  0.0072297   1.492 0.135577
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8572.82  on 2396  degrees of freedom
Residual deviance:  921.09  on 2386  degrees of freedom
AIC: 14096
```

Setting aside the fact that there are a lot of regressors included in the model, there are several notable aspects. First, we noticed the model has a high f-stat, high adjusted R-squared, and a low p-value. It is

very likely that at least one of the coefficients in this model is not equal to 0. To confirm this, we ran an f-test. The results below show that indeed, this model is significant and that at least one of the regressors is not equal to 0.

```
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   2396 355325
2   2376  45164 20    310161 815.86 < 2.2e-16 ***
```

One drawback of our model is obviously the number of regressors. This many regressors makes model interpretability more challenging. Using the iterative approach of analyzing p-values we mentioned at the outset, we were able to reduce the model further. Once we removed all insignificant regressors, we employed the partial f-test. This test would allow us to say, with high confidence, whether the regressors we had identified could indeed be removed. The results of the partial f-test are below.

```
  Res.Df   RSS Df Sum of Sq      F Pr(>F)
1   2380 45199
2   2376 45164  4    35.229 0.4633 0.7627
```

Based on the results of the partial f-test,, we can conclude with a high degree of confidence that the 4 variables we identified as candidates for removal, can in fact be removed.

After the forward stepwise linear model approach, we turned to the forward stepwise logistic regression. The results from our forward stepwise search can be seen below:

Just like the forward linear model, our identified model has a high f-stat, high adjusted R-squared, and low p-value; also we have a large number of candidate regressors. To confirm our model was better than the null model, we performed a deviance of fit test comparing the full model to the null model. The results below show this model is significant and that at least one of the regressors is not equal to 0.

```
> 1-pchisq(fwd_glm_null$deviance-fwd_glm_full$deviance,1)
[1] 0
```

Once we confirmed our full model was significant, we removed all variables that seemed statistically insignificant. We used the deviance of fit test to compare the full and reduced model.

```
> 1-pchisq(fwd_glm_reduced$deviance-fwd_glm_full$deviance,1)
[1] 0.006525068
```

Based on the results of our deviance of fit test, we can conclude with a high degree of certainty that the reduced fit model better fits the data than the full model.

With forward stepwise completed for both linear and logistic regression, we turned to backward stepwise regression, which as we mentioned before, is a bit more thorough. The results of our backward stepwise linear regression model can be seen below:

```
Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                      70.58198   11.37169   6.207 6.36e-10 ***
X2B_off                          -3.77956    0.63289  -5.972 2.70e-09 ***
BB_off                            0.73960    0.35757   2.068 0.038710 *
HBP_off                           2.17107    1.35982   1.597 0.110491
LOB_off                          -0.78298    0.43749  -1.790 0.073627 .
SB_off                            1.56984    0.42322   3.709 0.000213 ***
SO_off                            0.16900    0.16194   1.044 0.296765
age_bat_off                       0.36551    0.08123   4.499 7.14e-06 ***
batters_used_off                 -0.09238    0.02207  -4.186 2.94e-05 ***
onbase_plus_slugging_plus_off     0.43642    0.02238  19.498  < 2e-16 ***
runs_per_game_off                 8.97949    0.44188  20.321  < 2e-16 ***
BB_pit                            0.69569    0.37048   1.878 0.060532 .
BK_pit                           -7.14316    3.09599  -2.307 0.021128 *
LOB_pit                           2.03046    0.60038   3.382 0.000731 ***
batters_faced_pit                -2.98715    0.34021  -8.780  < 2e-16 ***
earned_run_avg_plus_pit           0.47768    0.01672  28.565  < 2e-16 ***
fip_pit                          -5.66770    0.45210 -12.536  < 2e-16 ***
strikeouts_per_base_on_balls_pit  0.25125    0.46597   0.539 0.589808
leagueNL                          1.70048    0.21240   8.006 1.83e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.588 on 2378 degrees of freedom
Multiple R-squared:  0.8592,    Adjusted R-squared:  0.8581
F-statistic: 806.1 on 18 and 2378 DF,  p-value: < 2.2e-16
```

Just like the forward linear model, our identified model has a high f-stat, high adjusted R-squared, and low p-value, and again like the forward linear model, there appears to be a large number of candidate regressors. To confirm our model was better than the null model, we performed an f-test comparing the full model to the intercept-only model. The results below show that indeed, this model is significant and that at least one of the regressors is not equal to 0.

```
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   2396 355325
2   2378  50026 18    305299 806.25 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

After we systematically reduced the number of regressors to only the significant ones, we ran a partial f-test to ensure the candidate regressors identified for exclusion, could indeed be excluded. The results below show that indeed the reduced model does a better job than the full model.

```
Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)           -1.017e-01  3.650e-01  -0.279   0.7805
LOB_off                1.786e-02  1.077e-02   1.658   0.0974 .
R_off                  4.076e-01  6.717e-03  60.679   <2e-16 ***
BB_pit                 1.225e-02  1.080e-02   1.134   0.2570
ER_pit                -4.280e-01  1.277e-02 -33.524   <2e-16 ***
LOB_pit               -4.783e-02  2.127e-02  -2.249   0.0245 *
batters_faced_pit      4.006e-05  1.322e-02   0.003   0.9976
strikeouts_per_nine_pit  2.841e-02  2.443e-03  11.630   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8572.8  on 2396  degrees of freedom
Residual deviance: 1139.7  on 2389  degrees of freedom
AIC: 14309

Number of Fisher Scoring iterations: 3
```

```
  Res.Df   RSS Df Sum of Sq      F Pr(>F)
1   2382 50160
2   2378 50026  4    133.78 1.5898 0.1743
```

The results of our backward stepwise logistic regression model can be seen to the left.

Just like the other three models, our identified model has a high f-stat, high adjusted R-squared, and low p-value, and we have a large number of candidate regressors, though significantly less than what we identified for linear regression. To confirm our model was better than the null model, we performed a deviance of fit test comparing the full model to the null model. The results below show that this model is significant and that at least one of the regressors is not equal to 0.

```
> 1-pchisq(bkwrd_glm_null$deviance-bkwrd_glm_full$deviance,1)
[1] 0
```

With confirmation that our full model was significant, we again reduced the number of regressors until only the significant ones remained. We compared the null deviance for the full and reduced model, and the results are shown below:

```
> 1-pchisq(bkwrd_glm_reduced$deviance-bkwrd_glm_full$deviance,1)
[1] 0.03778048
```

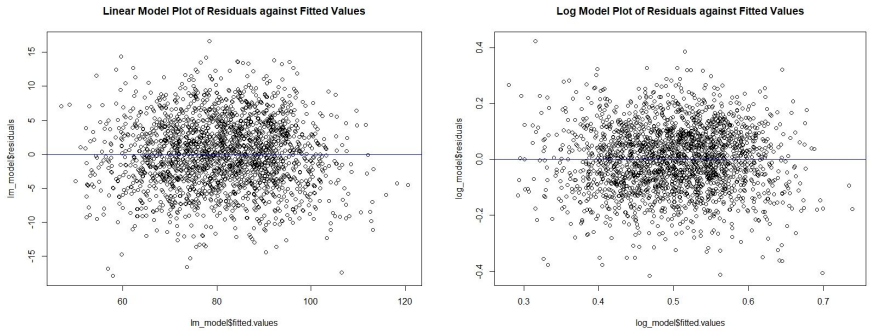
With the identification of four significant models, we turned our focus on selecting the best models. To narrow down our choice between the forward and backward linear regression models, we turned to the Adjusted R-squared. Though the forward linear regression model had two more predictors in the model, it still had a higher value at 0.8719 (compared to 0.858 for backward). This, coupled with the lower residual standard error, led us to choose the forward stepwise regression model as the best model to predict the number of wins in a season.

We performed a similar analysis for the logistic regression models, but instead of using Adjusted R-squared as our metric, we evaluated the deviation of fit tests for the forward and backward logistic regression models. We elected to choose the forward stepwise logistic regression model because its 0.0065 deviance of fit value is smaller than the backward stepwise model, which had a deviation of fit value of 0.0377.

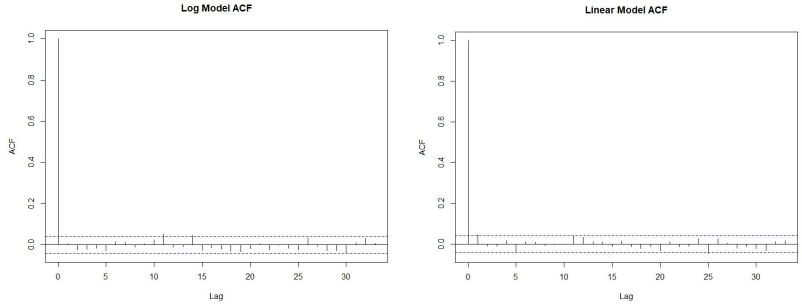
Since both of these models are statistically significant and explain a high percentage of the variance/deviance, we can utilize both of these models, one for when we wanted to predict full season results, and one when we wanted to predict individual game results; however, we prefer the forward stepwise logistic regression model. The reasons for this preference are three-fold: first, the logistic regression model has fewer predictors, making it more interpretable; second, the logistic regression model, since it represents the likelihood of winning one game, can scale to a series of games, or even an entire season of games; and finally, using the test data, we compared the root mean square error of the linear regression model (formula: sqrt(sum((predict(fwd_lm_reduced,test) - test$wins)**2) / nrow(test)) to the scaled version of the logistic regression model (formula:

sqrt(sum((predict(fwd_glm_reduced,test,type="response") - test$wins_pct)**2) / nrow(test)) * 162) and found that the logistic regression model had a root mean square error of approximately 3.98 games, while the linear regression model had a root mean square error of approximately 4.21 games. To reiterate, we think both models have some predictive power, but given the reasons mentioned above, we would slightly prefer the logistic regression model to the linear regression model if we had to only select one.

After we decided on our final models, we wanted to check to see if assumptions were met. To assess the distribution of the errors, we examined residual plots for both of our models.

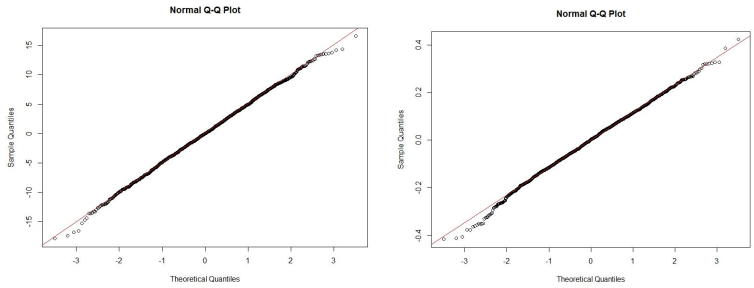


The residuals were evenly distributed which suggests that the errors have a mean zero. In addition, both residual plots appear to display constant variance regardless across all fitted values. Thus, the plots suggest that the residuals follow a distribution with a mean of zero with homoskedasticity. After assessing the residual plots, our next step would be to check for autocorrelation amongst the errors. Since our dataset spans across many different years, we wanted to make sure that our errors wouldn't be correlated at certain lags. Significant autocorrelation of errors would suggest that there is information left unaccounted for that would be helpful for our model. The ACF plots for both of our models are displayed below.



The plots satisfied assumptions for the most part, but both plots had two instances of autocorrelation significance. For the sake of model simplicity, we determined that that the significance of the error autocorrelation at these lags were negligible. Since they barely surpassed the rejection region, we determined that the effect of not accounting for this potential autocorrelation would have a minimal impact.

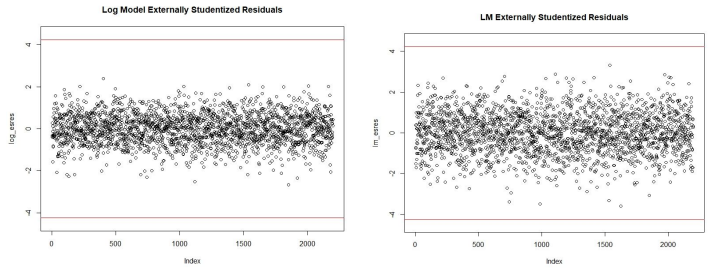
The last assumption to check was normality errors. To verify normality, we produced QQ plots for both models.



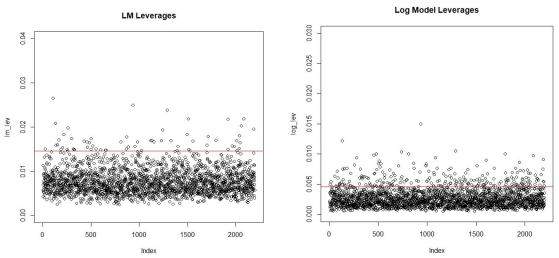
Linear Model Logistic Model

In both plots, a clear linear relationship is displayed. There is some minor deviation from the fitted lines toward near both ends of the plots, but both plots ultimately display strong linear relationships. The linearity suggests that our errors follow a normal distribution, and so our final assumption is satisfied.

After checking to see assumptions were met, our next step was to address potential outliers. The plots of externally studentized residuals are shown below.



All points for both models fall within the boundaries, which suggest that weren't any significant outliers in the response variables. Next, we examined potential outliers in the predictor variables. Plots displaying the leverage for each datapoint are shown below.



The plots display the existence of many leverage points which exceed the 2p/n boundary. Points that exceed this boundary are considered to be “high leverage” and need to be evaluated because they have the potential to affect the accuracy of our models. To address this, we then computed Cook’s distance for all data points. Since Cook’s distance is a statistic that measures how much the regression would change if a certain observation is removed, we decided to remove these high leverage data points if their corresponding Cook’s distance was also significant. After computing the Cook’s distance for all these points, we found none of them to have values surpassing our cutoff value (.96 for linear model, .87 for log model). Thus, we kept all data points in our model because we determined that any negative affect they may have was negligible.

In conclusion, we ended up with two models which work well in different ways to predict baseball. Our linear regression model provides 16 significant factors that teams can focus on to improve their number of wins in a season. With an adjusted R-squared of 0.87, the teams can be assured that our model explains 87% of the variation in the number of wins per season, and the regression assumptions proved valid. By contrast, our logistic regression model, with only 5 predictor variables, could be a simpler way for teams to focus on their odds of winning a single game. The equation is "win_odds = exp(0.12 + 0.0025*earned_run_avg_plus_pit + 0.00243*onbase_plus_slugging_plus_off - 0.3678*R_pit + 0.3749*runs_per_game_off - 0.0185*AB_off)". This can be interpreted as: a 3-unit increase in the number of runs scored increases the odds ratio of winning the game by exp(3*0.3749) = 3.079, when the other predictors are held constant. This model may seem intuitive, but it provides simple, clear explanatory power when predicting baseball game results.