**Strikeout Predictions**

Ben Popernik

Amardeep Singh

Shadhana Palaniswami

Shukran Amdeen

**Abstract**

Major League Baseball is an American professional sports league with thirty individual teams located across the country. The pitcher starts every play in baseball, making them a key part in the team’s defense. The pitcher’s job is to throw the ball to the batter and try to get the batter out. Pitchers can record outs via strikeout. Major League Baseball teams search for pitchers who record a high number of strikeouts in a game, which makes it more difficult for the other team to score runs. With analytics being used more in sports, Major League Baseball teams are collecting data to evaluate pitchers. To help a baseball team win more games, analyzing pitching data to find which statistics result in more strikeouts can help teams identify which pitchers they want to recruit to their team. We propose a polynomial regression model that uses the most relevant pitching statistics to predict the number of strikeouts in a single season. Our findings indicate the most important statistics to predict strikeouts are innings pitched, walks per 9 innings, and the strikeout to walk ratio.

**Introduction**

“Major League Baseball (MLB), North American professional baseball organization that was formed in 1903 with the merger of the two U.S. professional baseball leagues,” (Augustyn, 2019). 116 years later, Major League Baseball has become one of the most profitable sports organizations in the United States (Amoros, 2016). This largely has to do with the history of professional Baseball in the United States. “Approximately a dozen entirely professional teams existed by 1869, most notably the all-salaried Cincinnati Red Stockings, whose players earned from $600 to $2,000” (Reiss 2017). Today, Major League Baseball teams can make more money if they maintain a good win percentage (Ablin, 2019).

A high win percentage can be acquired from a good offense that scores a lot of runs, a good defense, and pitching. For pitchers, having a lower number of runs given up increases the chances of winning more baseball games that season. “...strikeouts improve a pitcher's E.R.A. by reducing it, whereas walks tend to increase it” (Spurr, 1994). This report is designed to find which pitching statistics can help predict the number of strikeouts a pitcher has in a single season. Our findings can help teams hire the right pitchers to play for their organization, along with player develop in both Minor and Major League Baseball.

For years baseball managers and organizations were skeptical about certain statistics and analytics. “‘Managers tend to pick a strategy that is the least likely to fail, rather than to pick a strategy that is most efficient," Said Palmer. " The pain of looking bad is worse than the gain of making the best move.’” (Lewis, 2013). The quote describes how managers were afraid to look for new strategies which might result in failure and potentially their job. Moneyball discusses the success of the early 2000’s Oakland Athletics baseball team which had among the lowest salary paid to their players while maintaining a high win percentage (Lewis, 2013). Since then, the success of the Athletics manager, Billy Bean, has caused other managers and MLB teams to hire personnel to analyze baseball statistics thus launching the “sabermetrics revolution”. More importantly, analytics is used to make current players perform better (Latson, 2019).

A strikeout (SO) occurs when 3 pitches, resulting in strikes, are thrown in a single at bat resulting in an out. A strike is recorded in one of three ways. One is where the batter swings his bat and fails to make contact with the baseball. The second way is when the batter makes contact with the baseball and the ball lands in foul territory. The third way a strike can be recorded is based on the Umpire who determines if the ball was pitched within the strike zone, meaning the batter could have attempted to hit the ball but didn’t.

To help predict which variables equate to the season strikeout total, we gathered our data from the 2012 to 2018 Major League Baseball seasons. Each observation will show the statistics of a pitcher’s single season. This dataset contains pitchers from the entire league with at least 50 pitches thrown that season. Some pitchers may appear twice since the observations are based on a pitchers single season performance, i.e if a pitcher pitched 3 seasons between 2012-2018 they will have three observations in the dataset.

With polynomial regression techniques and the dataset described, our group will attempt to create a regression model to predict the number of strikeouts a pitcher throws in a given season. We hypothesize that there are predictors that are significant enough to predict the number of strikeouts in a given season.

**Literature Review:**

A research study which influenced our decision to write a regression model on strikeouts is Rebecca L. Sparks and David L. Abrahamson’s (2005) “A Mathematical Model to Predict Award Winners”. The report explores coming up with a model using baseball statistics to predict the likelihood of end of season awards, specifically the Cy Young Award. Sparks (2005) discusses the importance of end of season pitching statistics that determine the winner of the Cy Young Award, an award given to the best pitcher during the season. Sparks highlights the importance of the following pitching statistics, earned run average (ERA), wins (W), losses (L), team winning percentage (TWP), and strikeouts (K). “Earned run average is the next most important to the voters, followed by strikeouts.” (Sparks, 2005).

Of the statistics that sparks has decided to include in her model, strikeouts carries the least amount of weight (Sparks, 2005). However, the model still shows the importance of strikeouts in predicting the winner of the Cy Young Award. “A starter with just 50 strikeouts earns a zero in the parameter *p­lj1* while one who fans 383 to tie Nolan Ryan’s record gets ten points, and so on.” (Sparks, 2005). From the predictors in this model, a score of 10 marks a historic achievement for that specific category (Sparks, 2005). The most influential predictor in this model was wins followed by losses, ERA, team winning proportion, and strikeouts respectfully. Team winning percentage is an important statistic because it can show how good or bad a team performs outside of the pitcher’s performance. The model was able to predict correctly the first, second, and third place Cy Young Award winners between 1994 and 2002 MLB seasons (Sparks, 2005). Since strikeouts is a predictor in Sparks model measuring the best pitcher in the league, our group focused on statistics that created opportunities for strikeouts.

**Methodology**

1. Data cleaning & collection:

Obtaining the dataset proved to be a somewhat simple endeavour as all the statistics were available in a consolidated location online via fangraphs.com (FANGRAPHS, 2019). After obtaining our original dataset, we removed a significant amount of columns of baseball statistics that were not predictors in our model. Originally the dataset contained more than 300 columns and we only needed 10 columns of statistics for our predictors and independent variable. Thus, we deleted the rest and moved our new dataset from a .csv file to a .txt file.

The statistics we chose to be our predictors were: innings pitched (IP), home runs (HR), walks (BB), Balls, Strikes, Pitches, strikeouts per nine innings (K/9), walks per nine innings (BB/9), strikeout to walk ratio (K/BB). Our dependent variable is strikeouts (SO). With our domain knowledge, we were able to discern that these statistics were all very relevant to our main objective: predicting strikeouts per season. Our dependent and independent variables are numeric values. Innings pitched is the amount of innings the pitcher has pitched during an entire season. HR is the number of home runs a pitcher allows in a season which represent a pitch that was hit far enough to score a run. BB is the number of walks the pitcher allows in a season, a negative outcome resulting in the batter getting on base. SO is the number of strikeouts a pitcher collects in a season. Balls is the number of pitches thrown out of the strike zone in a season. Strikes is the number of pitches thrown inside the strike zone in a season. Pitches is the number of times a ball is thrown to the catcher in a season. K/9 is the ratio of strikeouts per nine innings in a season. BB/9 is the ratio of walks per nine innings in a season. K\_BB is the ratio of strikeouts for every walk in a season.

The smaller dataset we collected and cleaned from the original datafile contained no null or invalid statistics. The population of interest in this project is all Major League Baseball pitchers specifically from 2012 to 2018. Out of all MLB pitchers, our sample is a record of 2,337 pitchers' statistics for a given season. For example, pitcher A could have an observation for his 2017 season as well as his 2018 season statistics.

The parameter of interest for this project is the number of strikeouts in a given baseball season. Parallel to the parameter of interest, the number of strikeouts in a given baseball season is our dependent variable using our sample. With the number of strikeouts being our dependent variable, the remaining 9 statistics are left to be predictors for regression analysis.

1. Data exploration:

After collecting our data, cleaning it and ensuring that it is ready for analysis, we began trying to visualize it. We first fit the dependent variable, SO, to a histogram to see the general distribution of strikeouts throughout the seasons in our parameter of interest (Appendix A). Due to the fact that the distribution of SO is unimodal but not normal (based on the kurtosis being 0.0859 and skewness being 1.110 (Appendix B)) we identified transformations were eventually necessary for SO. Along with the skewness value of 1.11, the median of SO is 78 while the mode is only 54 (Appendix C). Considering that the median 78, the standard deviation of 50.33 shows that the deviation from the mean shows that the data isn’t too dispersed nor too condensed. Becauses the median is greater than the mode, the distribution of SO is rightly skewed as the Interquartile range is 69 with a Q1 value of 57 and a Q3 value of 126. The range of SO in our dataset is 288 with a minimum value of 20 and a maximum value of 308. Considering there are pitchers who’ve only had 20 strikeouts in a season while others have had 308 strikeouts can test our model for not only starting pitchers or relief pitchers, but all pitchers in the MLB. This spread of data confirms our regression model we create will be practical.

Another significant tool we’ve used throughout the analysis is the Pearson’s Correlation Coefficients table as well as the matrix scatterplot (Appendix D). The matrix scatterplot shows a clear linear relationship between Balls, Strikes and Pitches. Other predictors have a general linear relationship with SO. However, some are more strongly linearly associated than others. For example, IP, HR and BB are more linearly associated to SO than K\_BB, K\_9 and BB\_9. The association between SO and the predictors from highest to lowest is as follows: K\_9, K\_BB, IP, BB, HR, BB\_9, Strikes, Balls and last, Pitches with 0 association as Pitches is defined as Balls + Strikes by SAS (Appendix E, F, and G). Balls added to Strikes proved to be exactly collinear to Pitches. This unique relationship we found between our predictors proved that the original regression is not of full rank. When running the first regression, SAS tries to find a non-zero linear relationship between SO and every predictor to formulate the first regression analysis and parameter estimates. However, when the model is not of full rank, there are an infinite number of solutions. In other words, the exact collinearity and dependency between Strikes, Balls and Pitches allows for Pitches to be removed from the model as Pitches exists within Balls + Strikes.

1. **Analysis**
2. Linear regression model

When running the regression model the first time, we ran into the above correlation issue between Balls, Strikes and Pitches from our matrix scatter plot(Appendix D). SAS was able to detect that the sum of Balls and Strikes was Pitches (Appendix G). Using domain knowledge, we decided to permanently remove Pitches from our model since Strikes and Balls were better predictors for our model. After removing pitches, another linear regression model was fit to the data. The residual plots for this new model without Pitches showed patterns for the predictors(Appendix K). The new model is seen in Appendix J without Pitches but including Balls and Strikes - therefore, eliminating the exact collinearity issue. While checking for assumptions, the normal curve (Appendix L) was not linear and was forming an “S” shape along with patterns in the residuals (Appendix K) suggesting transformations were necessary.

To eliminate the patterns that appear throughout the predictor’s residuals, we attempted to transform the linear regression model using log (Appendix M) and square root (Appendix T) transformations. Each transformation attempt resulted in more issues with the residual plots for the predictors. We applied log transformations on both our dependent and independent variables. There were 4 predictors that had VIF values above 10, otherwise the model had a high F Value, Adj R-Square, and low RMSE (Appendix M). The log SO proved to create more issues than we first had. By transforming SO to log(SO), the residuals plots for IP, Balls, Strikes and K\_9 show an extreme inverted ‘U’ shape (Appendix N). However, it is important to note that the studentized residual for the log transformation shows a very clear linear line at a 45 degree angle. Either way, we cannot neglect the residual plots. For that reason, we try another transformation. The following model we tried to fit included a log transformation for all variables. One of our predictors, HR, has observations with a ‘0’ as it’s possible for a pitcher to allow 0 homeruns in a season. To be exact, there are exactly 3 observations where pitchers allowed no home runs in a given season (Appendix P). When transforming every variable into a log of that variable, the model had a perfect 1.0 adjusted R-squared (Appendix P). When analyzing the residual plots for that model, we find that there are slight yet very noticeable patterns in all of the residual plots (Appendix Q). There is a slight bow-tie pattern in log(IP), log(Strikes), log(Balls) as well as an uneven, horizontally dispersed pattern in log(HR) (Appendix Q). For that reason, ultimately, log transformations were proven not useful for this dataset.

Next, we applied the square root transformation to all the variables. As opposed to the log transformation, a square root transformation did not cause any observations to be eliminated as a square root of every value is valid. For example, while a log of 0 is invalid, a square root of 0 is 0. Furthermore, the residuals for log results showed patterns in the shape of a lowercase n (Appendix Q) so we decided to try a different transformation method. Our next transformation was the square root transformation applied on both the independent and dependent variables (Appendix T). The residual plots for the square root transformation show clear patterns throughout the 8 graphs. The residual plots for sqrt(IP), sqrt(BB), sqrt(Balls), sqrt(Strikes), and sqrt(K\_9) show the bow-tie shape in some fashion (Appendix U). This model passed the goodness of fit test with a Pr>F value less than 0.0001 with a high F Value of 26019.0 and a low RMSE, but there were VIF values above 10 (Appendix T). While an adjusted R-squared of 0.9889 is considered very high, it wasn’t as high as the log transformation residual plots.

Considering that the residual plots were creating issues in all of the transformation operations, we started to look for other clues we might be missing. After, we went back to the original linear model without any transformations (Appendix J), we noticed that the predicted value by studentized residual plot is polynomial (Appendix W). For that, explored the possibility that we might have to build a polynomial regression model.

1. Second Order Polynomials:

Our first attempt at creating a polynomial regression was running selection methods using second order polynomial regression. The selection methods we used were backward, forward and stepwise selection. First, we ran a backward selection method.

*Backward Selection Method*

When we used backward method on our model, because of the nature of polynomial regression it formulated 19 new predictors where some were a combination of different variables(Appendix X). This model is a 2nd order polynomial, so some of the predictors came out as a product of two variables. Therefore, we had to create the variables in the model in SAS to run a quadratic model we called quadratic\_backward. After creating this model, we checked the residuals distribution of data around 0 (Appendix Y). Some residuals turned good like K\_9 and K\_9\*IP. However, the majority were forming patterns like funnel, cone and others that were not randomly scattered they were all towards the left of the chart(Appendix Y). While this model had a very high F value, we realized the VIF values were pretty high as well (Appendix Z). Upon reading the article introduced in class, we applied the concept of centering the VIF values using the formula given below (Allison, 2012):

New\_variable = variable - mean (Appendix AA).

After centering the VIF values, each VIF value was lower but would need manual removal of variables with high VIF values (above 10) one by one (Appendix AA). We started removing variables manually, we started by removing HR, and then we removed 8 other variables one by one while rerunning the model until all predictors had a VIF value of 10 or lower (Appendix FF). The resulting model contained 9 independent variables including Strikes, Balls\*Strikes, K\_9, Strikes\*K\_9, BB\_9, IP\*BB\_9, Strikes\*K\_BB, K\_9\*K\_BB and BB\_9\*K\_BB. These predictors all have VIF values of less than 6 and the model’s adjusted R-squared is very high at 99.2% (Appendix FF). The model ended up having a high F-value of 32192, however, we noticed that the residual plots were still a problem(Appendix GG). The residual plots exhibited funnel-shaped patterns that were troublesome (Appendix QQ). For that reason, we explored other selection methods.

*Forward Selection Method*

Next, we tried implementing a forward selection where we encountered 12 new variables which were a result of the forward selection method (Appendix RR). This model had a very high F value of 8.362E7, however the predictors had very high VIF values (Appendix SS). The majority of the residual plots had patterns like a less than sign or bow tie showing the data was not randomly scattered around 0 (Appendix TT). Therefore, we tried centerning the predictors to lower the VIF values - however, it didn’t get any better (Appendix UU) and residuals remained the same (Appendix VV). A pattern can be witnessed in studentized plot(Appendix WW) showing values weighted to the left side of the graph and a sideways V shape. This method couldn’t bring VIF values down to below 10, therefore, we considered moving onto one last selection method.

*Stepwise Selection Method*

When implementing the Stepwise selection method, we encountered 3 new variables that were a product of other predictors (Appendix XX).We repeated the same steps discussed in backward selection. This model had a very high F value of 2.277E8 with VIF values higher than 10 (Appendix XX). Even though the model had a perfect Adjusted R-squared value of 1.0, the residual plots proved to be pretty ok, but there were still some patterns present (Appendix YY). Even after centerning the values it didn’t get any better and residuals were still the same (Appendix AAA). In the normal plot(Appendix BBB), it can be seen that it is not linear. Also, the adjusted R-squared is considerably lower than the previous model’s value at 10.89% and the F value dropped to 72.39 (Appendix ZZ). After centring, the VIF values lowered drastically in comparison to the uncentered model (Appendix ZZ).

Considering that every 2nd order polynomial model we’ve created has had a variety of issues from absurd residual plot patterns to low adjusted R-squared values every centring the high VIF values, we decided to explore 3rd order polynomials.

1. Third Order Polynomials:

Since second order polynomial regression models didn’t provide any effective models, we decided to switch to third order polynomial regression models. Just like the second order polynomial models, we will create 3rd order polynomial models with stepwise, forward, and backward selection methods.

*Stepwise Selection Method*

We started with the original dataset and ran GLMSELECT with a stepwise selection method to combine our predictors together to create a new model (Appendix CCC). The stepwise selection method had 4 predictors. While all except one of the predictors had considerably high VIF values, the adjusted R-squared was a perfect 1.0 (Appendix DDD). Also, the model had a very high F value of 2.316E8 (Appendix DDD). This model also passed the goodness to fit test with a very low P-value of less than 0.0001. When we removed the predictors that contained the highest variance inflation values, the VIF values for the remaining 2 variables dropped considerably (Appendix EEE). We removed ipk9 and bb\_kbb because of high variance inflation to create our final stepwise third order polynomial model (Appendix EEE). This final model contained a high F value of 4.732E7 along with adjusted R-squared value equal to 1.0 (Appendix EEE). The variance inflation for both of the predictors were under 10. The only issue we encountered was the residual plots of our predictors at this point indicating transformation methods were necessary (Appendix FFF). We decided to run a square root transformations on the dependent and independent variables. Our results showed a strong F value of 4.337E7 and a perfect adjusted R-squared value of 1.0 and variance inflation values of all predictors falling under 10 (Appendix GGG). The residual plots with the square root transformation looked much better and the data was randomly spread around the zero line and well within the +/- 3 range (Appendix HHH). The normal curve plot for the square root transformation method showed a straight line making this a suitable model for testing and training sets(Appendix III). The stepwise selection method using a square root transformation proved to create a very sufficient and reliable model. However, we wanted to test other selection methods for the possibility of a better model. Our final model for the Stepwise Third Order Polynomial was the following:

Equation:

sqrt\_(SO) = 0.0211 + 0.0008\*sqrt(bbk9kbb) + 0.3322\*sqrt(ipbb9kbb)

Predictors:

bbk9kbb = (BB)\*(K\_9)\*(K\_BB)

ipbb9kbb = (IP)\*(BB\_9)\*(K\_BB)

*Forward Selection Method*

The forward selection method for the third order polynomial contained more predictors than stepwise, but less than the backward selection method. The forward selection method contained 23 predictors with an F value of 5.592E7 and a perfect adjusted R-squared value of 100% (Appendix JJJ). After getting our predictors from the forward selection method, we ran the proc reg procedure to check the residuals and removed variables with high VIF values. Afterwards we noticed the F Value dropped to 8770.49 and the adjusted R-Squared value dropped to 94.95% with all predictors having VIF values under 10 (Appendix KKK). The RMSE was also higher at 11.3199. We noticed that the residual plots showed patterns of waves suggesting we would need to perform a transformation on our model (Appendix LLL). The first transformation method we tried was a square root transformation on all the independent and dependent variables. After doing this we noticed the adjusted R-Squared value increased to 98.37% and the F Value increasing to 28109.6 (Appendix MMM). The only issue with performing the square root transformation method was the variance inflation values of sqrt(hr\_bb\_bb92) and sqrt(hr\_balls\_kbb2) were above the recommended threshold of 10 (Appendix MMM). To fix the variance inflation, we removed each variable in order from highest VIF value until all predictors had VIF values under 10 (Appendix NNN). Our final forward selection third order polynomial model had an adjusted R-Squared value of 98.13%. This model had an adjusted R-squared value similar to the adjusted R-Squared value from before we removed the predictors with VIF values greater than 10 (Appendix NNN). The F Value increased to 30728.1 and the RMSE dropped to 0.33330 indicating low errors and the model had a P-value of <0.0001 meaning it passed the Goodness of Fit test (Appendix NNN). After checking the residuals, the points were much more randomized around the zero line (Appendix OOO) compared to the model without transformations (Appendix LLL). The last things we checked were the Predicted Value residual plot and the Normal Curve Distribution. The Predicted Value residual showed randomized points around the zero line along with most points staying within the +/- 3 lines (Appendix PPP). The Normal Curve Distribution plot showed most points closely bunched together and had a very small bend to it (Appendix QQQ). We believe this isn’t enough of a bend to cause any concern and fitted a forward selection third order polynomial model resembling the following equation.

Equation:

sqrt(SO) = 2.9691 + 0.0003\*sqrt(bb\_balls\_strikes2) + 0.0417\*sqrt(bb\_k9\_bb92) + 0.0013\*sqrt(hr\_balls\_kbb2) + 0.0232\*sqrt(strikes\_k9\_kbb2)

Predictors:

bb\_balls\_strikes2= (BB)\*(Balls)\*(Strikes)

bb\_k9\_bb92 = (BB)\*(K\_9)\*(BB\_9)

hr\_balls\_kbb2 = (HR)\*(Balls)\*(K\_BB)

strikes\_k9\_kbb2 = (Strikes)\*(K\_9)\*(K\_BB).

*Backward Selection Method*

The backward selection ran on a third order glm select method gave us more predictors than the stepwise selection method (Appendix RRR). With 55 predictors, we ran another regression model using the 55 predictors where some are products of each other. After removing all the variables with high VIF values, the model remained with 5 predictors using IP, BB\*BB\_9, HR\*K\_9\*BB\_9, HR\*K\_BB, B\*K\_9\*K\_BB (Appendix SSS). The adjusted R-squared value for the new model is 0.9759 and the F value is 18,923.8 (Appendix SSS) while passing the goodness of fit test with a P-value of <0.0001. When running the residual plots, we found the residuals do not fit between +3 and -3 (Appendix TTT). We also noticed that some of the residual plots had some sort of pattern or shape. The residual plot for IP has a bow tie pattern while Balls\*K\_9\*K\_BB has a wave shape facing downward (Appendix TTT). To try to randomize the residuals for our predictors we decided to run transformations on our model. The first transformation we decided to do was a log transformation on our dependent variable as well as BB\*BB\_9, HR\*K\_9\*BB\_9, HR\*K\_BB, Balls\*K\_9\*K\_BB and an inverse transformation on IP (Appendix UUU). After quickly trying various transformations, we found that this one provided the most normal residual plots (Appendix VVV). This model had variance inflation values for all predictors above 10, but the F value and adjusted R-squared improved to 99.57% from 97.59% before the transformation (Appendix UUU). However, since the residuals were still not the most random, we decided not to move on without the backward selection method therefore not creating a model.

1. Splitting up the data into training and testing sets & building final models

After building a final model for the full dataset, we explored various selection methods in forming the model using the training set. First, we split our data into training and testing sets using a random seed number. The training set comprised of 75% of the data and the testing set comprised of the 25% of the data. The training set was used to create a new model. But first, we explored the forward selection method as well as the stepwise selection method in creating our final model. We built two models using the Training set, using the forward selection method and the stepwise selection method.

*Forward selection method using the training set:*

Our first model we made with our training set was using a forward selection technique for third order polynomial models. We noticed the F Value was very high and the predictors were similar to the Forward Selection model before transformations (Appendix WWW). We ran the regression model to check for residuals and variance inflation and noticed a strong Adj R-Squared, F Value, and RMSE (Appendix XXX). However, there were many variables with VIF greater than 10 (Appendix XXX). After removing the variables with high VIF values, our model consisted of 3 variables with VIFs less than 10(Appendix YYY). As VIFs were in good shape, we checked for residuals but they still were forming patterns and were not randomly scattered (Appendix ZZZ), implying that transformations were necessary. The transformation method we decided to run was a square root transformation on all our independent and dependent variables. We noticed an improvement in our F Value increasing to 47844.7 and our RMSE decreasing to 0.26752 while our adjusted R-Square stayed relatively the same at 98.79% (Appendix AAAA). However, the residuals were still not satisfactory (Appendix BBBB), showing a wave pattern for the sqrt\_ip\_balls\_kbb residual plot. Otherwise, the normal curve was quite linear with an Adj-r2 of 0.9879 (Appendix CCCC). Because of the residuals we tried a different transformation method, changing to a log transformation. After running the model again with log transformations, we noticed the F Value improving to 58388.9 which was higher than both the normal model and square root transformation and passed the goodness of fit test (Appendix DDDD) . The adjusted R-Squared improved to 99.01% and the RMSE decreased to 0.05018 while the predictors VIF values were less than 10 (Appendix DDDD). We again checked the predictors residuals and noticed they became more randomized around the zero line and were not forming shapes(Appendix EEEE). The predicted value residuals were also randomized around the zero line (Appendix FFFF). The normal curve for this model was also very straight and didn’t show any bending (Appendix GGGG). Our final training model was the following equation:

Equation:

log(train\_y) = -3.18880 + 0.28818\*log(k9) +

0.48449\*log(balls\_k9)+

0.24843\*log(ip\_balls\_kbb)

Predictors:

k9 = K\_9

balls\_k9 = (Balls)\*(K\_9)

ip\_balls\_kbb = (IP)\*(Balls)\*(K\_BB)

*Stepwise selection method using the training set:*

Our second third order polynomial training model we fitted was using the stepwise selection method which gave us 5 predictors (Appendix LLLL). We then ran a regression model with these predictors to check for variance inflation. We noticed there were three values that had high variance inflation and we decided to remove the values one by one manually (Appendix MMMM). After removing the three 3 variables there were 2 predictors left, bb9kbb and ipbb9kbb (Appendix OOOO). The adjusted R-Squared was 100% with an F Value of 3.483E7 (Appendix OOOO). This model passed the goodness of fit test with a Pr>F value less than 0.0001 and an RMSE value of 0.25026 which is low (Appendix OOOO). Although the predictor’s residual plots did look scattered around the 0 line (Appendix PPPP), we decided to try some transformation methods with hopes of improving the model. The transformation method we decided to apply on our dependent and independent variables was the square root transformation. This improved our original stepwise model by decreasing the RMSE value to 0.01268, keeping our Adj R-Squared value at 100%, and retaining a high F Value of 3.197E7 (Appendix QQQQ). The best part was the residuals were even better than before showing a random spread of data around the zero line (Appendix RRRR). VIFs were under 10 now. Our final stepwise training model uses the following equation:

Equation:

sqrt(train\_y) = 0.00235 + 0.00687\*sqrt(bb9\_kbb) +

0.33296\*sqrt(ip\_bb9\_kbb)

Predictors:

bb9\_kbb = (BB\_9)\*(K\_BB)

ip\_bb9\_kbb = (IP\*BB\_9)\*(K\_BB)

**IV. Results and Findings**

The training set gave us two separate models as seen above. The first model was based on the forward selection method using the log transformation. The second model was based on the stepwise selection method using the square root transformation. Each final model will be used to apply the testing set on. This is an effort to find the best fit model out of the two. First, we applied the testing set (comprised of 584 observations) on the forward selection method with the log transformation.

1. Applying the testing set on the first final model.

We applied the test set on the training model using the forward selection model with log transformation. This model contains 3 predictors. The RMSE and MAE values were a low 0.050075 and 0.040133 respectively (Appendix TTTT). The predicted values for the dependent variable come out to be very close to the original values. It can also be seen in the correlation between the predicted values and the log(SO) are also pretty well as the y\_hat value is 0.99520 (Appendix TTTT). Using the y\_hat value, we deduct that the adjusted R-squared value for this testing set is 0.9859 with a CV-R-squared value of 0.0058. Considering that these statistics are considerably high, we can deduce that this is a decent model.

When comparing the performance of the training set to the testing set for this model, they both perform relatively similar. However, the training set has a higher RMSE value of 0.30205 and a lower adjusted R-squared value of 0.9846. Compared to the testing set, the training set performs slightly worse.

1. Applying the testing set on the second final model.

After applying the testing set to the first model, we had to apply the testing set to the second model as well. The second model is a stepwise selection method ran with a square root transformation. This model contains 2 predictors. The RMSE and MAE values were a low 0.012671 and 0.010422 respectively (Appendix UUUU). The predicted values for the dependent variable come out to be very close to the original value of SO after retransformation (Appendix VVVV). The correlation coefficient for the predicted value for sqrt(train\_y) is 0.9999 (Appendix UUUU). Using the y\_hat value, we deduced that the adjusted R-squared is 0.999 and the CV-R-squared value is 0.0001.

Comparing the performance of the testing set to the training set, we see that the training set performs slightly better. For example, while the adjusted R-squared for the training set is a perfect 1.0, the adjusted R-squared for the testing set is only 0.9999. Moreover, the RMSE values for the two sets are almost indifferentiable considering that the RMSE value for the training set is 0.01265.

1. Comparing the performance of the 2 models

To analyze the performance of the 2 models, we compare how each model performs itself when using the training set and when using the testing set. To do this, we will create a small table to better visualize the performances. The table below compares training set and testing set performances for each respective model. In the table, it is evident that all 4 performances are considerably high. However, to truly select the best model, we will ensure that the testing set is performing up to par or better than the training set.

The first model’s testing set is performing better than the training set as evident in the RMSE values specifically. The testing set has lower errors with an RMSE value of 0.05 compared to the training set’s RMSE value of 0.30. Also, the adjusted R-squared values have a difference of about 0.01. While this isn’t a significant difference, the testing set does perform better in the first model generally.

The second model’s testing set is performing very slightly worse than the training set. This can be seen in the RMSE value differences as well as the adjusted R-squared difference. The difference in RMSE values is about 0.00002 while the difference in the adjusted R-squared values is approximately 0.00001. For this reason, the training set performs slightly better.

|  |  |  |
| --- | --- | --- |
|  | **Model #1:** (forward selection using log transformation) | **Model #2:** (stepwise selection using sqrt transformation) |
| *Training set* | RMSE - 0.30205  R-squared - 0.9846  Adj-R-squared - 0.9846  GOF - P-value is <0.0001  Residuals - Great! | RMSE - 0.01265  R-squared - 1.0  Adj-R-squared - 1.0  GOF - P-value is <0.0001  Residuals - Great! |
| *Testing set* | RMSE - 0.050075  MAE - 0.040133  R2 - 0.9904  Adj- R2 - 0.9859  CV- R2 - 0.0058 | RMSE - 0.012671  MAE - 0.010422  R2 - 0.9999  Adj- R2 - 0.9999  CV- R2 - 0.0001 |

Next, we must compare the performances of the models to each other. The first model has higher RMSE values for both the training and testing set than the second model’s. Also, while the adjusted R-squared values are still high for both models, the second model has perfect or near perfect values. Furthermore, the CV-R-squared values for the testing sets have a significant difference of about 0.005. For this reason, our final model will be the second which is ran by using a stepwise selection method and a square root transformation.

**V. Conclusion**

Throughout the development of each regression model, we encountered several roadblocks when checking the diagnostics and assumptions. First, we ran a linear regression model that soon proved insufficient. Second, we ran a second-order polynomial regression. While providing much better results, the second-order polynomial regression brought about more issues. The biggest issue that came with the second-order polynomial regression models is the effects of centering the absurdly high VIF values. After centering the VIF values, our residual plots for each predictor had complex patterns that were not fit for a proper regression model. For that reason, we moved on to third-order polynomial regression. Third-order polynomial regression allowed us to eliminate absurd patterns in the residual plots after transforming certain variables.

After splitting our data into training and testing sets, we built 2 final models using stepwise and forward selection methods as well as various transformations. We ended up with 2 final models: 1. Forward selection method using log transformations and 2. Stepwise selection method using square root transformations. These 2 respective models gave us incredibly high statistics as the adjusted R-squared values for each were close to perfect for each training and testing sets.

With the final 2 models being compared above in the Results and Findings section, we ended up with one final model that proved to be the best out of the bunch. Our *final model* is:

Equation:

sqrt(SO) = 0.00235 + 0.00687\*sqrt(bb9\_kbb) +

0.33296\*sqrt(ip\_bb9\_kbb)

Predictors:

bb9\_kbb = (BB\_9)\*(K\_BB)

ip\_bb9\_kbb = (IP\*BB\_9)\*(K\_BB).

With this final model, we can determine that walks per nine innings (BB\_9), strikeout to walk ratio (K\_BB), and innings pitched (IP) are the most important statistics for finding the number of strikeouts in a season.

**Future Work**

Analysis done in this paper suggests many opportunities to explore. However many statistics failed to get accurate results for baseball because they are too basic but a regression model is one reliable consideration to make (Dennis Moy. 2006). This model can help with strikeout prediction, player performances and many other aspects which are discussed further:

**College athletes performances:** This model can be very useful when evaluating the performance of college athletes in baseball. This can predict the strikeouts rates with the professional players and can actually help them identify their weak spots and where they need to improve. This model can be used to see the regular improvement of players and predict their performance in text tournament based on their evaluation.

**Performance Prediction:** As a major part of this model is predictive analysis. This model can help with prediction of players who are playing in the minor leagues and can help see their performance prediction, in the tournaments. It can help a minor league player to evaluate their performance prediction with a major league player and see what can be done to be in the major league.

While recruiting for new players not only from college athletes but for professional players as well Secounts can use this model to see the performance and area of improvements, which can really help picking up right players for the leagues. A richer and enhanced recruiting experience can be obtained with this model.

**Rapid growth:**  The baseball database is pretty dynamic with regular updation. This model when uses that data to create regression model, a variation in the model will be seen because of the dynamic nature of the dataset. With these rapid changes, players would be able to see their different patterns of playing and can see themselves growing rapidly. These patterns would make players aware of their strengths and weaknesses, by which they can decide where they need to work on and where they can apply their strengths.

Current model is great for evaluation and can be used for pointers above. It can come into the real world and can really help the players. Considering regression model further, it can help with:

**Different predictions:** This model just offers prediction for Strikeouts, while there are many other variables in this dataset which can be used to see the performance and other baseball statistics. Using this model and different y-variable can help see the other prediction which are very essential in baseball.

**Intuitive evaluation:** By using this model, we can go deeper into statistics of each player. Each player can get their own model with different variable and see their performance in different areas. It will give the player a very great perspective of his/her game and performance can use the opportunity to look back at the techniques they used to perform well in the leagues.

All in all, this model can really help with the predictions and evaluations, it can be used in a variety of forms to fulfill various motives.

References:

1. Augustyn, A. (2019, October 31). Major League Baseball. Encyclopedia Britannica.

Retrieved November 22, 2019, from https://www.britannica.com/topic/Major-League-Baseball.

1. Amoros, R. (2016, July 1). Top Professional Sports Leagues by Revenue. Retrieved

November 22, 2019, from https://howmuch.net/articles/sports-leagues-by-revenue.

1. Riess, S. (2017, February 27). Professional Team Sports in the United States. Oxford

Research Encyclopedia of American History. Retrieved 22 Nov. 2019, from https://oxfordre.com/americanhistory/view/10.1093/acrefore/9780199329175.001.0001/acrefore-9780199329175-e-198.

1. Ablin, J. (2019, July 9). The 2019 MLB Moneyball Report. Retrieved November 22,

2019, from https://www.worth.com/the-2019-mlb-moneyball-report/.

1. Spurr, S., & Barber, W. (1994). The Effect of Performance on a Worker's Career:

Evidence from Minor League Baseball. *Industrial and Labor Relations Review,* *47*(4), 692-708. doi:10.2307/2524667

1. Lewis, M. (2013). Moneyball: the art of winning an unfair game. New York: W.W.

Norton.

1. Latson, J. (2019, October 29). How the Astros play Moneyball. Retrieved November 22,

2019, from https://www.houstonchronicle.com/local/gray-matters/article/How-the-Astros-play-Moneyball-14569202.php.

1. Sparks, R., & Abrahamson, D. (2005). A Mathematical Model to Predict Award Winners.

Math Horizons, 12(4), 5-13. Retrieved from [www.jstor.org/stable/25678541](http://www.jstor.org/stable/25678541)

1. FANGRAPHS. (2019). Retrieved November 4, 2019, from

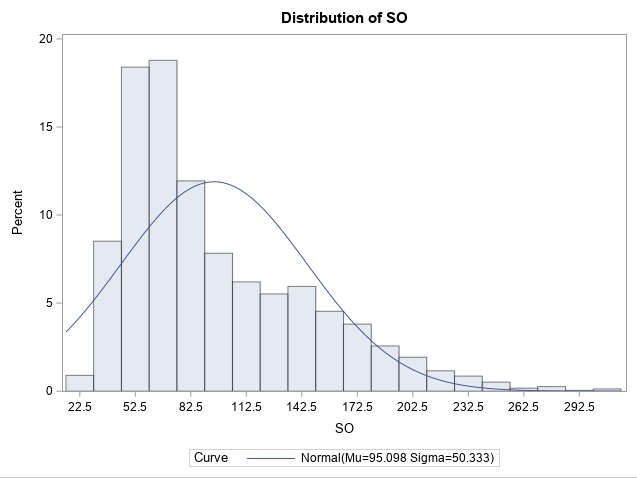
<http://www.fangraphs.com/leaders.aspx?pos=all&stats=pit&lg=all&qual=50&type=c,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,-1&season=2018&month=0&season1=2012&ind=1&team=&rost=&age=&filter=&players=&page=1_100000>.

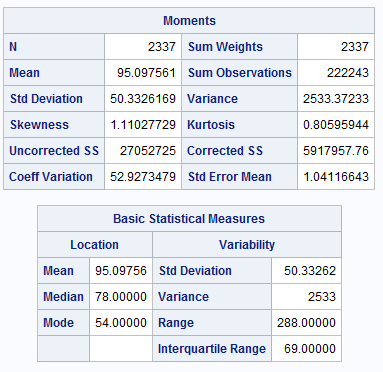
1. Allison, P. (2012, September 10). When Can You Safely Ignore Multicollinearity?

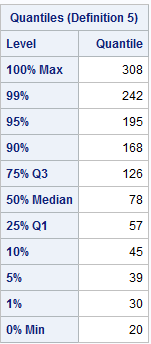
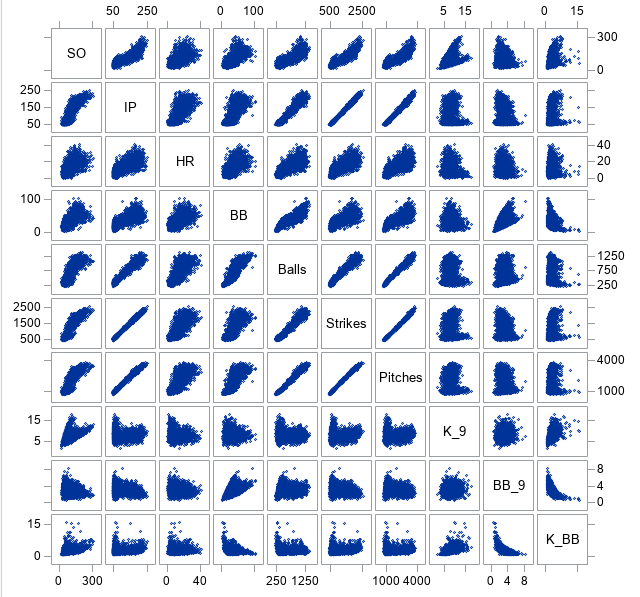
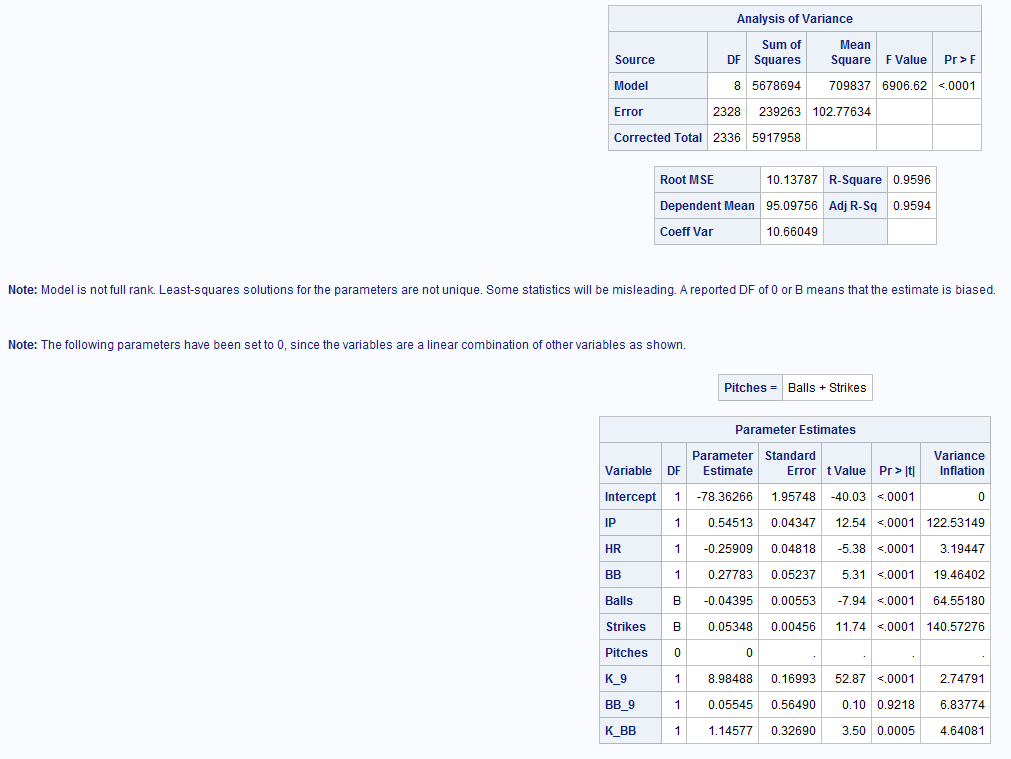
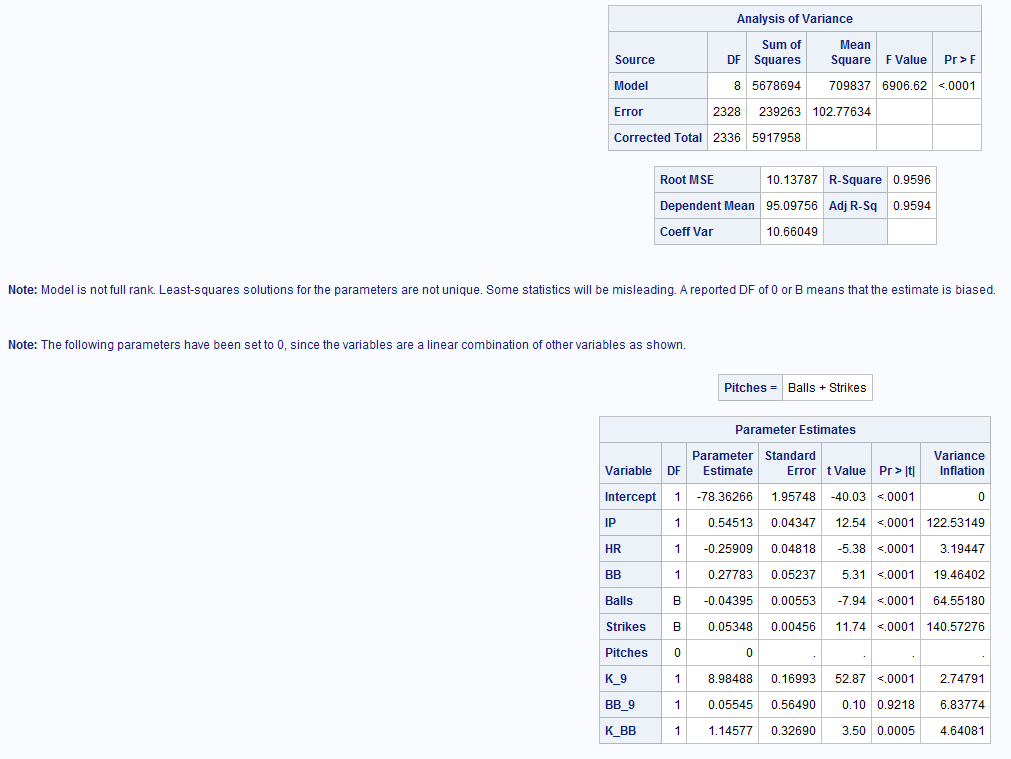
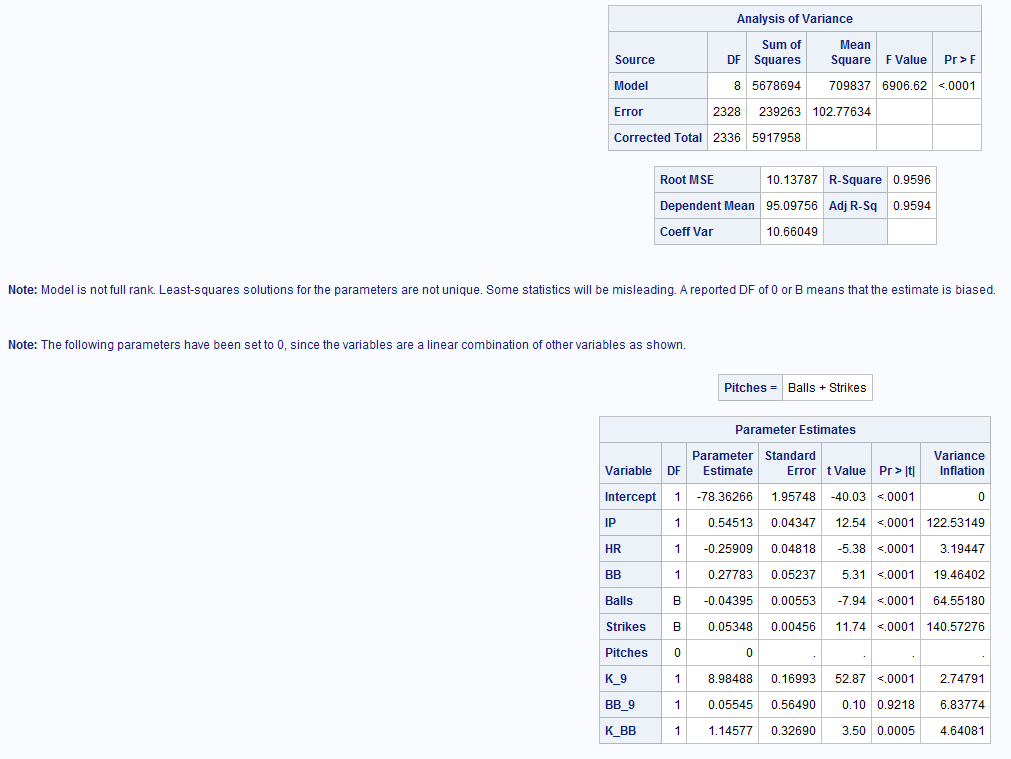
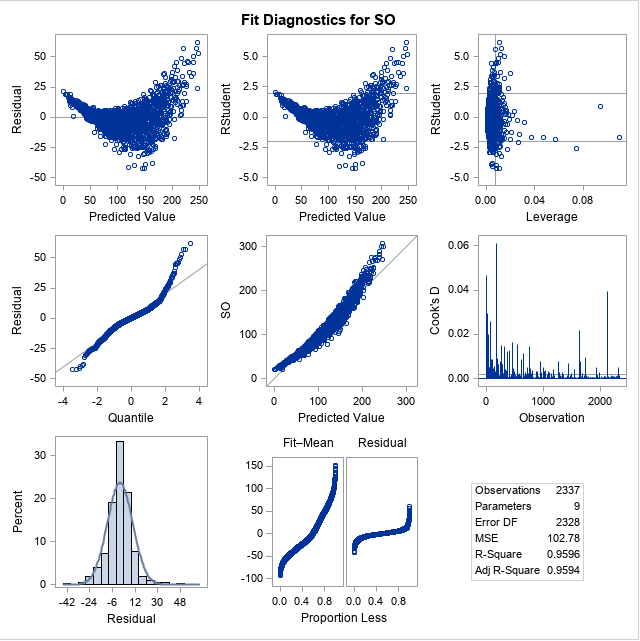
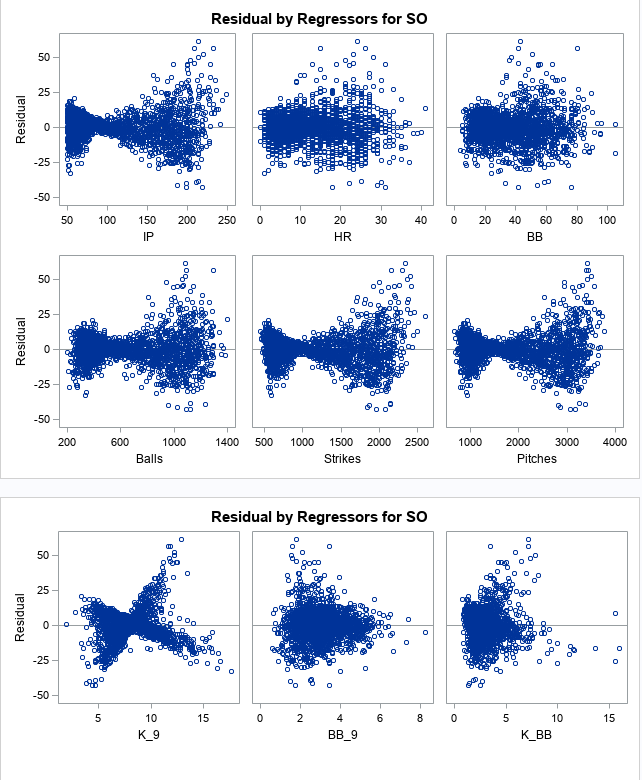
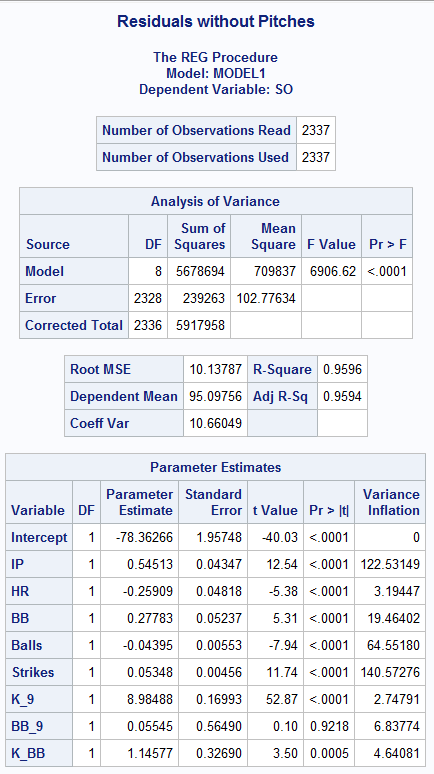
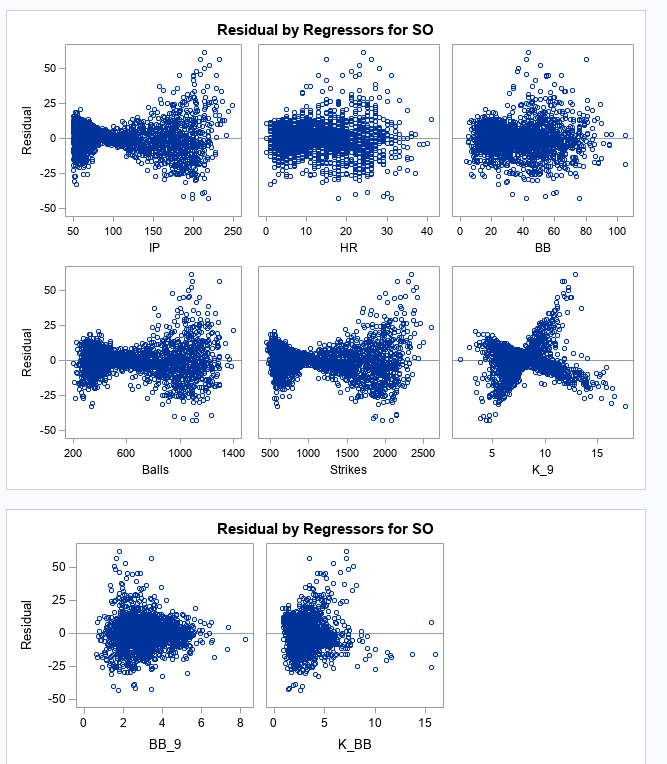
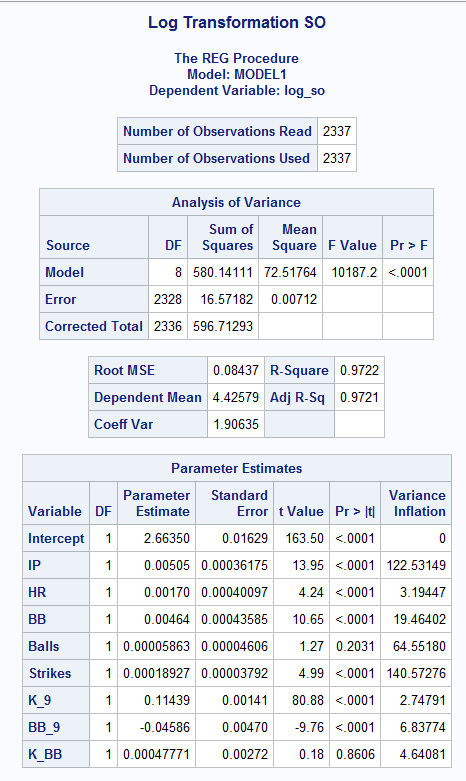
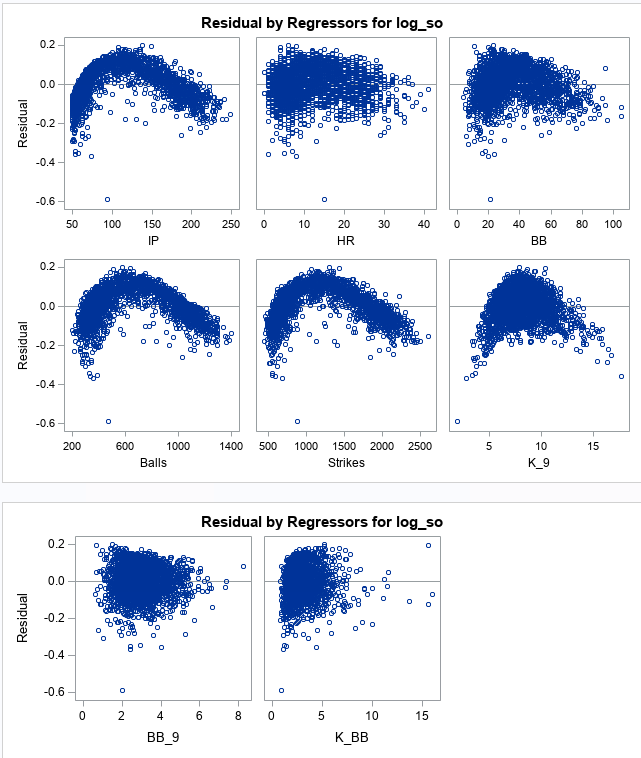
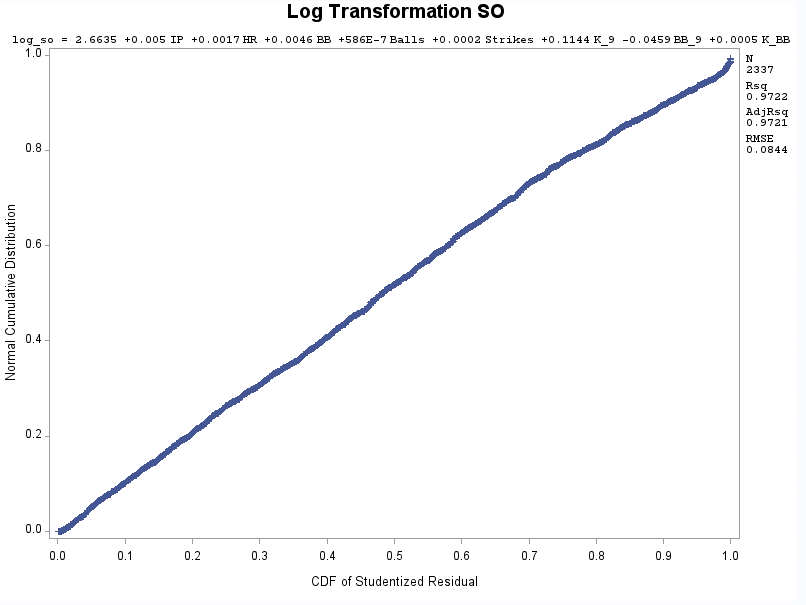
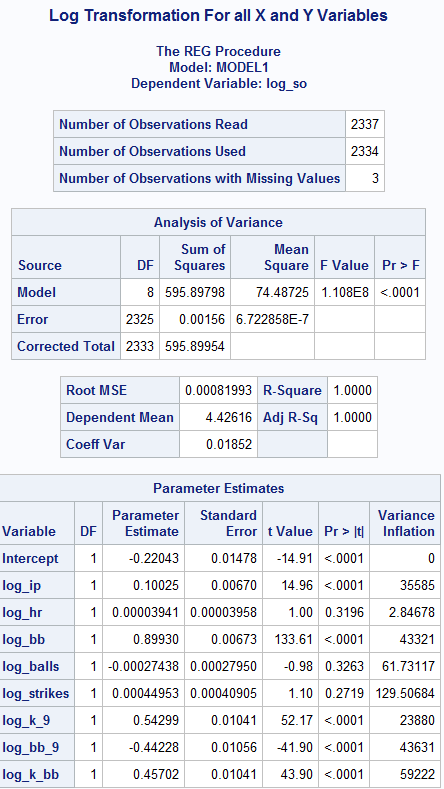
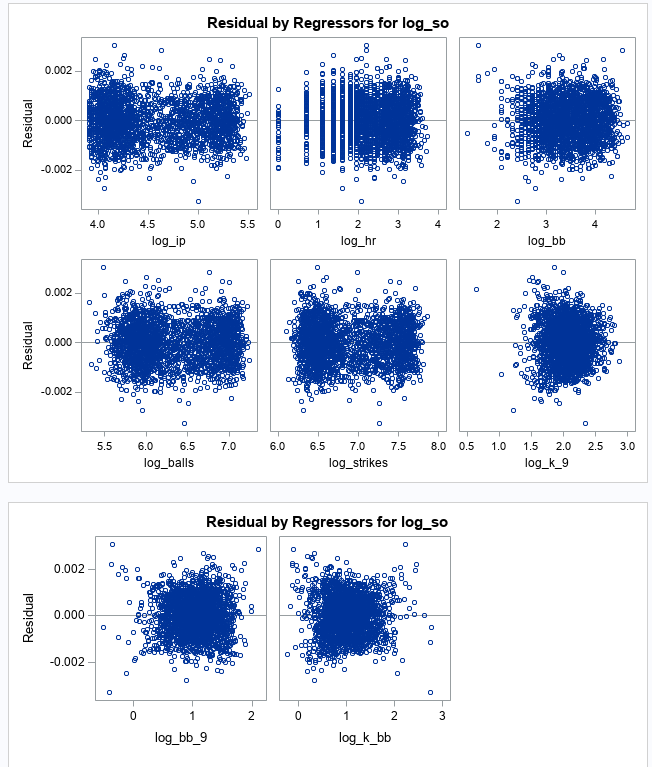
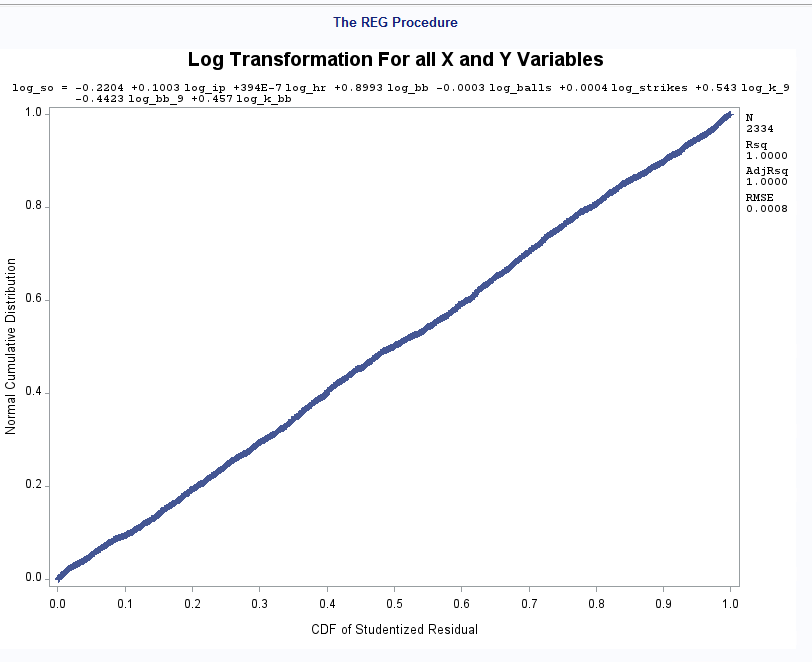
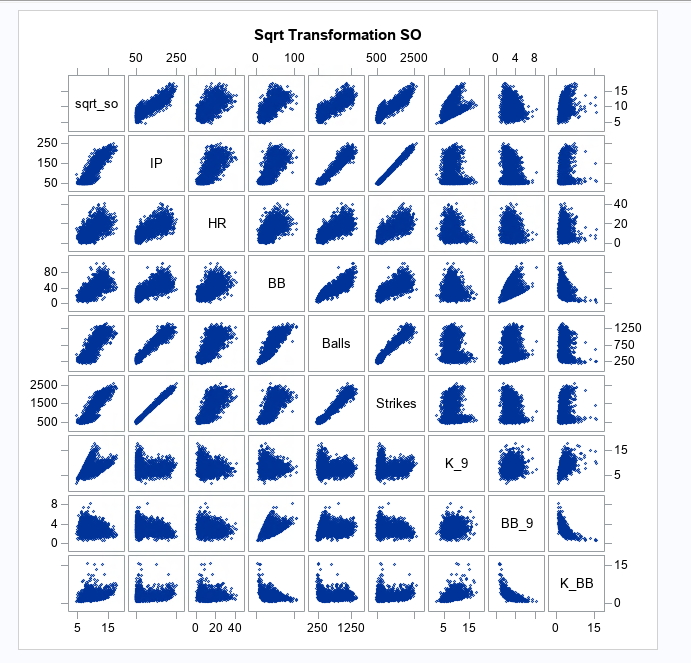
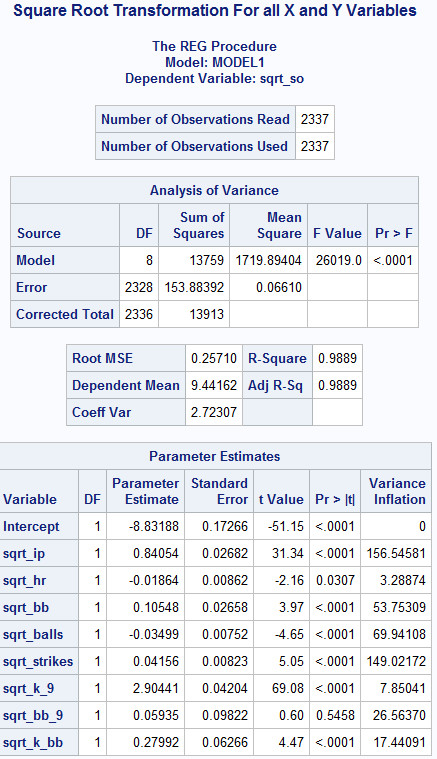
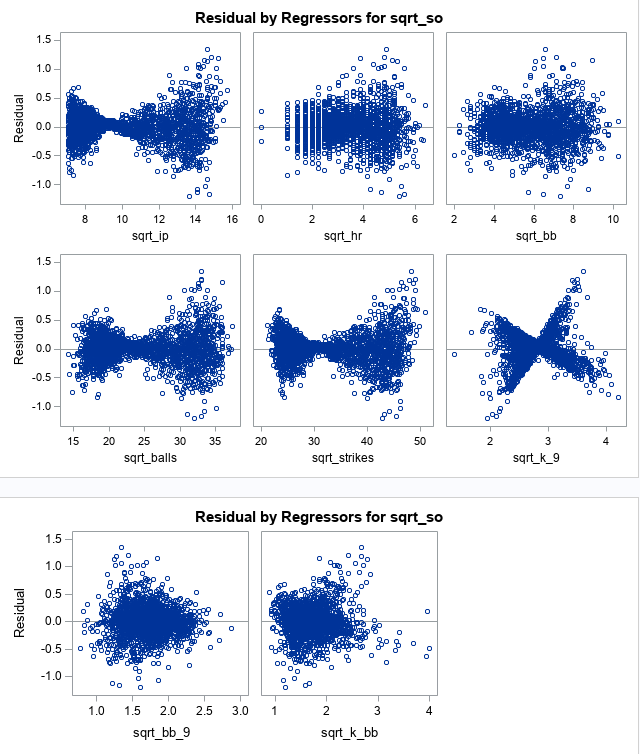
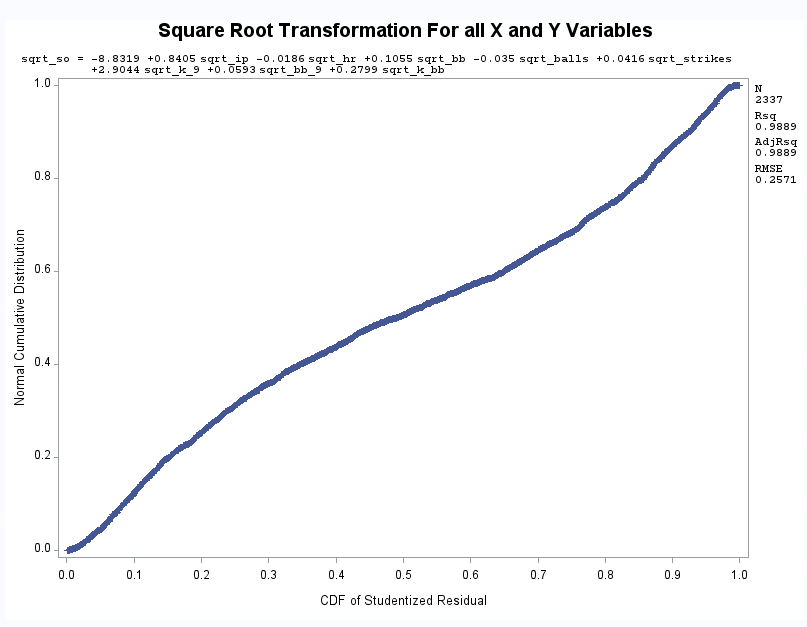
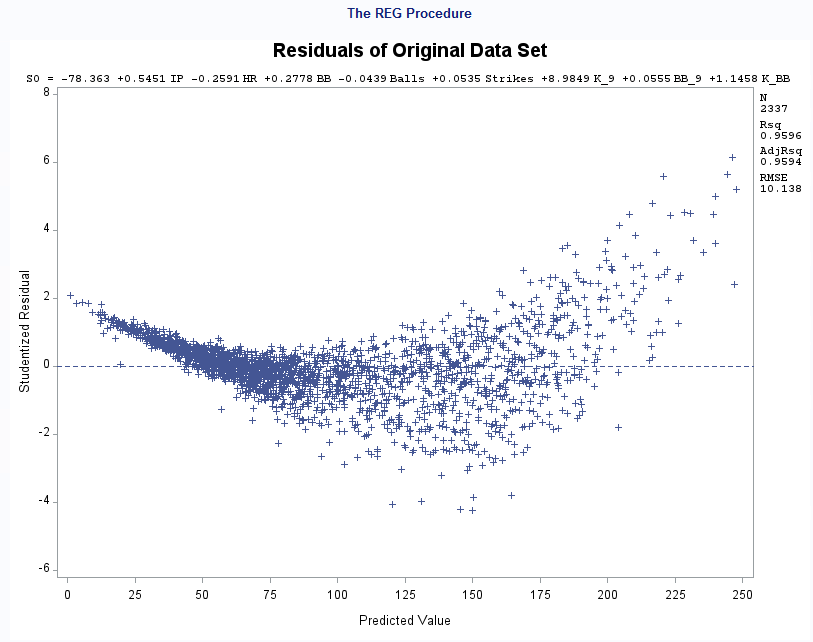
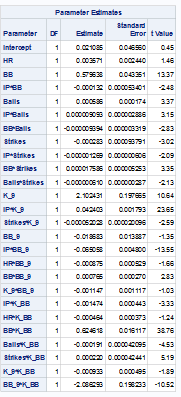
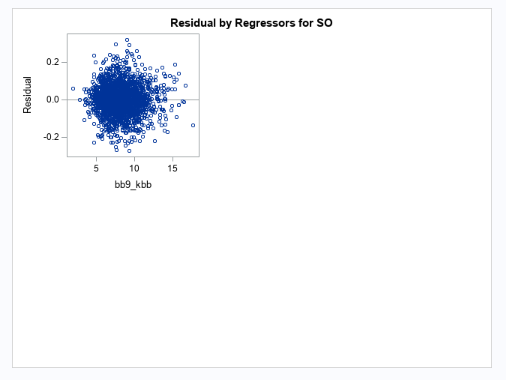
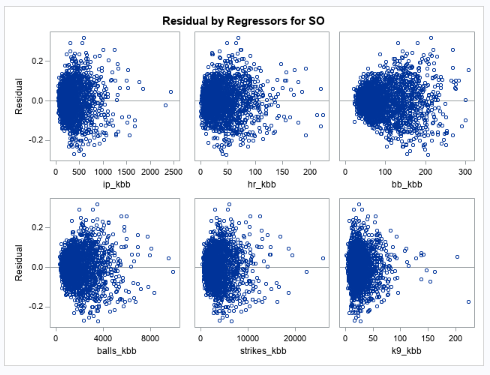
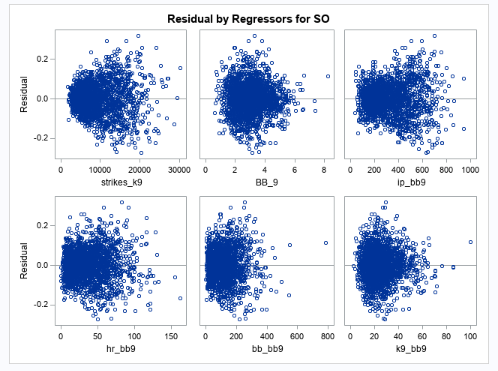
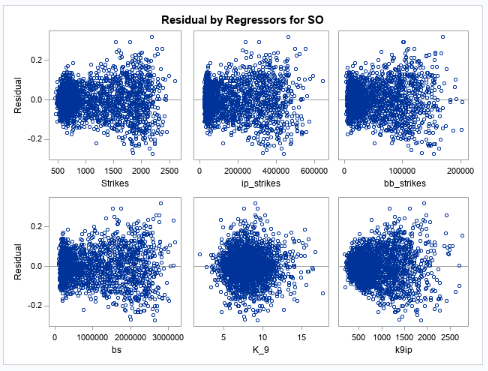
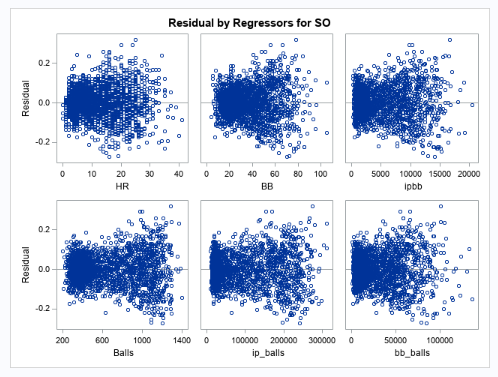
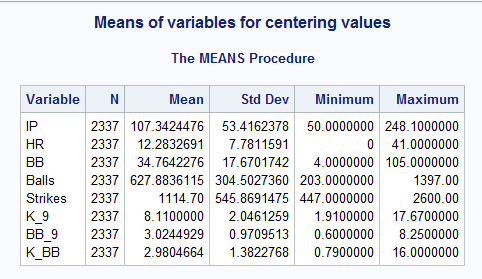
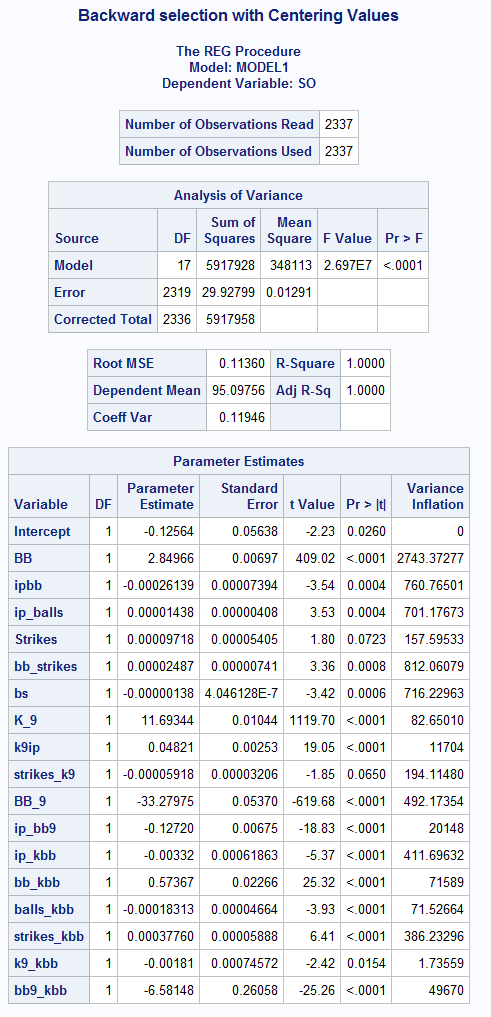
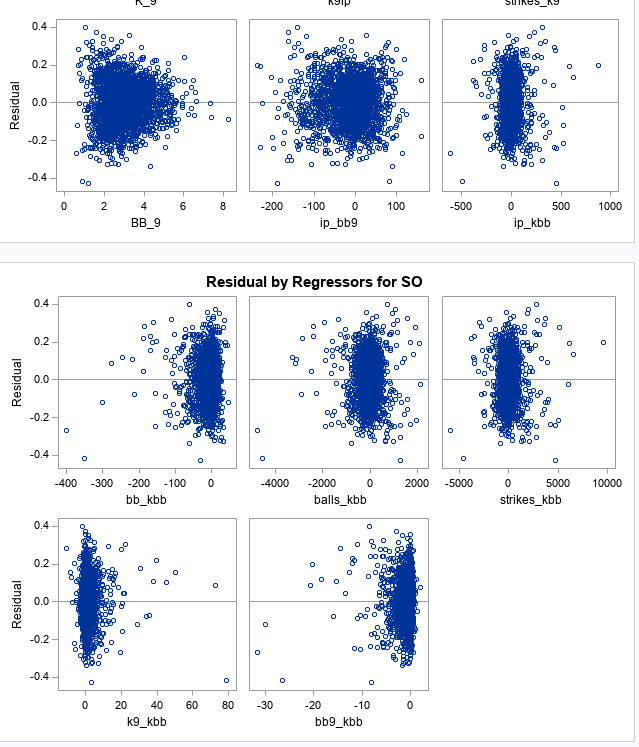
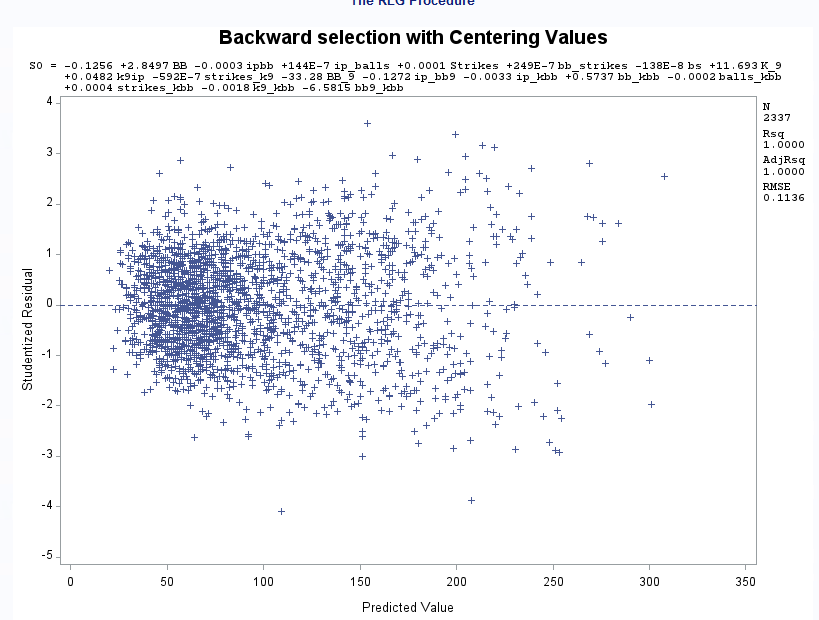
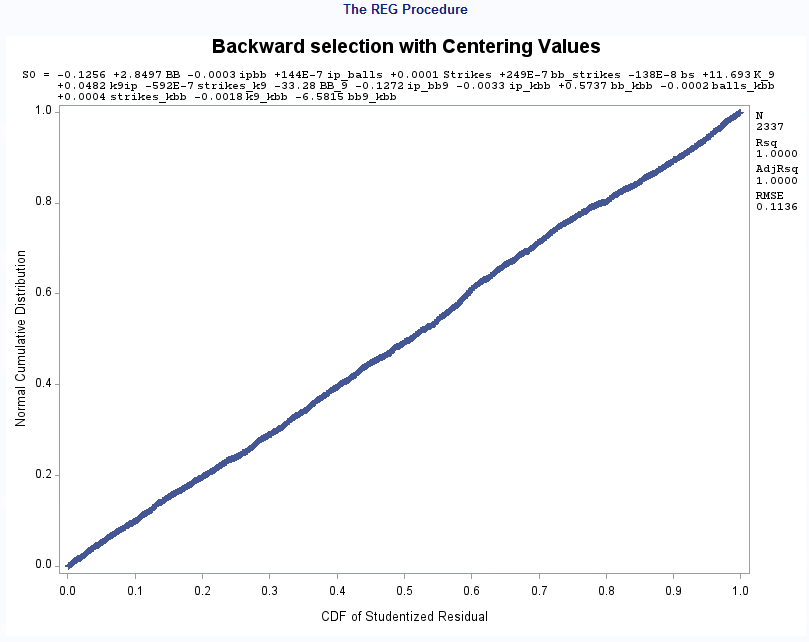
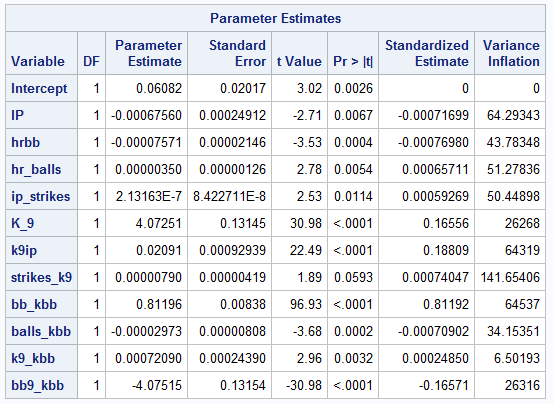
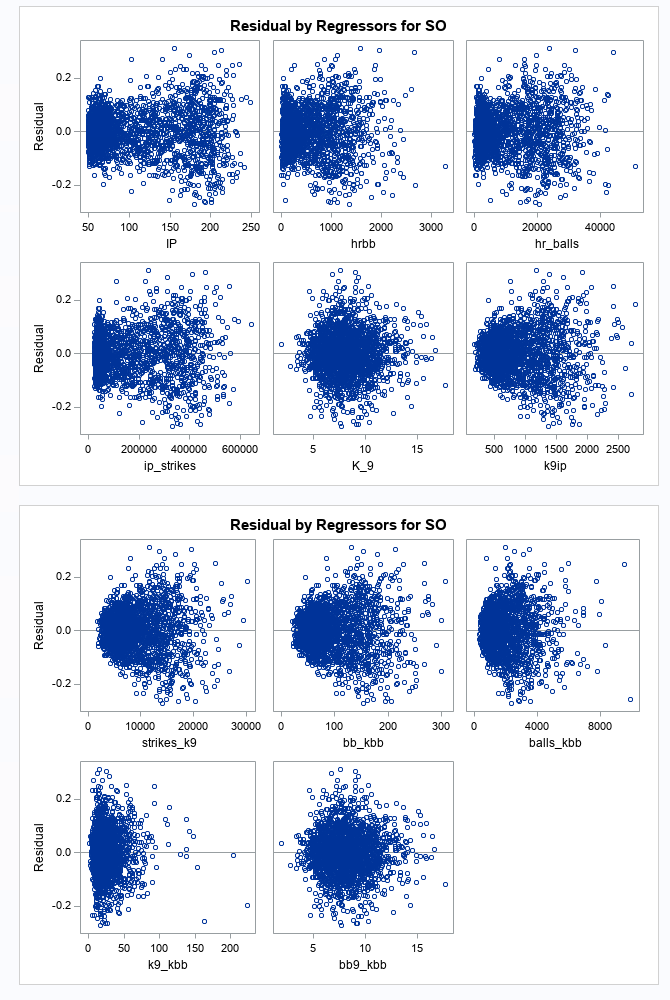
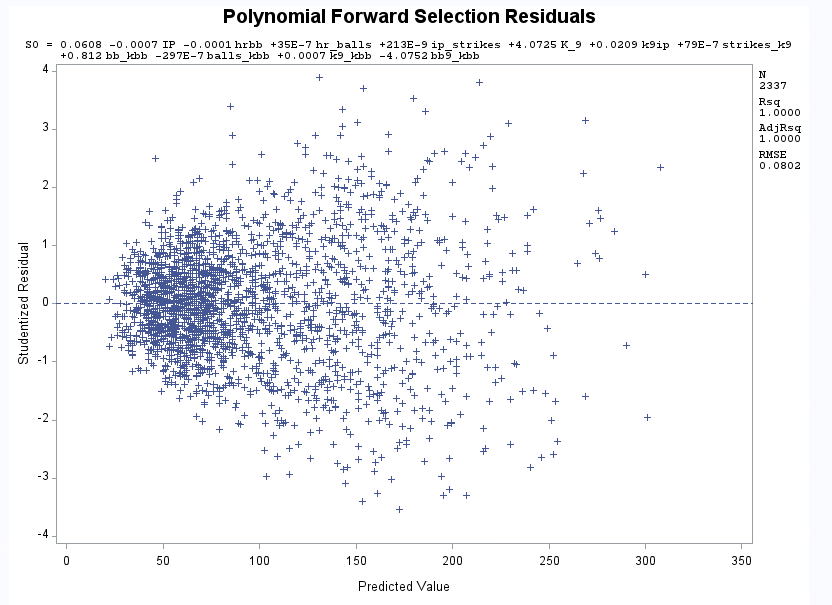
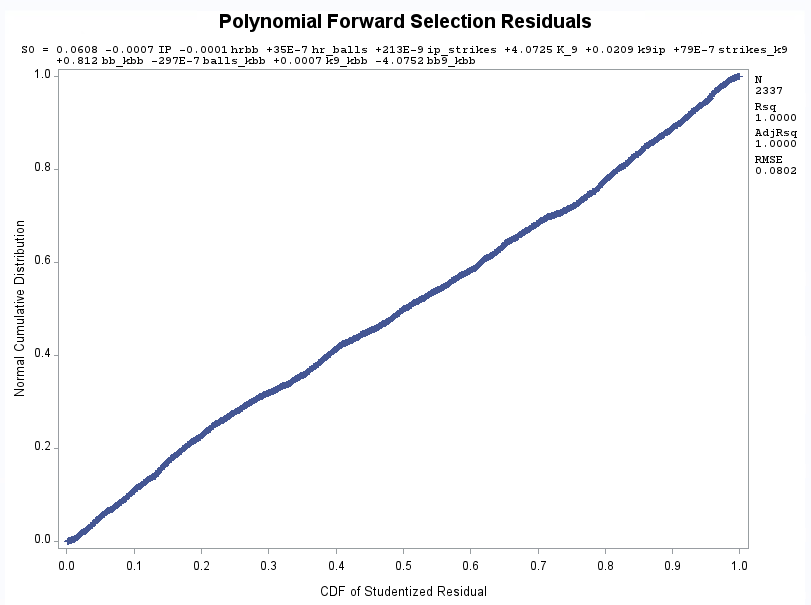
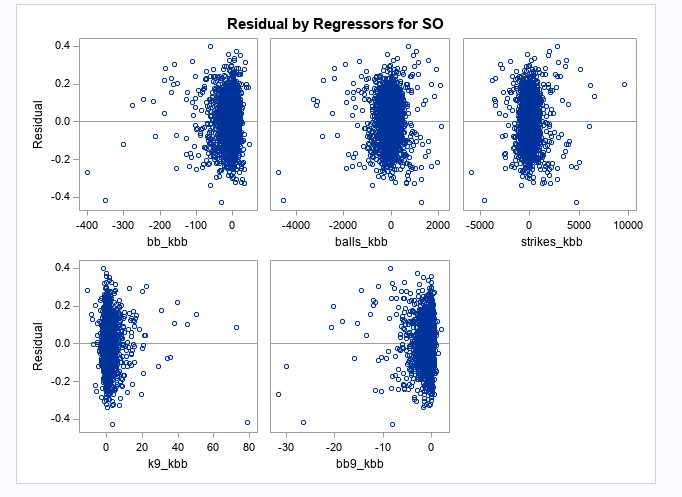
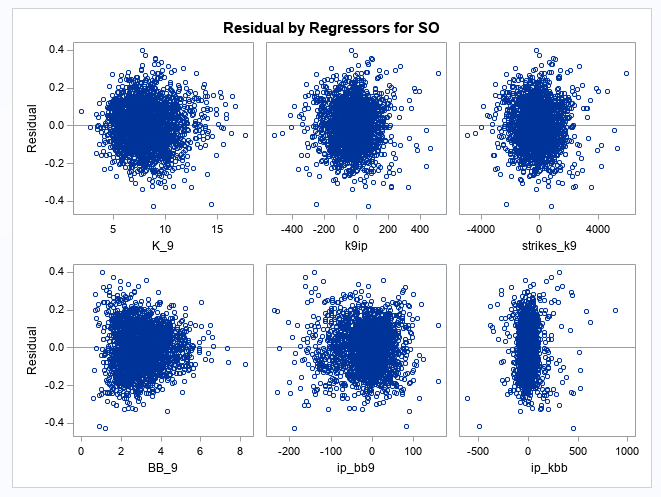
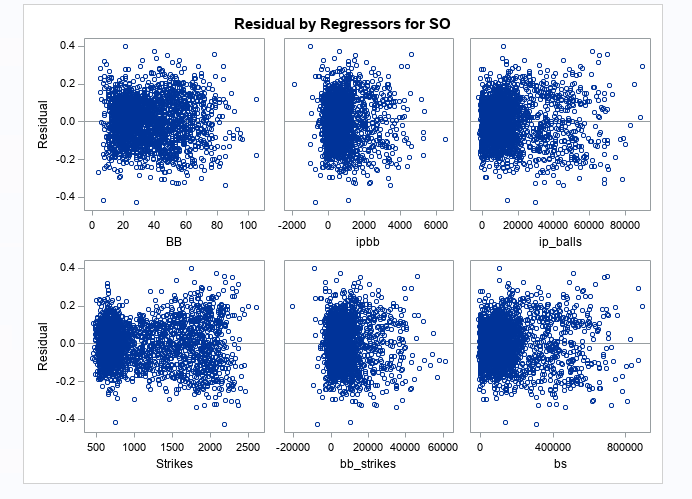
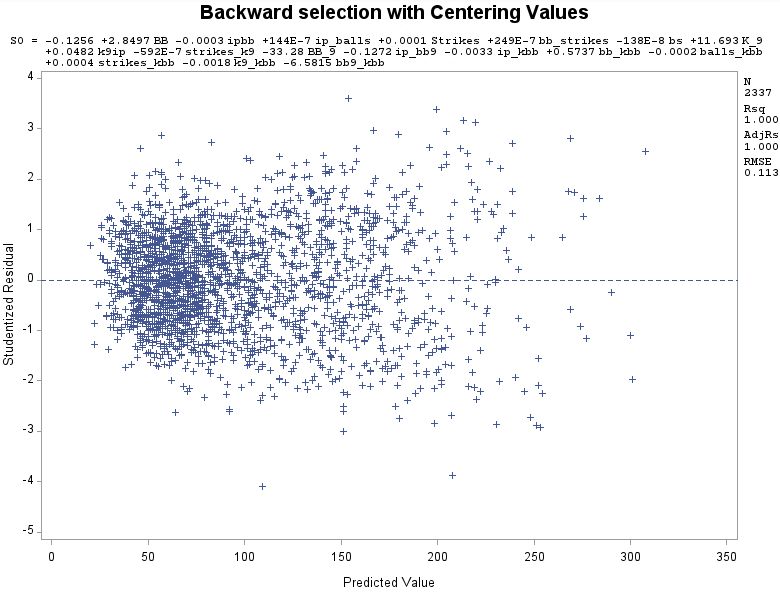
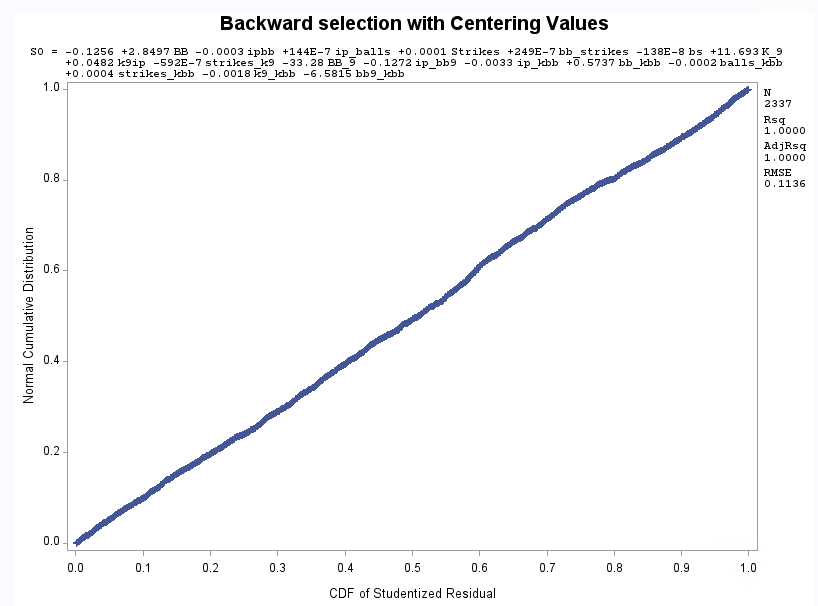
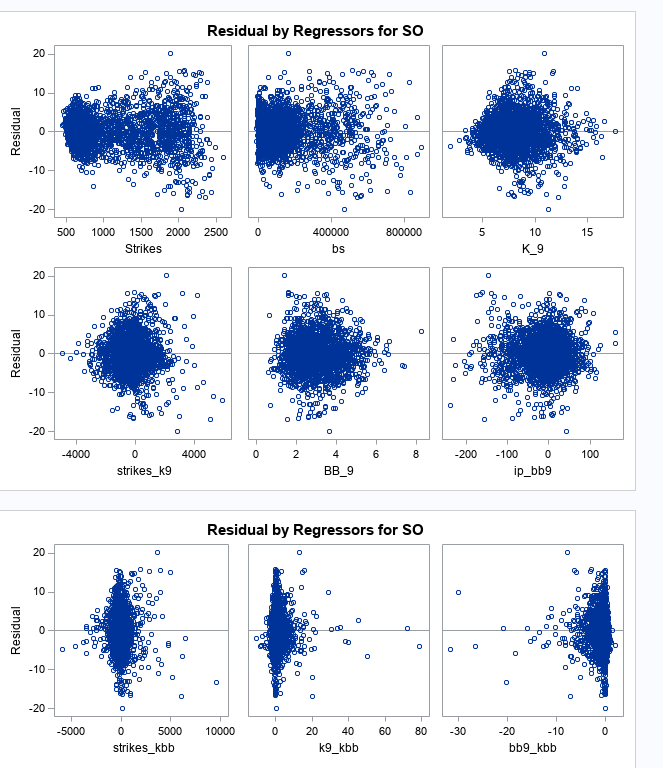
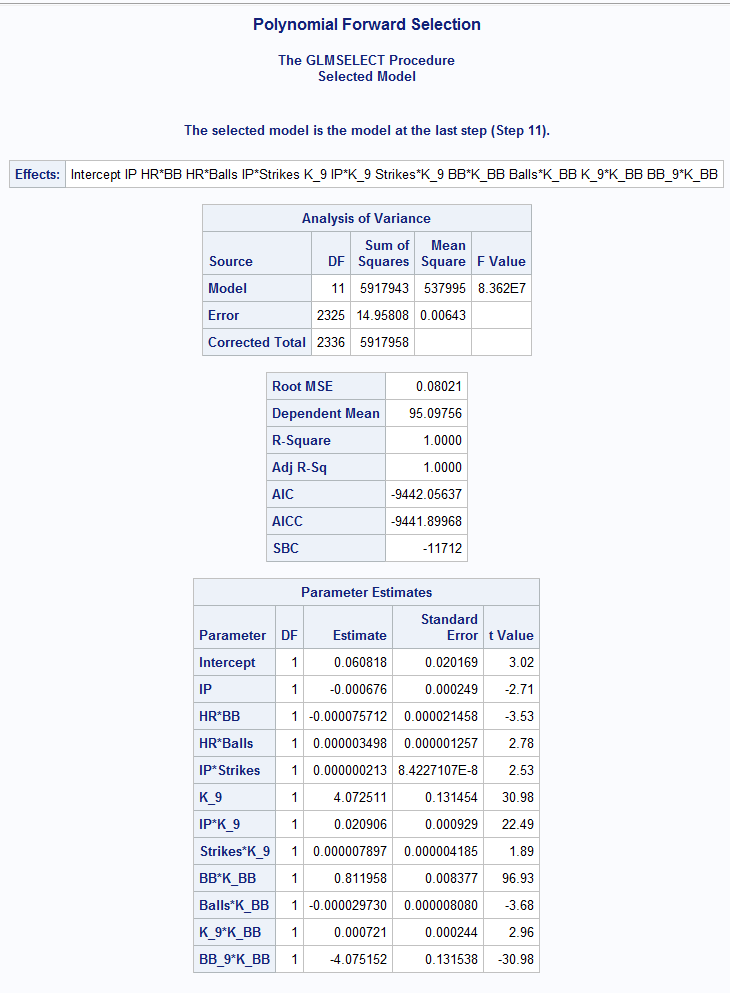
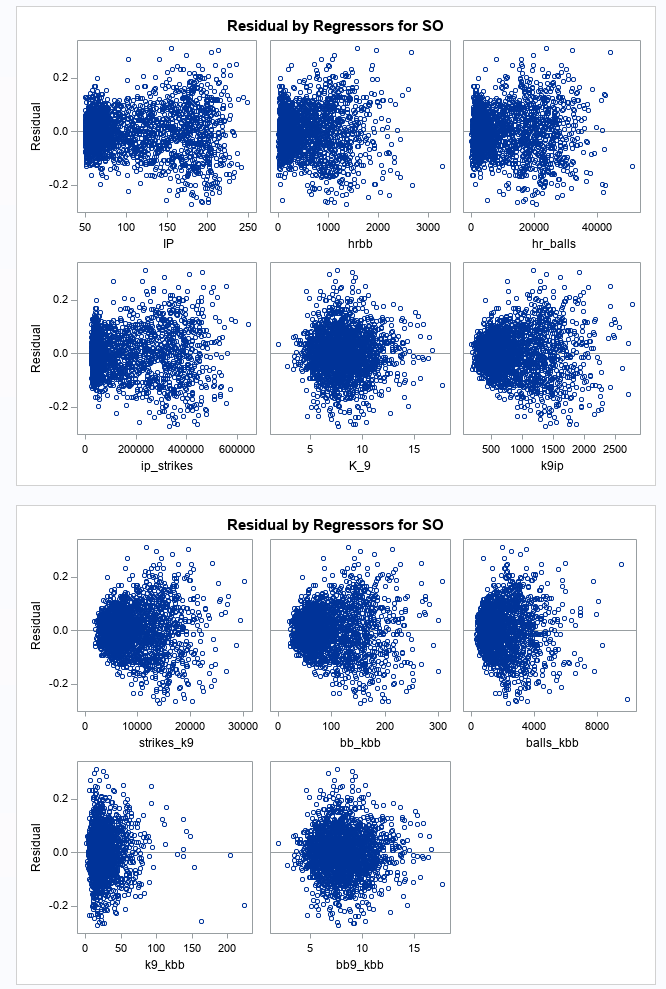
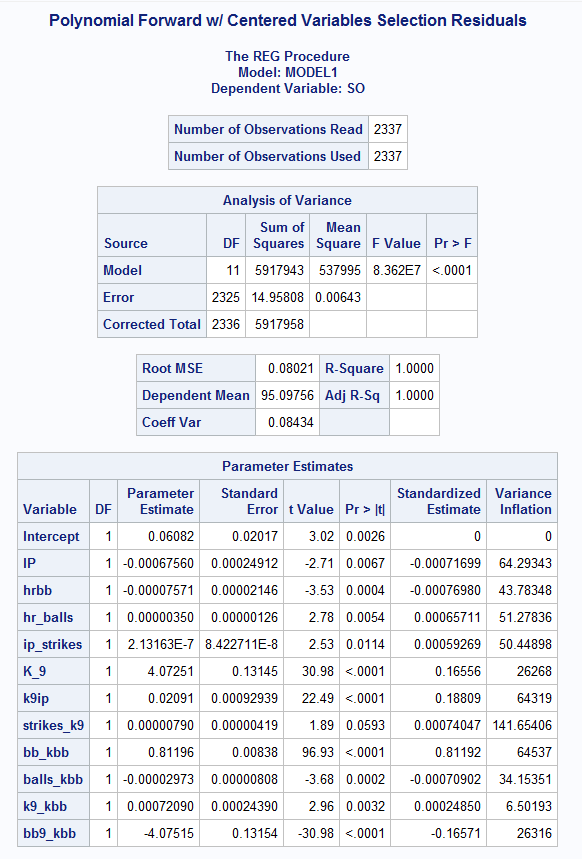
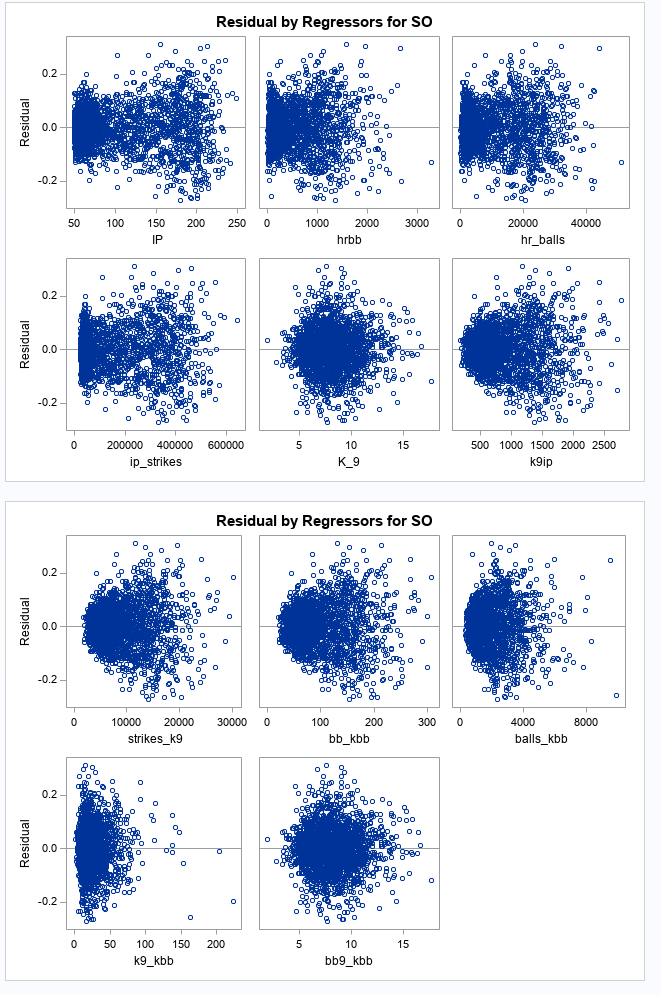
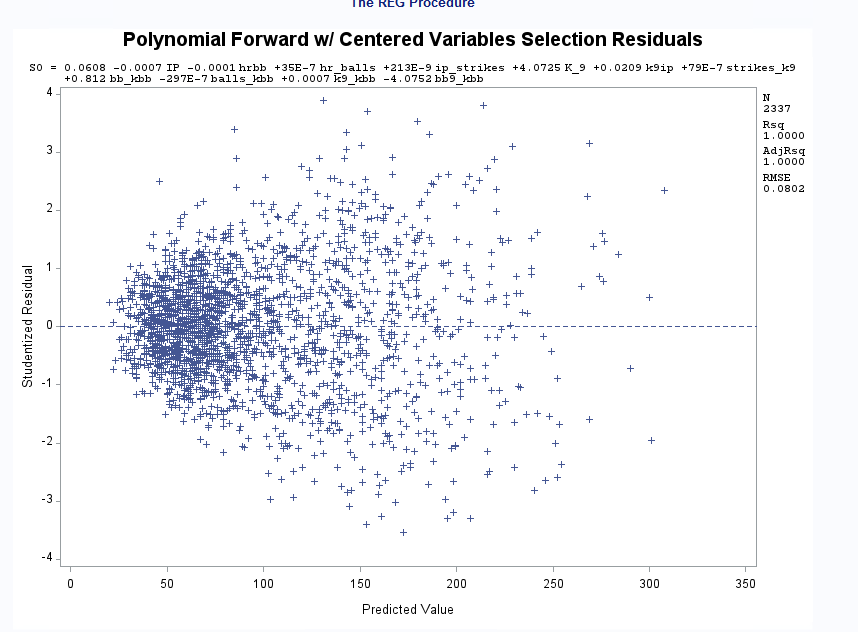
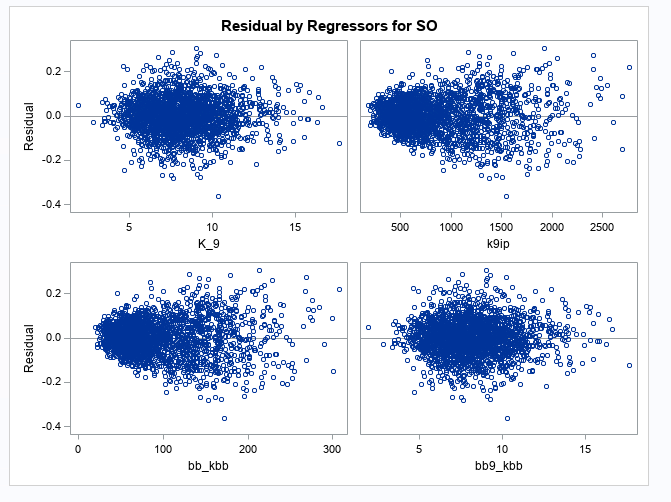
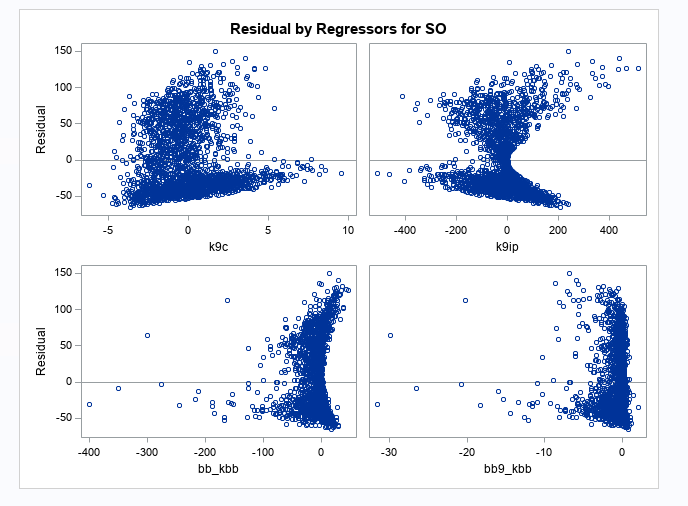
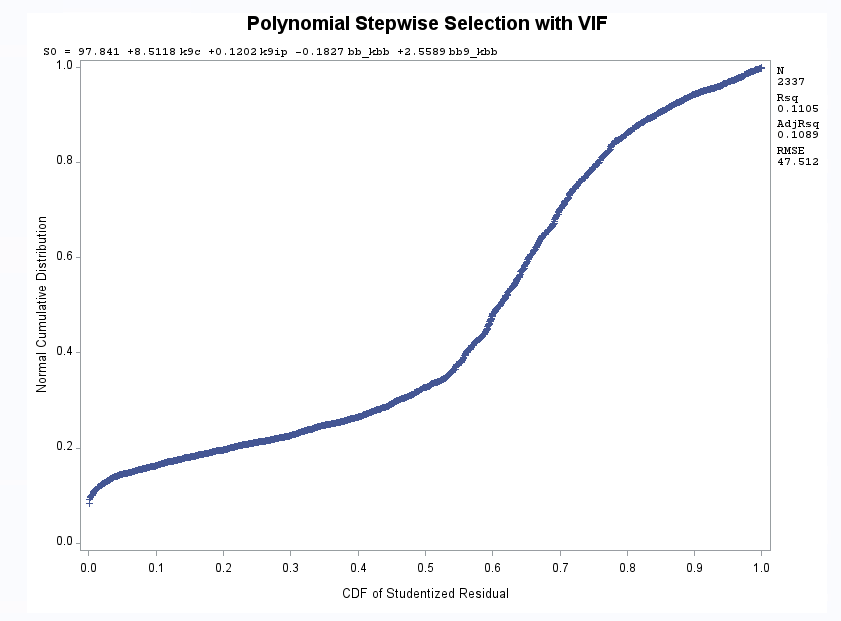
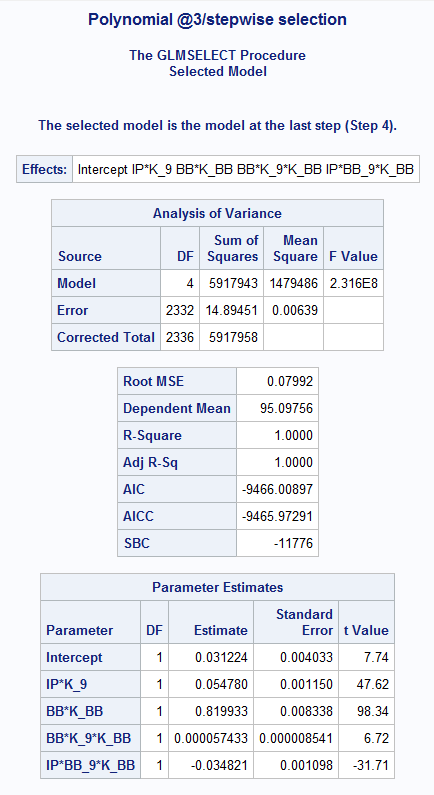
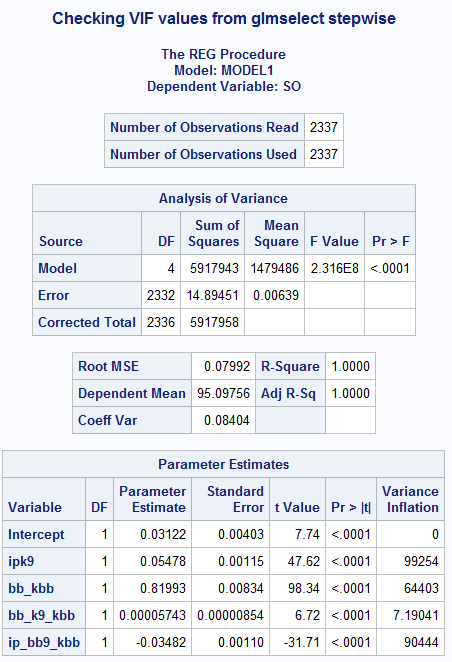
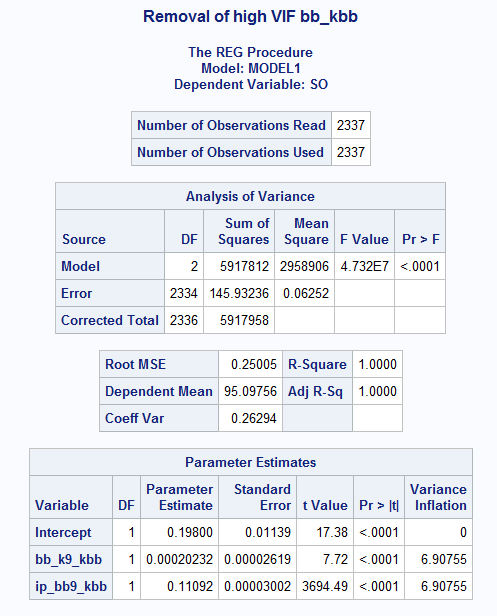
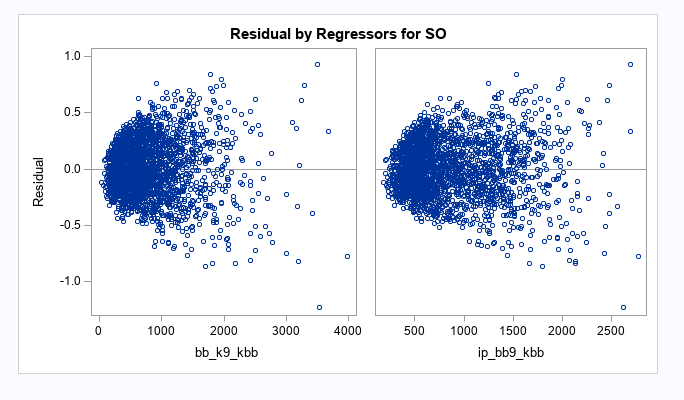
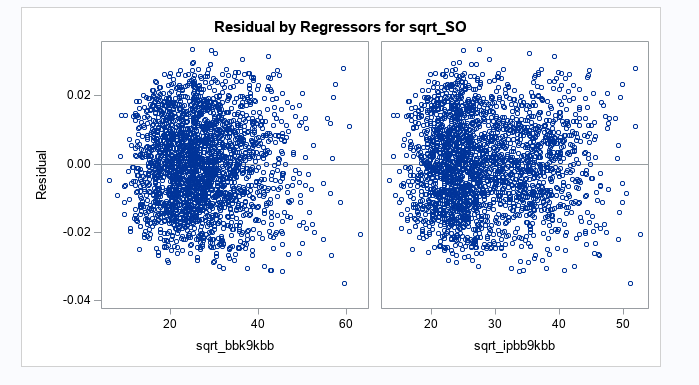
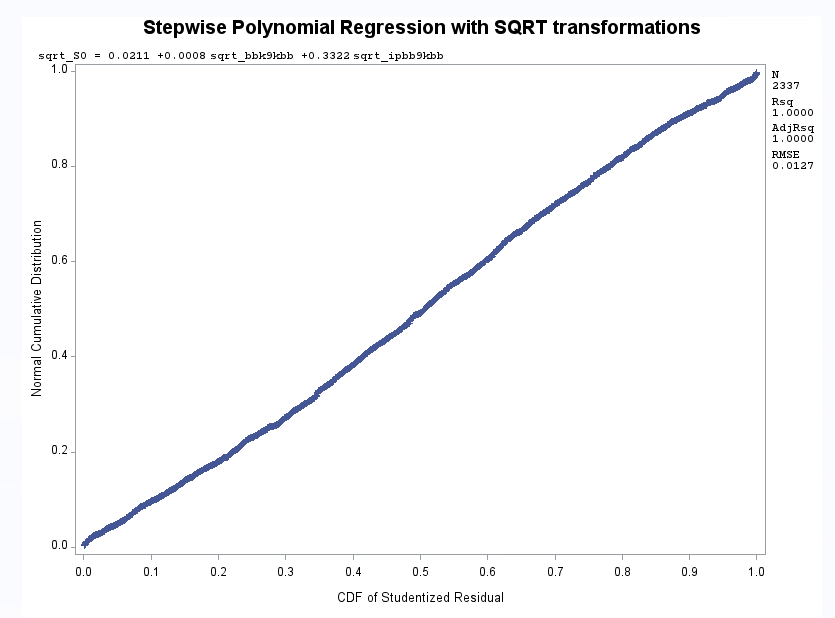
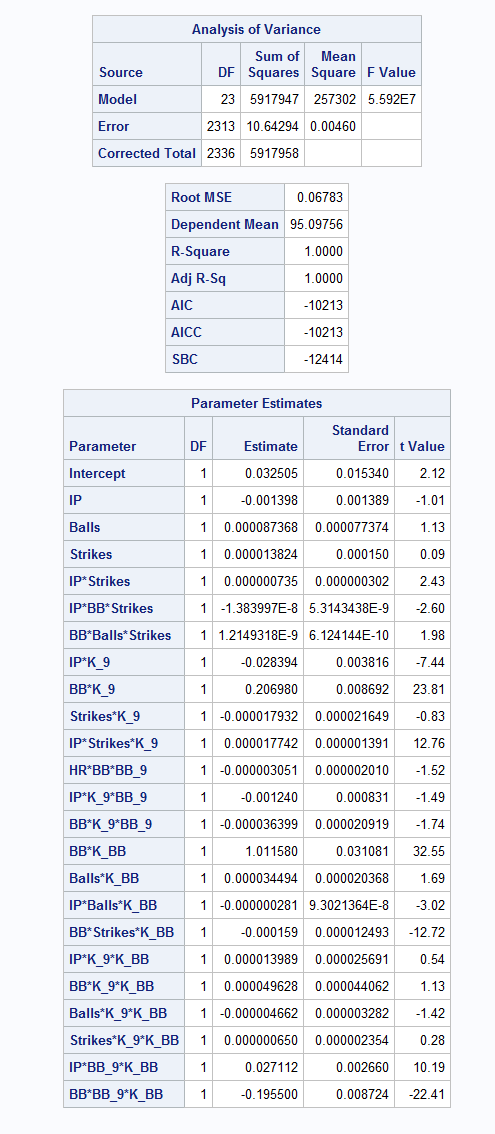
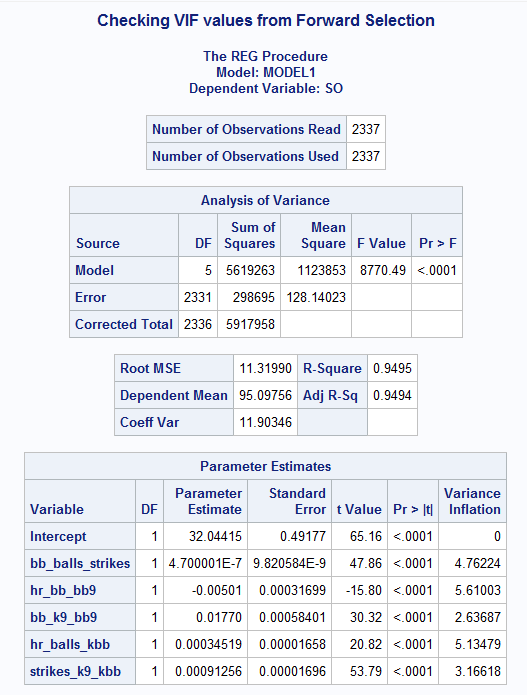
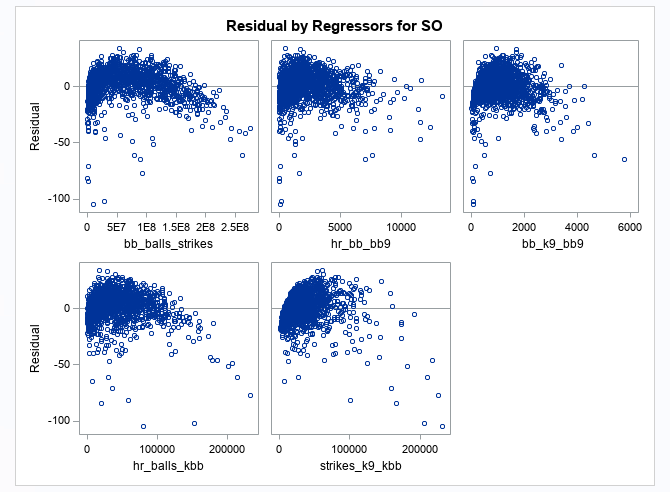
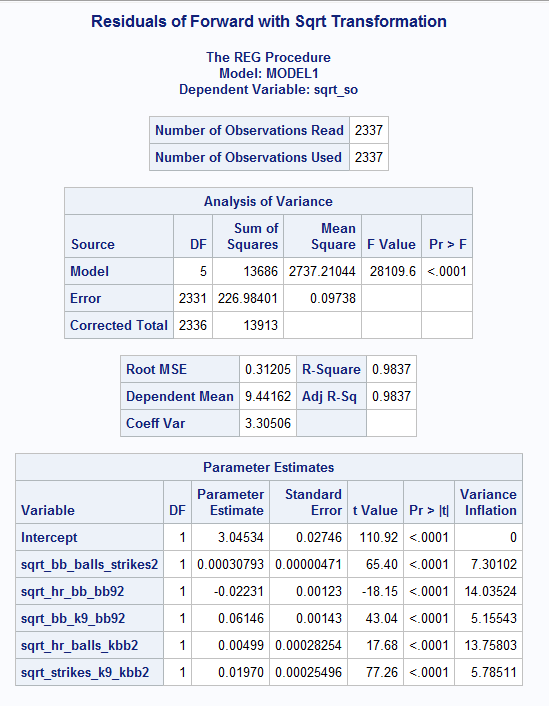
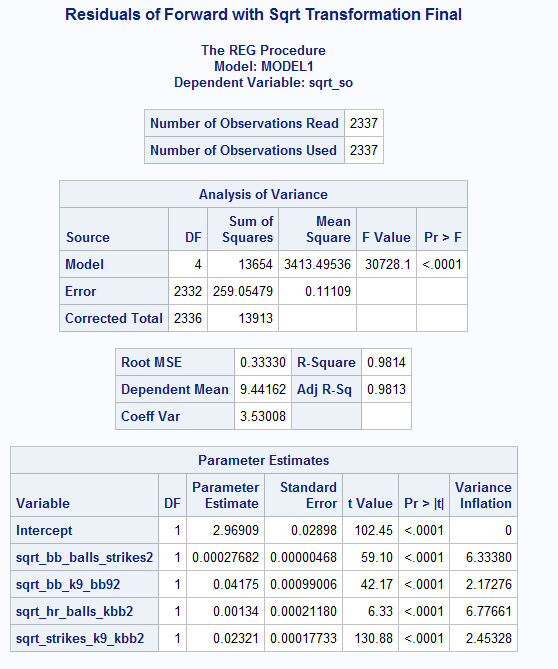
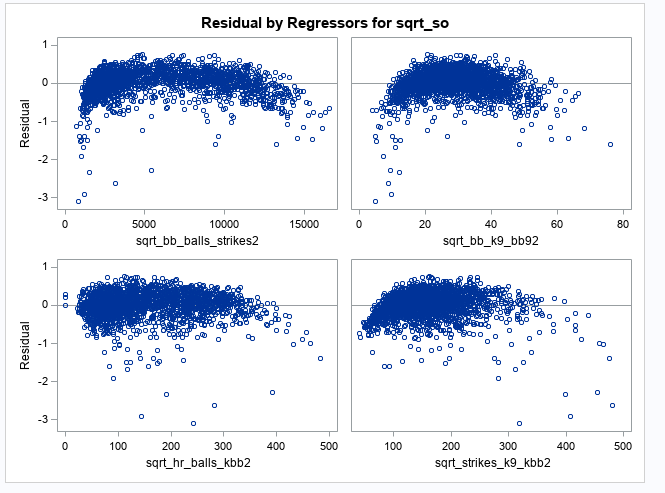
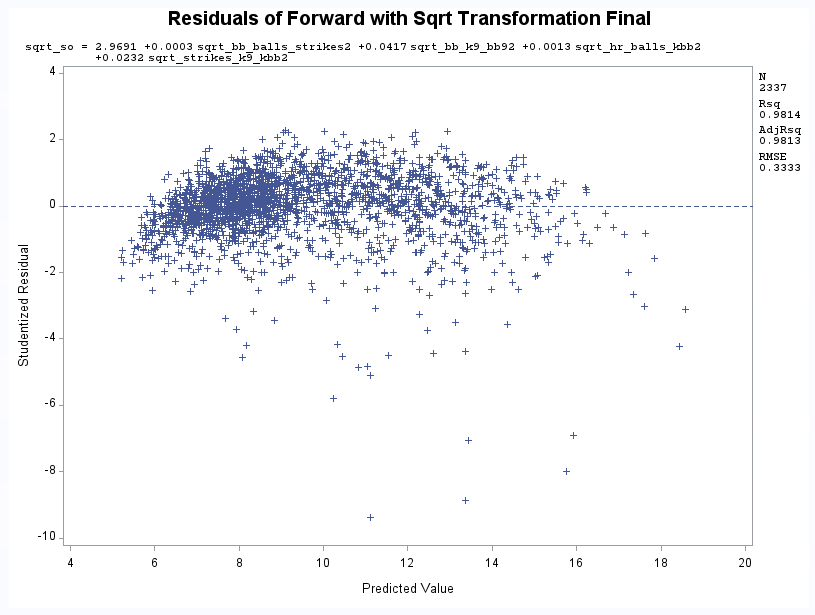
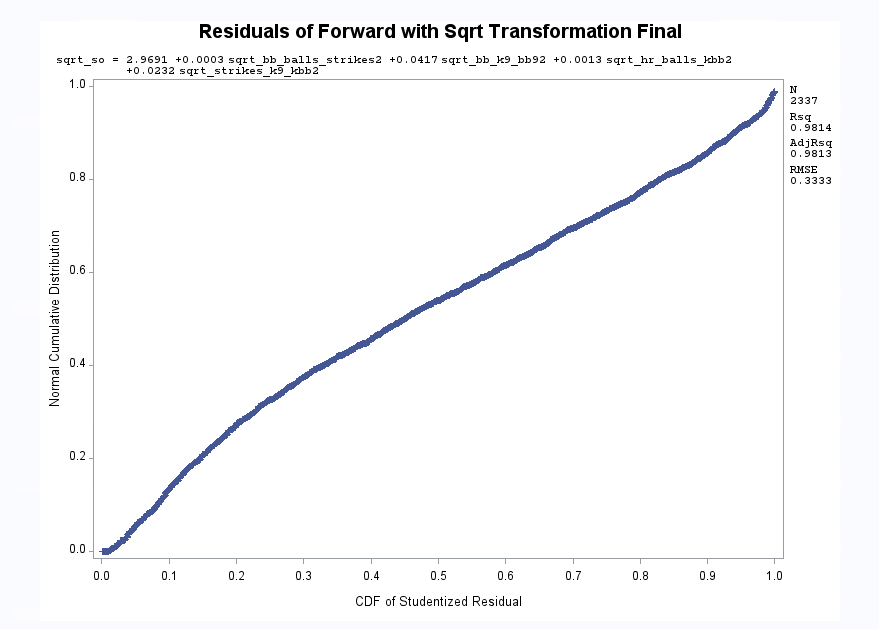
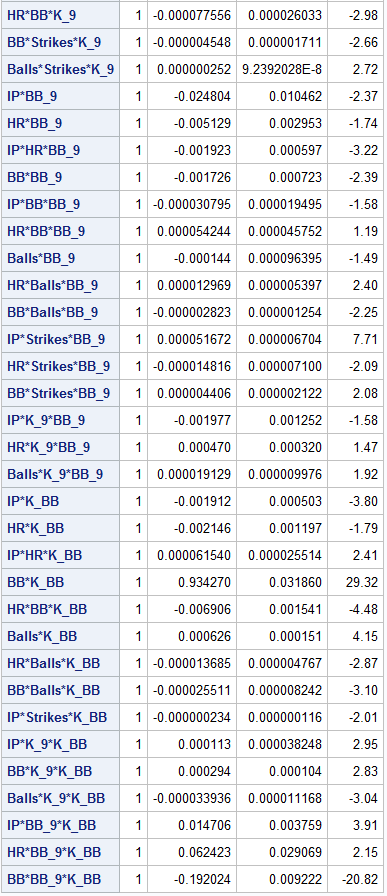
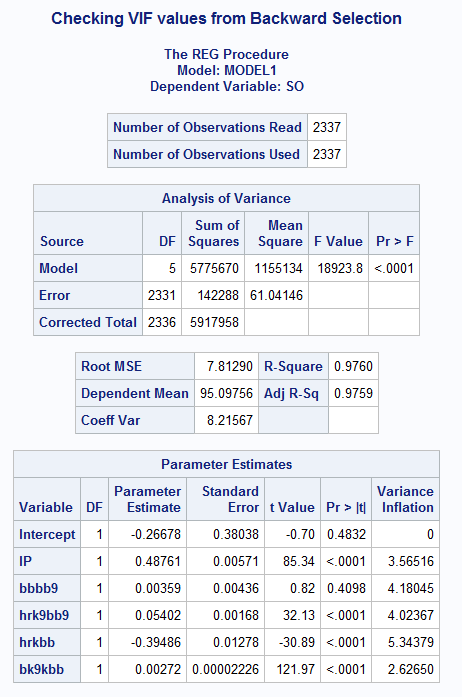
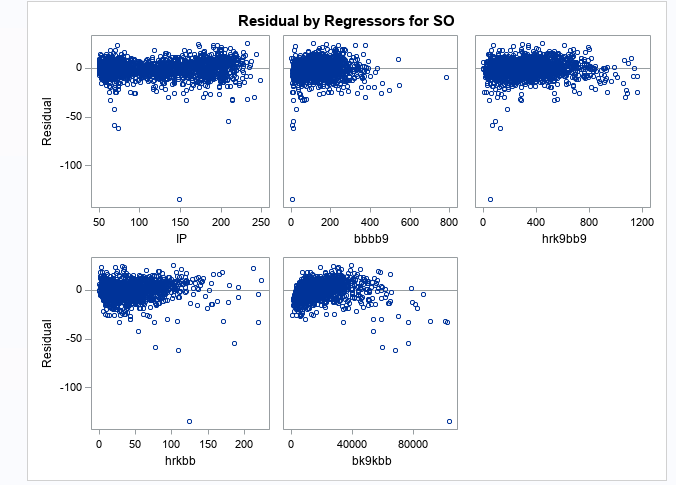
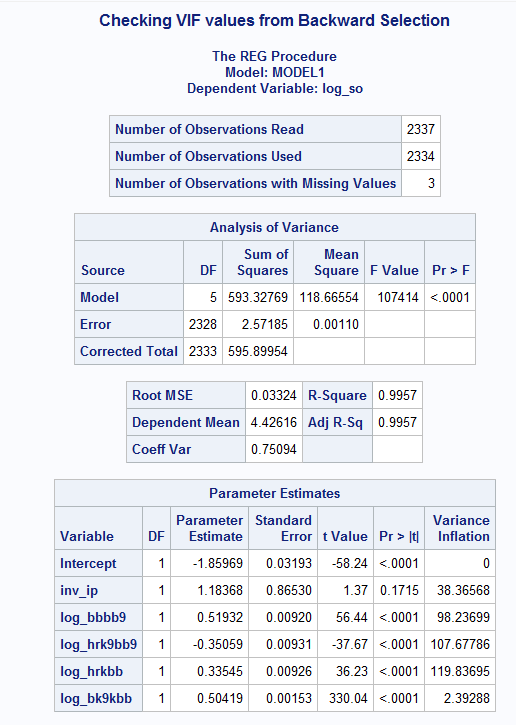
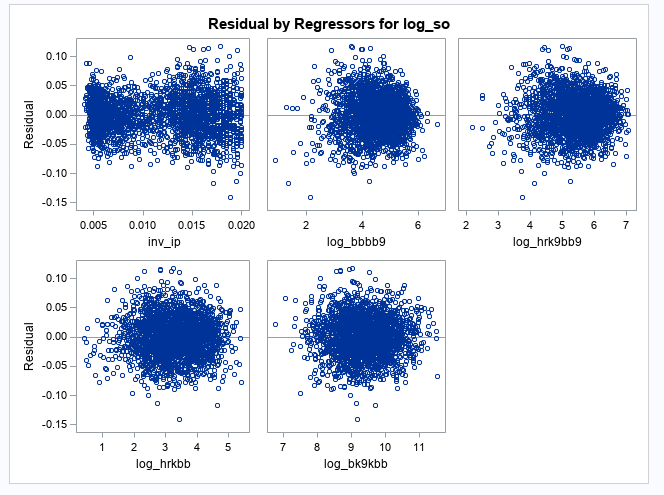
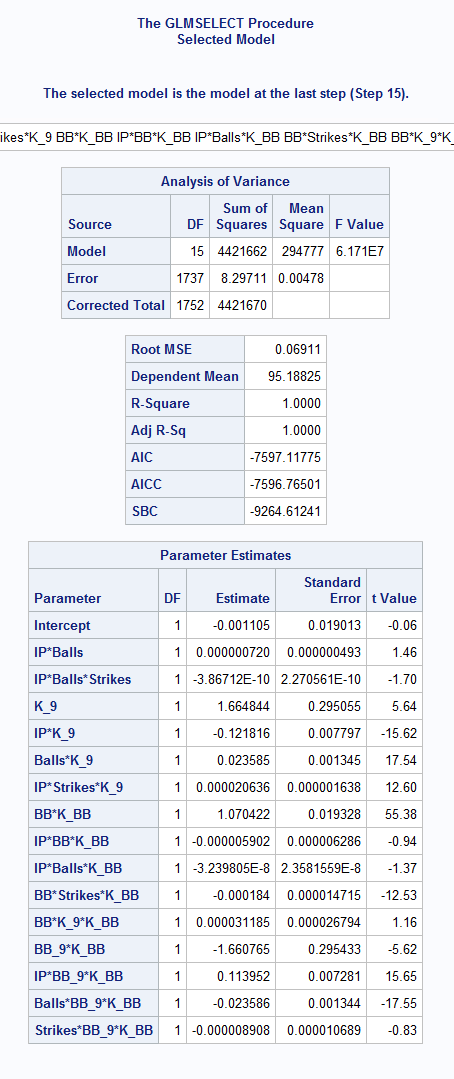
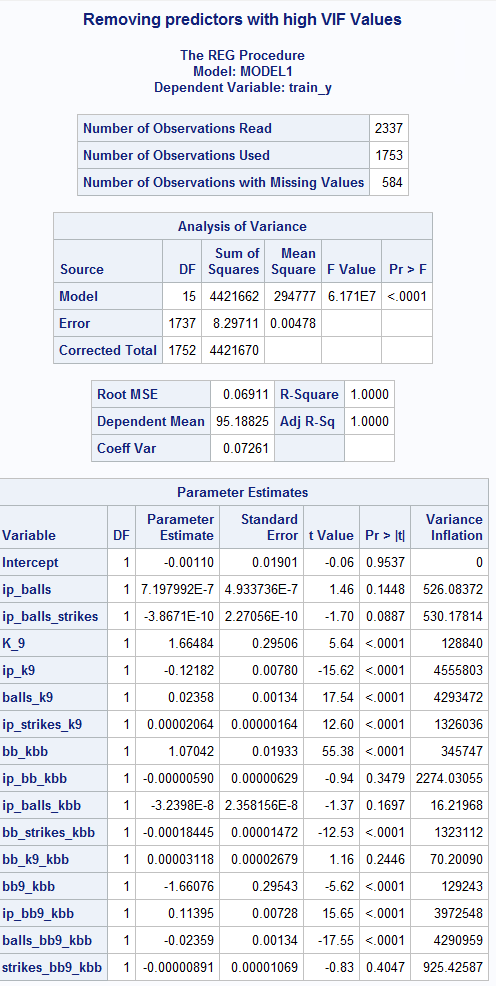
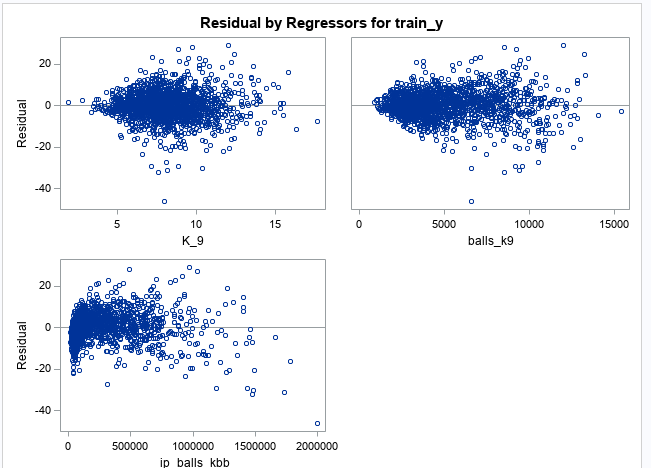
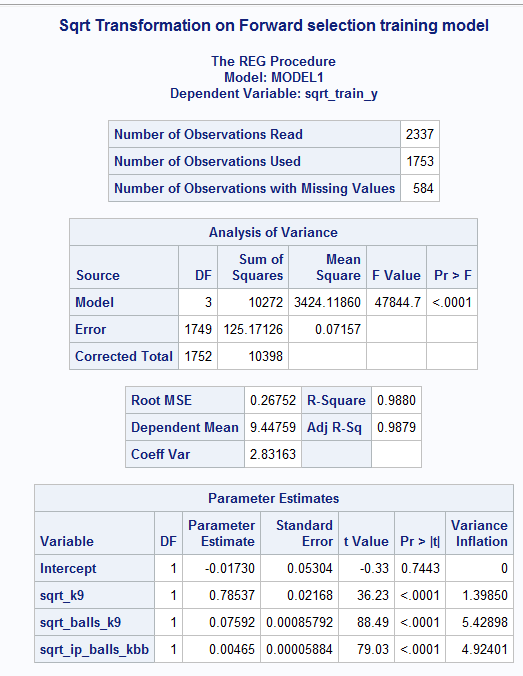
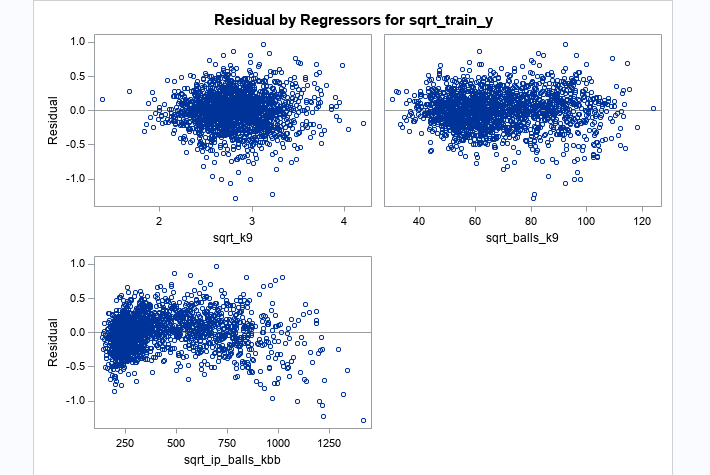
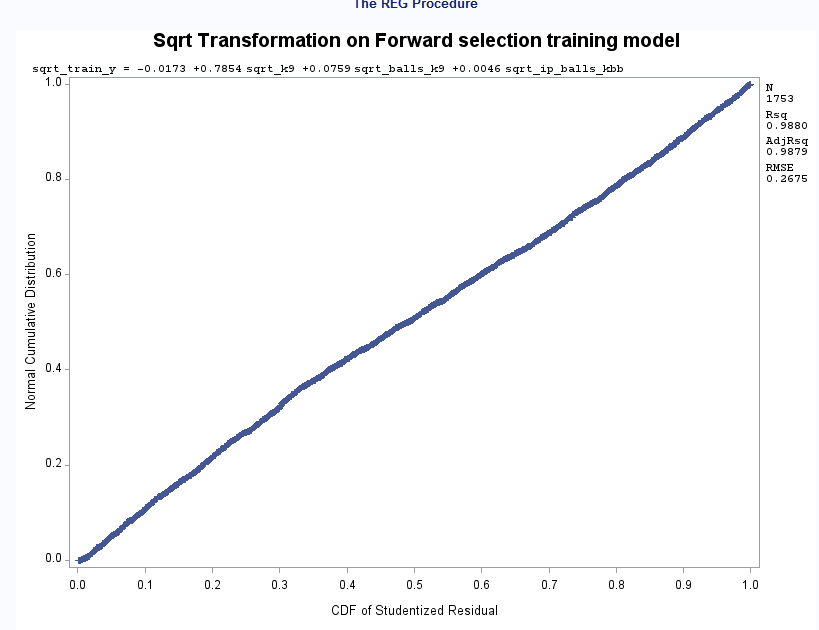
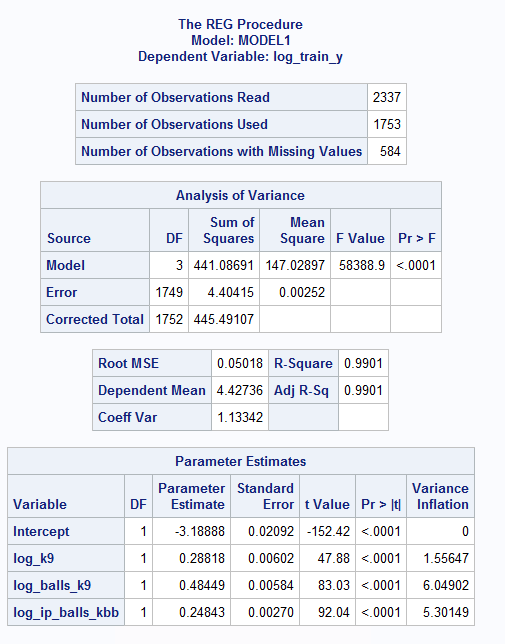
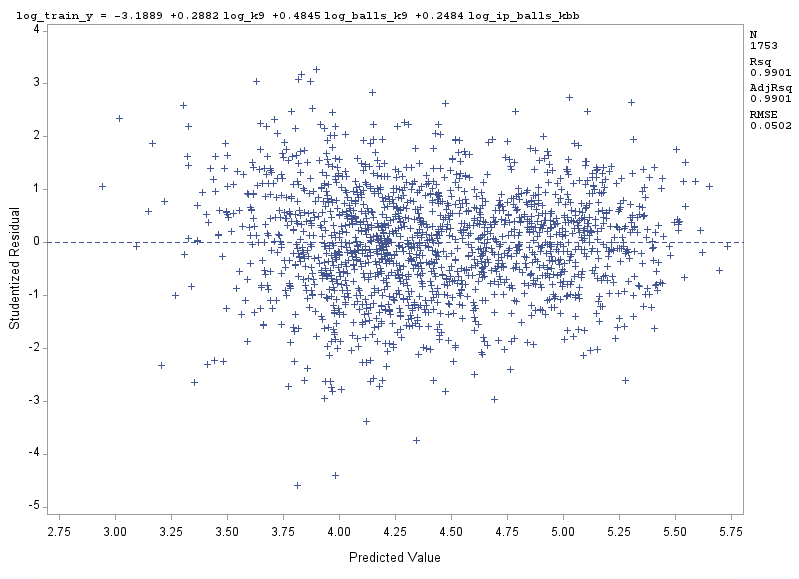
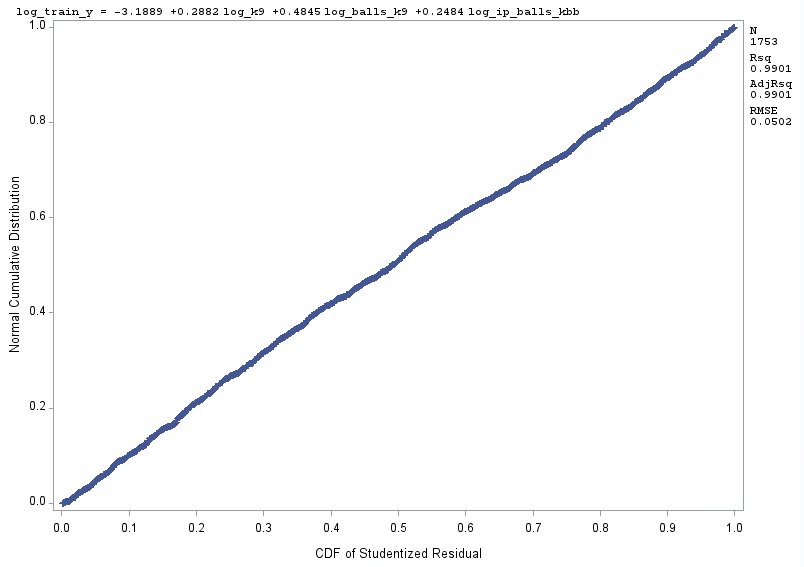
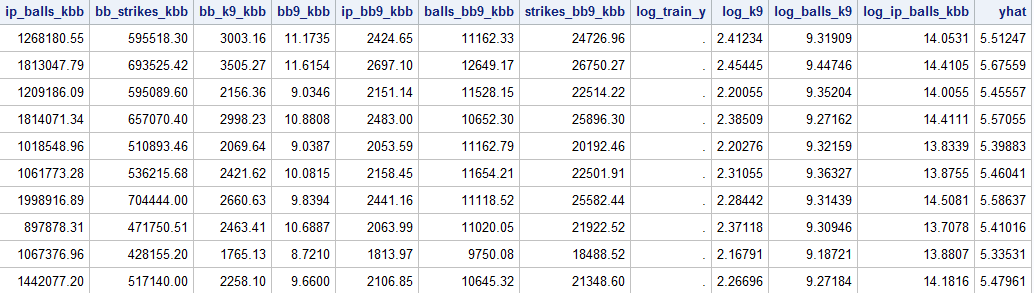
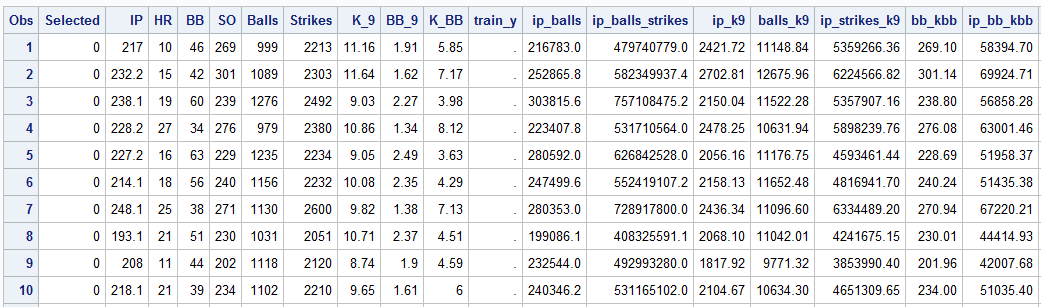
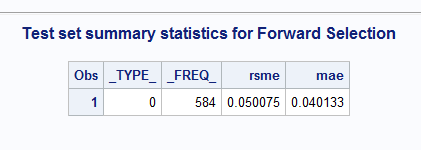
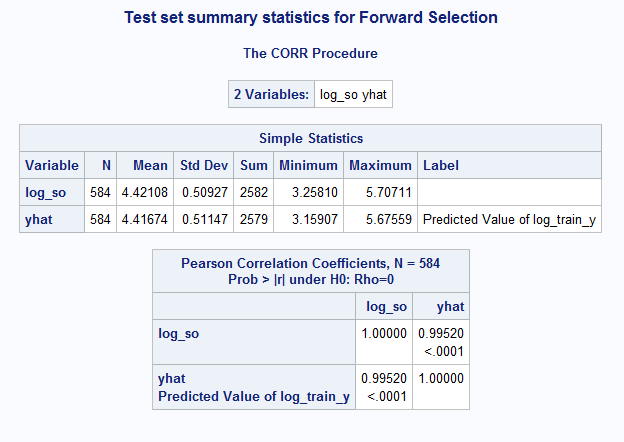
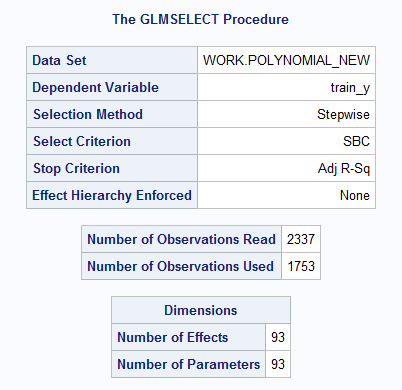
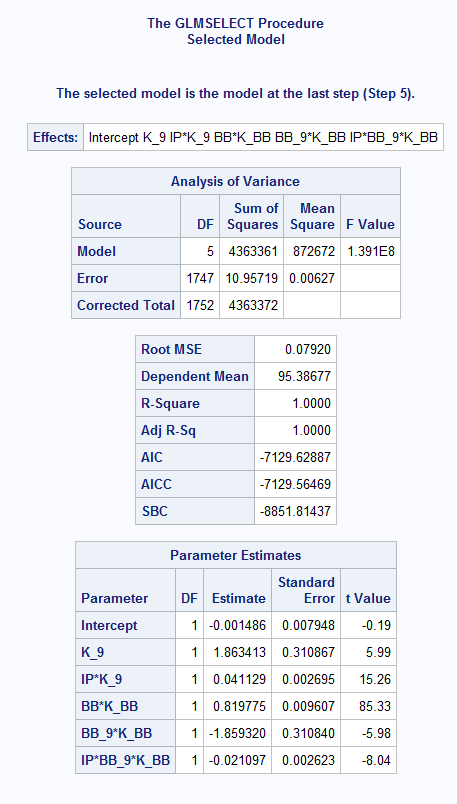
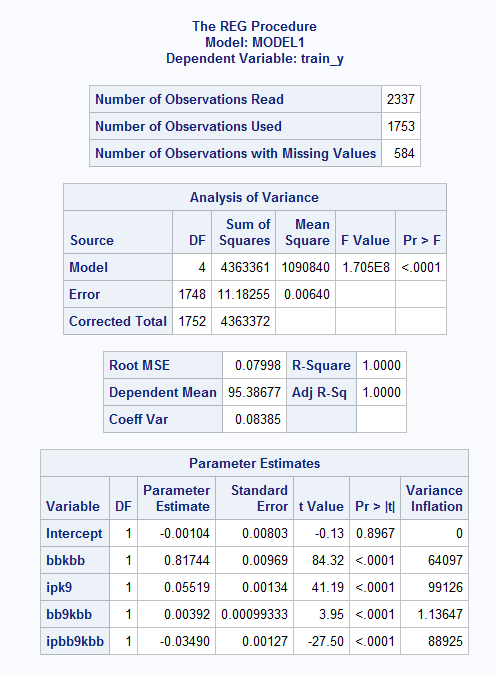
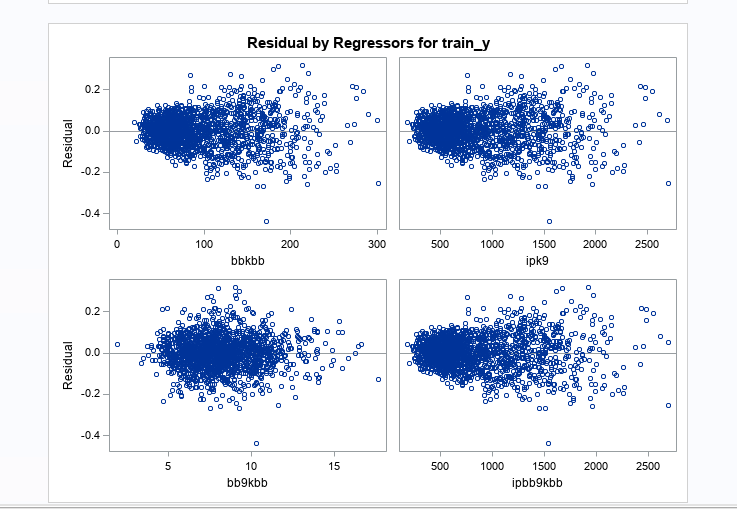
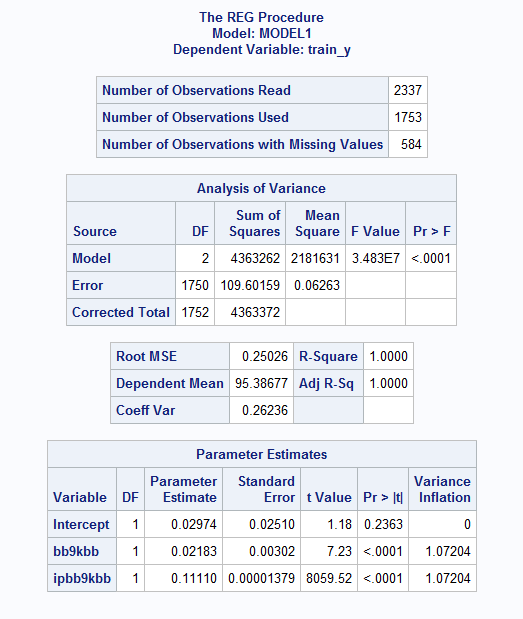
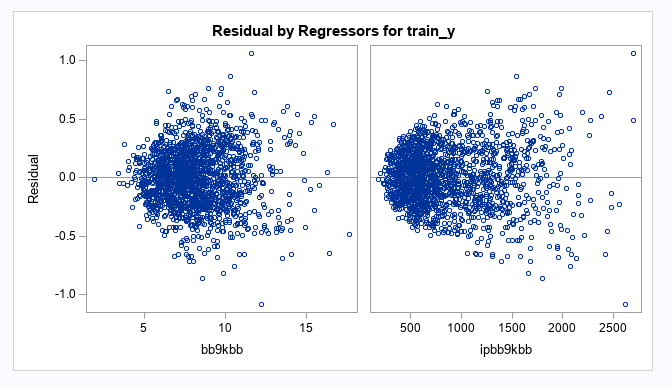
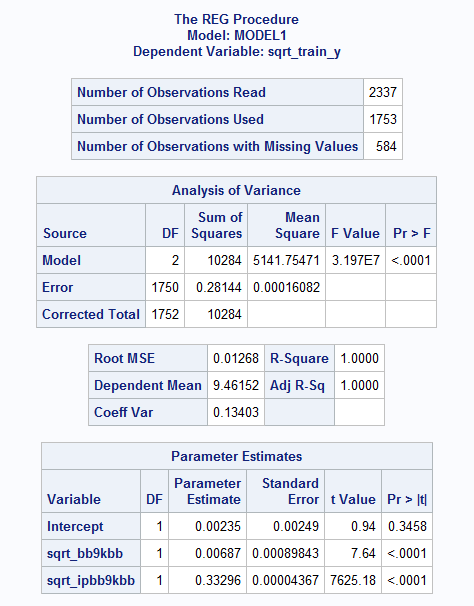
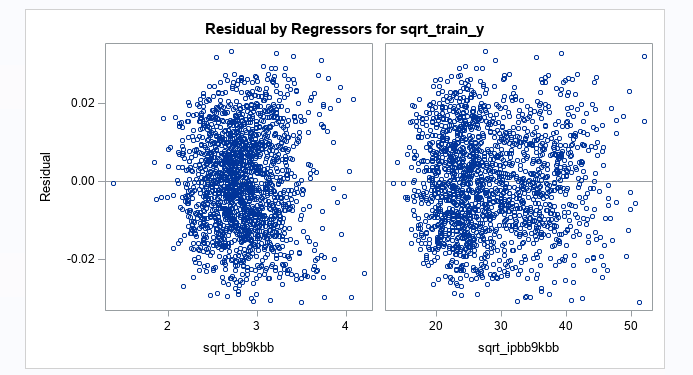
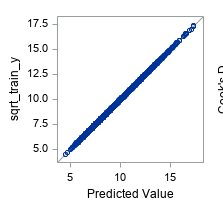
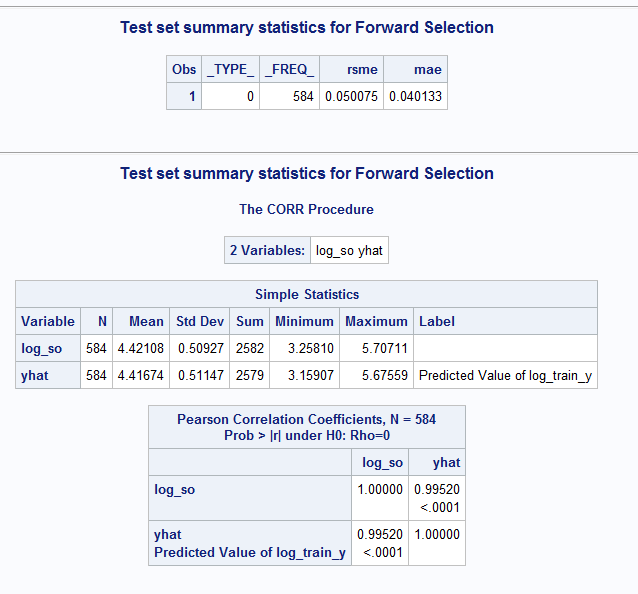
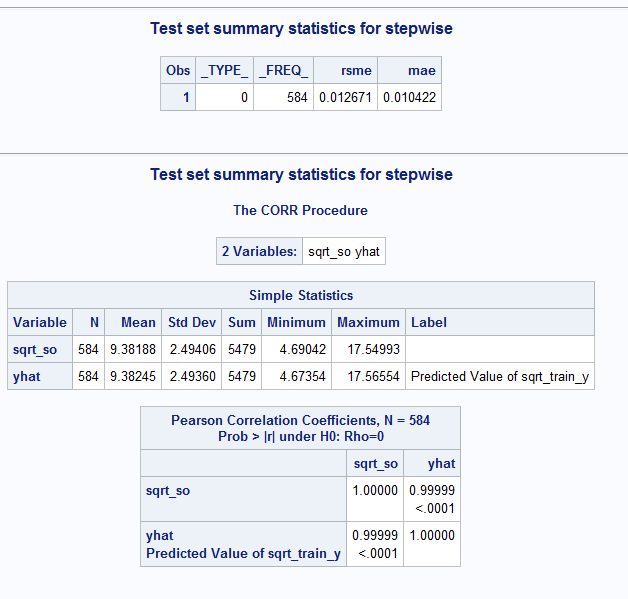
Retrieved November 15, 2019, from <https://statisticalhorizons.com/multicollinearity>.

1. Dennis Moy. (2006). Regression planes to improve the pythagorean percentage. Retrieved from [stat.berkeley.edu/~aldous/157/Old\_Projects/moy.pdf](http://stat.berkeley.edu/~aldous/157/Old_Projects/moy.pdf)

**Appendix**

1. 



1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 
12. 
13. 
14. 
15. 
16. 
17. 
18. 
19. 
20. 
21. 
22. 
23. 
24. 
25. 
26. 
27. 
28. 
29. 
30. 
31. 
32. 
33. 
34. 
35. 
36. 
37. 
38. 
39. 
40. 
41. 
42. 
43. 
44. 
45. 
46. 
47. 
48. 
49. 
50. 
51. 
52. 
53. 
54. 
55. 
56. 
57. 
58. 
59. 
60. 
61. 
62. 
63. 
64. 
65. 
66. 
67. 
68. 
69. 
70. 
71. 
72. 
73. 
74. 
75. 
76. 
77. 
78. 
79. 
80. 
81. 
82. 
83. 
84. 
85. 
86. 
87. 
88. 
89. 
90. 
91. 
92. 
93. 
94. 
95. 
96. 
97. 
98. 