Baseball Assignment #1

Derek Hughes
PRED 411 – Sec 55
Winter 2015

INTRODUCTION

Baseball, because of its nature, has a propensity for statistical analysis. For decades, many American kids remember a time when their parents took them to a professional baseball game, bought a game day program, and showed them how to "score" a game by recording the plays. Well, advances in computer processing and database performance have transformed that traditional American pastime rite of passage into something much more valuable for professional baseball teams. Specifically, we now have the ability to process hundreds of variables and thousands of baseball stats simultaneously to extract information and create models that predict the outcome of the game.

That is exactly what we are going to do here today. A good model should help the team field manager or general manager make decisions on what type of baseball players to recruit to maximize the team's potential – from a statistical perspective, of course. While some important variables, such as player injuries, are extremely difficult to predict, a statistical model can provide useful information to give a manager direction when making important player or game situation decisions.

DATA EXPLORATION

The data set we used to develop the predictive model uses approximately 2276 records from professional baseball team seasonal stats from 1871 to 2006 inclusive. Every record has been standardized to a 162 game season and includes 17 variables. These variables are:

Alphabetic List of Variables and Attributes				
#	**Variable**	**Type**	**Len**	**Label**
1	INDEX	Num	8	
2	TARGET_WINS	Num	8	
10	TEAM_BASERUN_CS	Num	8	Caught stealing
9	TEAM_BASERUN_SB	Num	8	Stolen bases
4	TEAM_BATTING_2B	Num	8	Doubles by batters
5	TEAM_BATTING_3B	Num	8	Triples by batters

Alphabetic List of Variables and Attributes				
#	Variable	Type	Len	Label
7	TEAM_BATTING_BB	Num	8	Walks by batters
3	TEAM_BATTING_H	Num	8	Base Hits by batters
11	TEAM_BATTING_HBP	Num	8	Batters hit by pitch
6	TEAM_BATTING_HR	Num	8	Homeruns by batters
8	TEAM_BATTING_SO	Num	8	Strikeouts by batters
17	TEAM_FIELDING_DP	Num	8	Double Plays
16	TEAM_FIELDING_E	Num	8	Errors
14	TEAM_PITCHING_BB	Num	8	Walks allowed
12	TEAM_PITCHING_H	Num	8	Hits allowed
13	TEAM_PITCHING_HR	Num	8	Homeruns allowed
15	TEAM_PITCHING_SO	Num	8	Strikeouts by pitchers

EDA observations:

We used PROC MEANS and PROC UNIVARIATE to learn more about the data, specifically the mean, median, standard deviation, distribution, missing values, and outliers for these variables.

Variable	N Miss	Mean	Mode	50th Pctl	Std Dev	Minimum	Maximum
INDEX	0	1268.46	.	1270.50	736.3490405	1.0000000	2535.00
TARGET_WINS	0	80.7908612	83.0000000	82.0000000	15.7521525	0	146.0000000
TEAM_BATTING_H	0	1469.27	1458.00	1454.00	144.5911954	891.000000	2554.00
TEAM_BATTING_2B	0	241.2469244	227.0000000	238.0000000	46.8014146	69.0000000	458.0000000
TEAM_BATTING_3B	0	55.2500000	35.0000000	47.0000000	27.9385570	0	223.0000000
TEAM_BATTING_HR	0	99.6120387	21.0000000	102.0000000	60.5468720	0	264.0000000
TEAM_BATTING_BB	0	501.5588752	502.0000000	512.0000000	122.6708615	0	878.0000000
TEAM_BATTING_SO	102	735.6053358	0	750.0000000	248.5264177	0	1399.00
TEAM_BASERUN_SB	131	124.7617716	65.0000000	101.0000000	87.7911660	0	697.0000000
TEAM_BASERUN_CS	772	52.8038564	52.0000000	49.0000000	22.9563376	0	201.0000000
TEAM_BATTING_HBP	2085	59.3560209	54.0000000	58.0000000	12.9671225	29.0000000	95.0000000
TEAM_PITCHING_H	0	1779.21	1494.00	1518.00	1406.84	1137.00	30132.00
TEAM_PITCHING_HR	0	105.6985940	114.0000000	107.0000000	61.2987469	0	343.0000000
TEAM_PITCHING_BB	0	553.0079086	536.0000000	536.5000000	166.3573617	0	3645.00
TEAM_PITCHING_SO	102	817.7304508	0	813.5000000	553.0850315	0	19278.00
TEAM_FIELDING_E	0	246.4806678	122.0000000	159.0000000	227.7709724	65.0000000	1898.00
TEAM_FIELDING_DP	286	146.3879397	148.0000000	149.0000000	26.2263853	52.0000000	228.0000000

We can see that six variables are missing values for records.

They are (missing):
TEAM_BATTING_SO (102)
TEAM_BASERUN_SB (131)
TEAM_BASERUN_CS (772)
TEAM_BATTING_HBP (2085)
TEAM_PITCHING_SO (102)
TEAM_FIELDING_DP (286)

Most of these variables could still be used because there is only a small amount of missing observations; however, there are two that need serious further consideration to be used in the model.

TEAM_ BASERUN_CS is missing over 700 of 2276 records. This is quite a bit, but it still has enough observations so replacing the missing values with another value should keep it useful in the model without introducing too much bias.

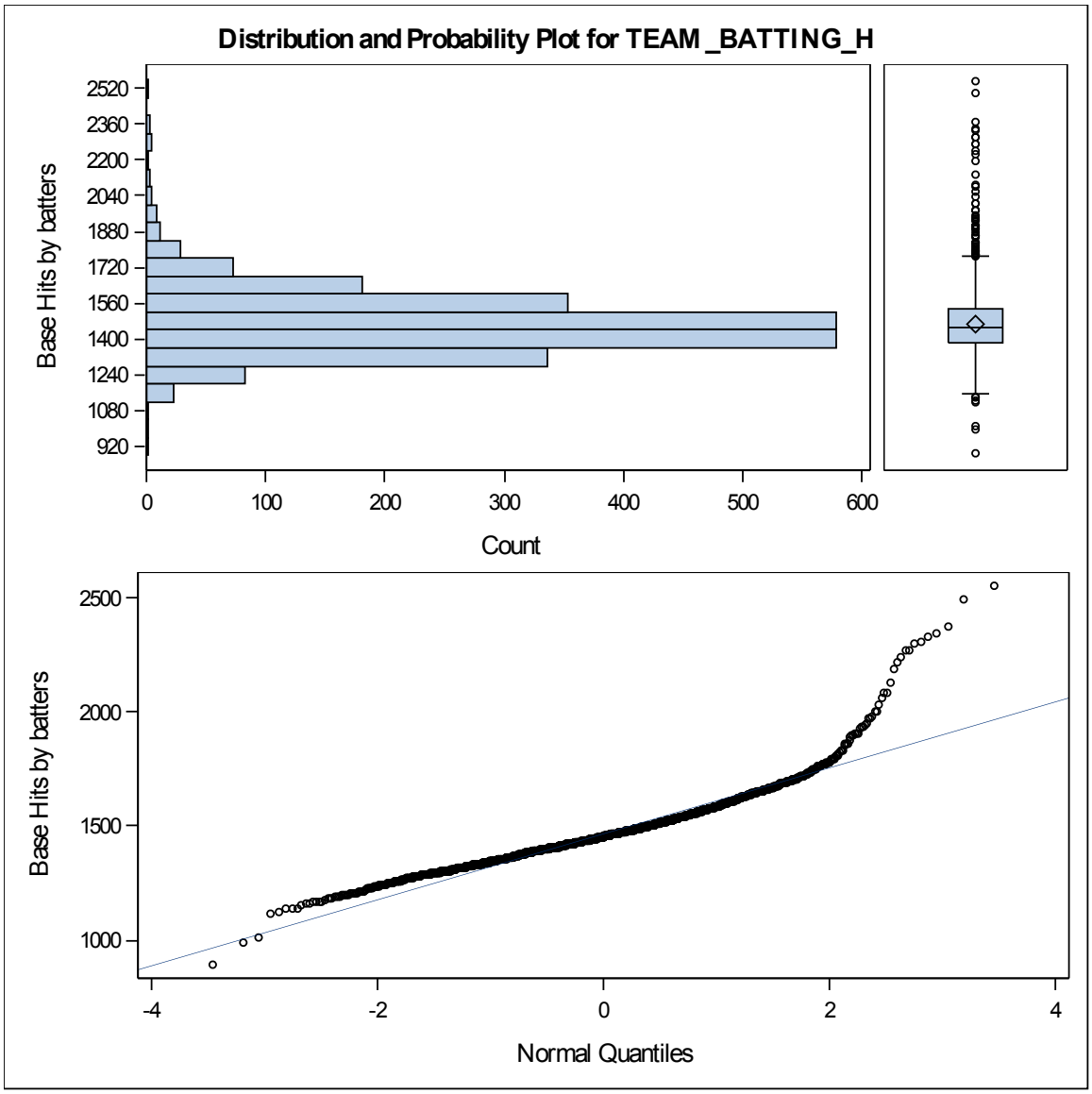
However, for TEAM_BATTING_HBP it was decided to omit the variable entirely because it does not even have 10% of the total number of observations (missing 2085 values). Another consideration was to combine it with TEAM_BATTING_BB, as getting hit by a pitch in baseball produces the exact same results as getting a walk. But, since there are so few TEAM_BATTING_HBP records, we felt it would only be added to less than 10% of the observations and, therefore, would not be evenly added to all of the observations.

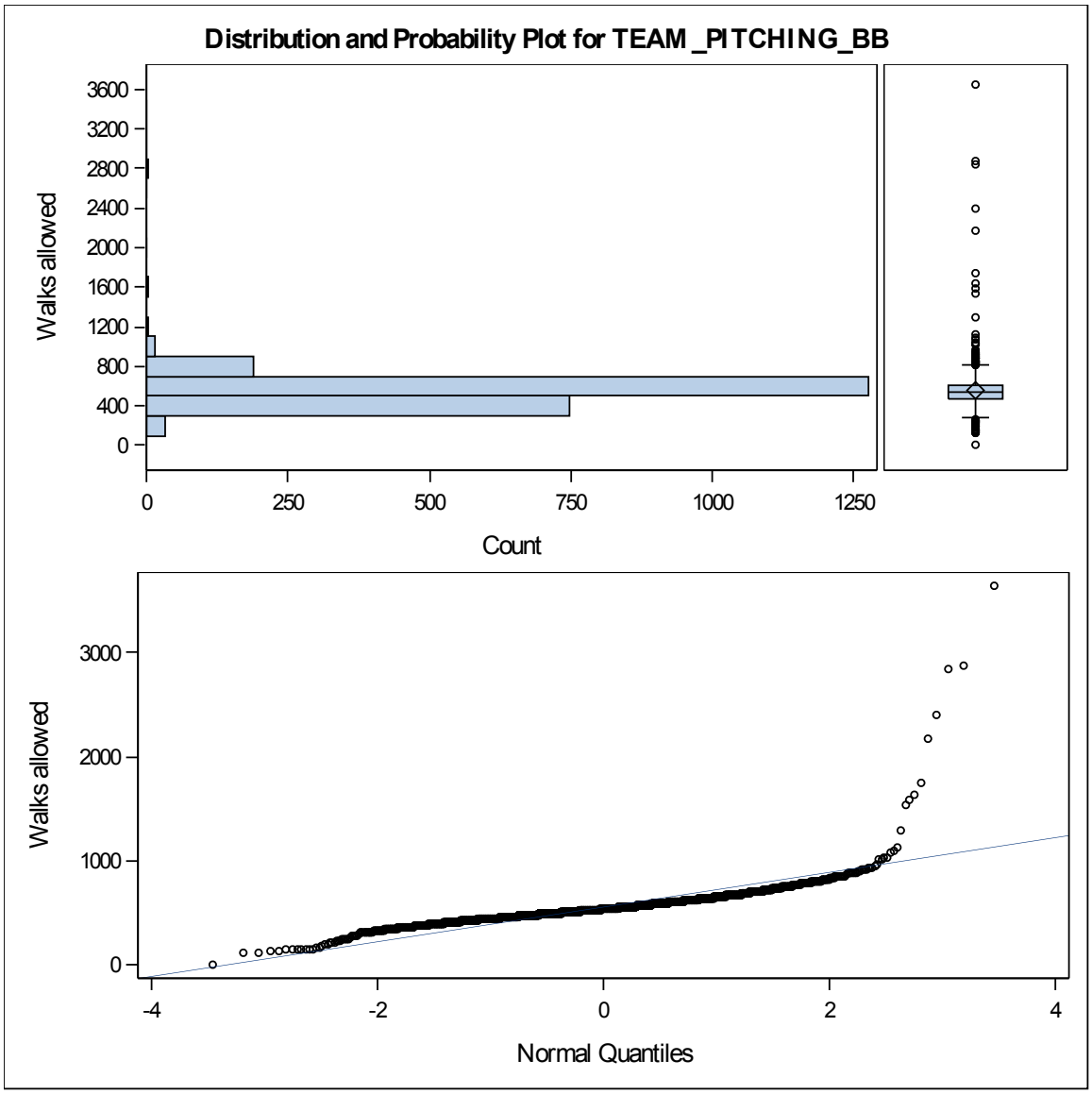
Another interesting statistic is that 10 of the variables have minimum values of zero, which is virtually impossible to achieve over an entire 162 game season. This greatly skews the mean and the standard deviations for over half of the variables. Many of these variables have standard deviations that are half the size of the mean or larger, implying there are large variations in the records. Each of these variables need to be investigated for outliers and decisions made on how to handle these outliers (addressed in DATA PREPARATION).

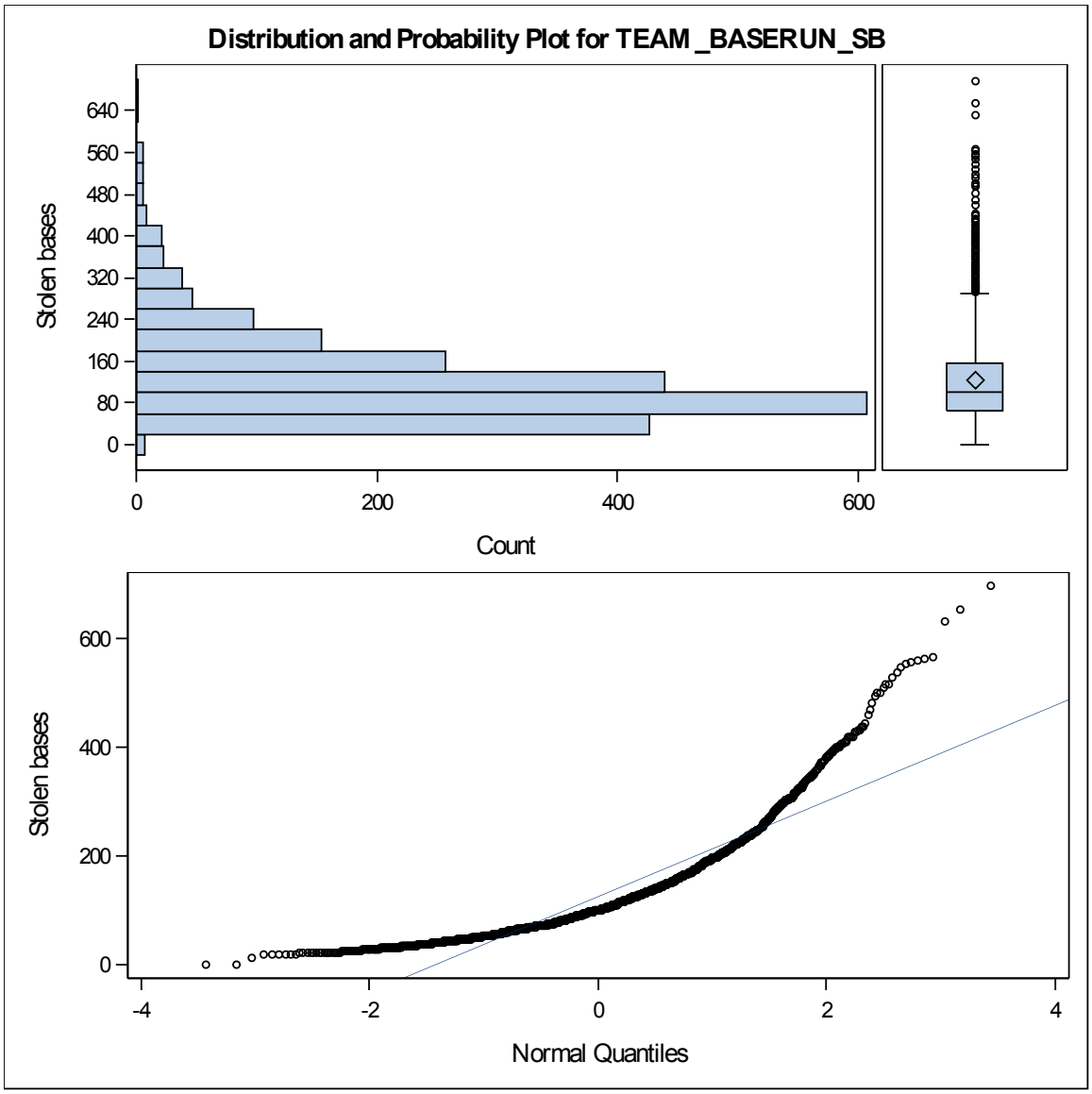
On the opposite end of the spectrum, the maximum values for TARGET_WINS, TEAM_BATTING_H ,TEAM_PITCHING_H, TEAM_PITCHING_BB, TEAM_PITCHING_SO, TEAM_FIELDING_E all have values much larger than would be expected as imaginable for one season (some of which are some 10x or greater of what would be considered normal). Similar to the minimum value scores, these variables need investigation for outliers and how to handle them. Without domain knowledge of baseball, this information might be overlooked.

A very important point to note is that some of the records were standardized to a 162 game season, which, most likely, is the reason for some of the unrealistic minimum and maximum value results throughout the dataset.

Below are a few examples of box plots and graphs from UNIVARIATE. We use these graphs to see how skewed the distributions are and for outliers. TEAM_PITCHING_BB and TEAM_BASERUN_SB are good examples of a variable being highly skewed with some strong outliers. It's important to handle these variables appropriately because they can have strong leverage and influence of the model's results.


Distribution and Probability Plot for TEAM _BATTING_H
Base Hits by batters
2520
2360
2200
2040
1880
1720
1560
1400
1240
1080
920
0
100
200
300
400
500
600
Count
Base Hits by batters
2500
2000
1500
1000
-4
-2
0
2
4
Normal Quantiles


Distribution and Probability Plot for TEAM_PITCHING_BB
Walks allowed
3600
3200
2800
2400
2000
1600
1200
800
400
0
0
250
500
750
1000
1250
Count
Walks allowed
3000
2000
1000
0
-4
-2
0
2
4
Normal Quantiles



To gain further insights and understanding of the data set, we use PROC CORR to observe which variables have the strongest correlation to the team winning the game (TARGET_WINS).

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations				
	INDEX	**TARGET_WINS**	**TEAM_BATTING_H**	**TEAM_BATTING_2B**
INDEX	1.00000 2276	-0.02106 0.3153 2276	-0.01792 0.3928 2276	0.01118 0.5939 2276
TARGET_WINS	-0.02106 0.3153 2276	1.00000 2276	0.38877 <.0001 2276	0.28910 <.0001 2276
TEAM_BATTING_H Base Hits by batters	-0.01792 0.3928 2276	0.38877 <.0001 2276	1.00000 2276	0.56285 <.0001 2276
TEAM_BATTING_2B Doubles by batters	0.01118 0.5939 2276	0.28910 <.0001 2276	0.56285 <.0001 2276	1.00000 2276
TEAM_BATTING_3B Triples by batters	-0.00581 0.7816 2276	0.14261 <.0001 2276	0.42770 <.0001 2276	-0.10731 <.0001 2276
TEAM_BATTING_HR Homeruns by batters	0.05148 0.0140 2276	0.17615 <.0001 2276	-0.00654 0.7550 2276	0.43540 <.0001 2276
TEAM_BATTING_BB Walks by batters	-0.02657 0.2052 2276	0.23256 <.0001 2276	-0.07246 0.0005 2276	0.25573 <.0001 2276
TEAM_BATTING_SO Strikeouts by batters	0.08145 0.0001 2174	-0.03175 0.1389 2174	-0.46385 <.0001 2174	0.16269 <.0001 2174
TEAM_BASERUN_SB Stolen bases	0.04027 0.0622 2145	0.13514 <.0001 2145	0.12357 <.0001 2145	-0.19976 <.0001 2145
TEAM_BASERUN_CS Caught stealing	0.00057 0.9825 1504	0.02240 0.3853 1504	0.01671 0.5174 1504	-0.09981 0.0001 1504
TEAM_BATTING_HBP Batters hit by pitch	0.07719 0.2885 191	0.07350 0.3122 191	-0.02911 0.6893 191	0.04608 0.5267 191

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations				
	INDEX	**TARGET_WINS**	**TEAM_BATTING_H**	**TEAM_BATTING_2B**
TEAM_PITCHING_H Hits allowed	0.01710 0.4148 2276	-0.10994 <.0001 2276	0.30269 <.0001 2276	0.02369 0.2585 2276
TEAM_PITCHING_HR Homeruns allowed	0.05099 0.0150 2276	0.18901 <.0001 2276	0.07285 0.0005 2276	0.45455 <.0001 2276
TEAM_PITCHING_BB Walks allowed	-0.01529 0.4660 2276	0.12417 <.0001 2276	0.09419 <.0001 2276	0.17805 <.0001 2276
TEAM_PITCHING_SO Strikeouts by pitchers	0.05589 0.0091 2174	-0.07844 0.0003 2174	-0.25266 <.0001 2174	0.06479 0.0025 2174
TEAM_FIELDING_E Errors	-0.00923 0.6598 2276	-0.17648 <.0001 2276	0.26490 <.0001 2276	-0.23515 <.0001 2276
TEAM_FIELDING_DP Double Plays	0.02006 0.3710 1990	-0.03485 0.1201 1990	0.15538 <.0001 1990	0.29088 <.0001 1990

Individual correlations to TARGET_WINS variable are listed in strength as:

Strongest positive correlations to TARGET_WINS and are significant:
TEAM_BATTING_H (.38877)
TEAM_BATTING_2B (.28910)
TEAM_BATTING_BB (.23256)
TEAM_PITCHING_HR (.18901)

Strongest negative correlations to TARGET_WINS and are significant:
TEAM_FIELDING_E (-.17648)
TEAM_PITCHING_H (-.10994)
TEAM_PITCHING_SO (-.07844)

Sometimes variables have unrealistic or opposite correlations with TARGET_WINS than expected. In these cases, each variable should be investigated, imputed, and/or adjusted for missing variables and outliers. If after the variables are transformed, and they still

produce unrealistic correlations with the TARGET_WINS variable, they should be considered for deletion from the model.

These showed unrealistic correlations:
TEAM_BASERUN_CS is positive correlation (.02240) but not significant (.3853)
TEAM_PITCHING_HR is positive correlation (.18901) AND significant (<.0001)
TEAM_PITCHING_BB is positive correlation (.12417) AND significant (<.0001)
TEAM_PITCHING_SO is negative correlation (-.07844) AND significant (.0003)
TEAM_FIELDING_DP is negative correlation (-.03485) but not significant (.1201)

We can also see which variables are correlated to other variables. This tells us which variables may be related and could be candidates to be combined to create new variables that may improve the model.

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations				
	INDEX	**TARGET_WINS**	**TEAM_BATTING_H**	**TEAM_BATTING_2B**
INDEX	1.00000 2276	-0.02106 0.3153 2276	-0.01792 0.3928 2276	0.01118 0.5939 2276
TARGET_WINS	-0.02106 0.3153 2276	1.00000 2276	0.38877 <.0001 2276	0.28910 <.0001 2276
TEAM_BATTING_H Base Hits by batters	-0.01792 0.3928 2276	0.38877 <.0001 2276	1.00000 2276	0.56285 <.0001 2276
TEAM_BATTING_2B Doubles by batters	0.01118 0.5939 2276	0.28910 <.0001 2276	0.56285 <.0001 2276	1.00000 2276
TEAM_BATTING_3B Triples by batters	-0.00581 0.7816 2276	0.14261 <.0001 2276	0.42770 <.0001 2276	-0.10731 <.0001 2276
TEAM_BATTING_HR Homeruns by batters	0.05148 0.0140 2276	0.17615 <.0001 2276	-0.00654 0.7550 2276	0.43540 <.0001 2276
TEAM_BATTING_BB Walks by batters	-0.02657 0.2052 2276	0.23256 <.0001 2276	-0.07246 0.0005 2276	0.25573 <.0001 2276
TEAM_BATTING_SO Strikeouts by batters	0.08145 0.0001 2174	-0.03175 0.1389 2174	-0.46385 <.0001 2174	0.16269 <.0001 2174
TEAM_BASERUN_SB Stolen bases	0.04027 0.0622 2145	0.13514 <.0001 2145	0.12357 <.0001 2145	-0.19976 <.0001 2145
TEAM_BASERUN_CS Caught stealing	0.00057 0.9825 1504	0.02240 0.3853 1504	0.01671 0.5174 1504	-0.09981 0.0001 1504
TEAM_BATTING_HBP Batters hit by pitch	0.07719 0.2885 191	0.07350 0.3122 191	-0.02911 0.6893 191	0.04608 0.5267 191

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations				
	INDEX	**TARGET_WINS**	**TEAM_BATTING_H**	**TEAM_BATTING_2B**
TEAM_PITCHING_H Hits allowed	0.01710 0.4148 2276	-0.10994 <.0001 2276	0.30269 <.0001 2276	0.02369 0.2585 2276
TEAM_PITCHING_HR Homeruns allowed	0.05099 0.0150 2276	0.18901 <.0001 2276	0.07285 0.0005 2276	0.45455 <.0001 2276
TEAM_PITCHING_BB Walks allowed	- 0.01529 0.4660 2276	0.12417 <.0001 2276	0.09419 <.0001 2276	0.17805 <.0001 2276
TEAM_PITCHING_SO Strikeouts by pitchers	0.05589 0.0091 2174	-0.07844 0.0003 2174	-0.25266 <.0001 2174	0.06479 0.0025 2174
TEAM_FIELDING_E Errors	- 0.00923 0.6598 2276	-0.17648 <.0001 2276	0.26490 <.0001 2276	-0.23515 <.0001 2276
TEAM_FIELDING_DP Double Plays	0.02006 0.3710 1990	-0.03485 0.1201 1990	0.15538 <.0001 1990	0.29088 <.0001 1990

Here are some of the stronger correlations between variables (possible indicator of multicollinearity). We can use this preliminary investigation to guide our focus on which variables to combine or remove because of overlapping information added to the model. We will observe the VIF numbers produced in our models to make decisions on how to handle these correlations between variables (see MODEL SELECTION).

TEAM_BATTING_H to TEAM_BATTING_2B (.56285) and TEAM_BATTING_3B (.42770)
TEAM_BATTING_2B to TEAM_BATTING_HR (.43770)
TEAM_BATTING_3B to TEAM_BASERUN_SB (.53351) to TEAM_BATTING_HR(-.63557) to TEAM_BATTING_SO (-.66978)
TEAM_BATTING_HR to TEAM_BATTING_SO (.72707) to TEAM_PITCHING_HR (.96937)

DATA PREPARATION

Handling Missing Values

The data set contains six variables with missing variables. To illustrate the importance of handling missing values, we ran an initial regression model on the full gamut of variables without any modifications.

REGRESSION ON FULL MODEL (no variable modifications)

Number of Observations Read	2276
Number of Observations Used	191
Number of Observations with Missing Values	2085

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	7	14921	2131.63809	30.09	<.0001
Error	183	12966	70.84977		
Corrected Total	190	27887			

Root MSE	8.41723	**R-Square**	0.5351
Dependent Mean	80.92670	**Adj R-Sq**	0.5173
Coeff Var	10.40105		

Obs	TEAM_PITCHING_SO	TARGET_WINS	_IN_	_P_	_EDF_	_RSQ_	_AIC_
1	.	-1	7	8	183	0.53507	821.595

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	60.40207	19.13209	3.16	0.0019	0
TEAM_BATTING_HBP	Batters hit by pitch	1	0.08679	0.04848	1.79	0.0751	1.05991
TEAM_BATTING_SO	Strikeouts by batters	1	-0.03136	0.00728	-4.31	<.0001	1.54185
TEAM_FIELDING_E	Errors	1	-0.17283	0.03962	-4.36	<.0001	1.16453
TEAM_FIELDING_DP	Double Plays	1	-0.11832	0.03513	-3.37	0.0009	1.02648
TEAM_PITCHING_BB	Walks allowed	1	0.05660	0.00939	6.02	<.0001	1.32813
TEAM_PITCHING_H	Hits allowed	1	0.02577	0.01011	2.55	0.0117	1.57573
TEAM_PITCHING_HR	Homeruns allowed	1	0.08959	0.02391	3.75	0.0002	1.60913

This model has ADJ-Rsqr=.53, AIC=821, Model DF=7, which is pretty good for the data set but these results are misleading.

The full model uses only 191 of the 2276 observations in its model because only 191 observations contain values in all 17 variables. The remaining (vast majority) of observations are missing values in at least one variable in its observation and, by default, are not used in the full model. This skews the full model's results because it uses less than 10% of the total data that is available.

The data shows that TEAM_BATTING_HBP has 2070 missing values out of a total of 2276 observations. Thus, the maximum amount of useable observations for the full model is only 206 (2276-2070 = 206). If some of those 206 observations are missing values in other variables, they will not be used in the full model as well, leaving us with only 191 observations remaining for the full model.

To improve the amount of observations, as noted earlier, the decision was made to delete the TEAM_BATTING_HBP variable. It only has values for less than 10% of the observations, is a rare statistic in baseball, and has essentially the same impact on the game as a walk (TEAM_BATTING_BB). I did consider adding TEAM_BATTING_HBP with TEAM_BATTING_BB then use the mean value for the remaining missing values in TEAM_BATTING_BB. However, again, since TEAM_BATTING_HBP will only

combine with less than 10% of the TEAM_BATTING_BB it seemed better to just remove that variable altogether.

Since the remaining five variables containing missing values have a much larger proportion of recorded values compared to missing values, the decision was to use either the mean or median value to replace the missing values.

The creation of flags to indicate variables that had missing values was not used. This was decided because the presence of a missing value is more indicative of the value not being recorded or tracked during a particular time in baseball history rather than it containing information that could predict the target variable.

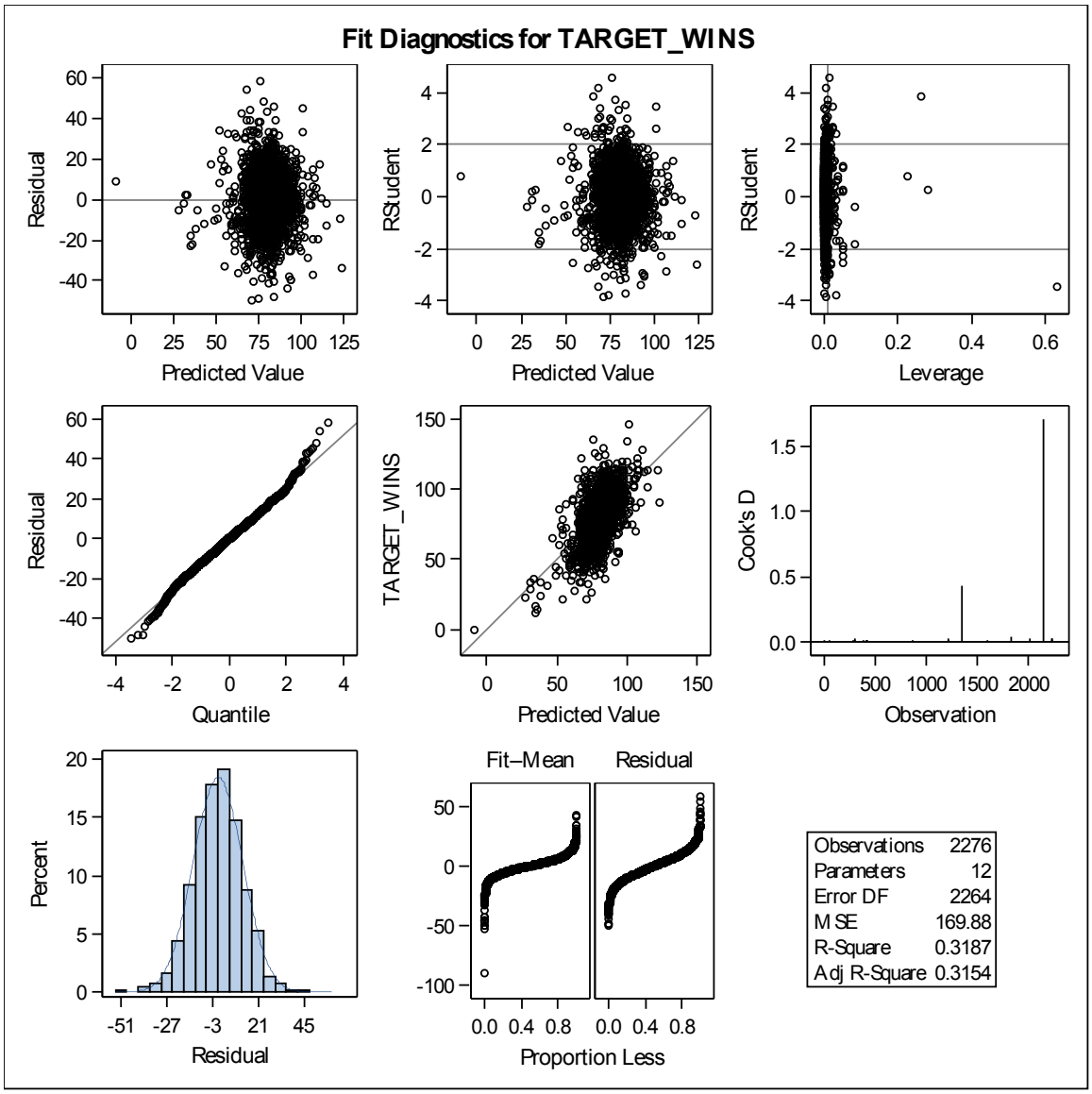
Another model was run after the missing values were replaced and the removal of BATTING_HBP.

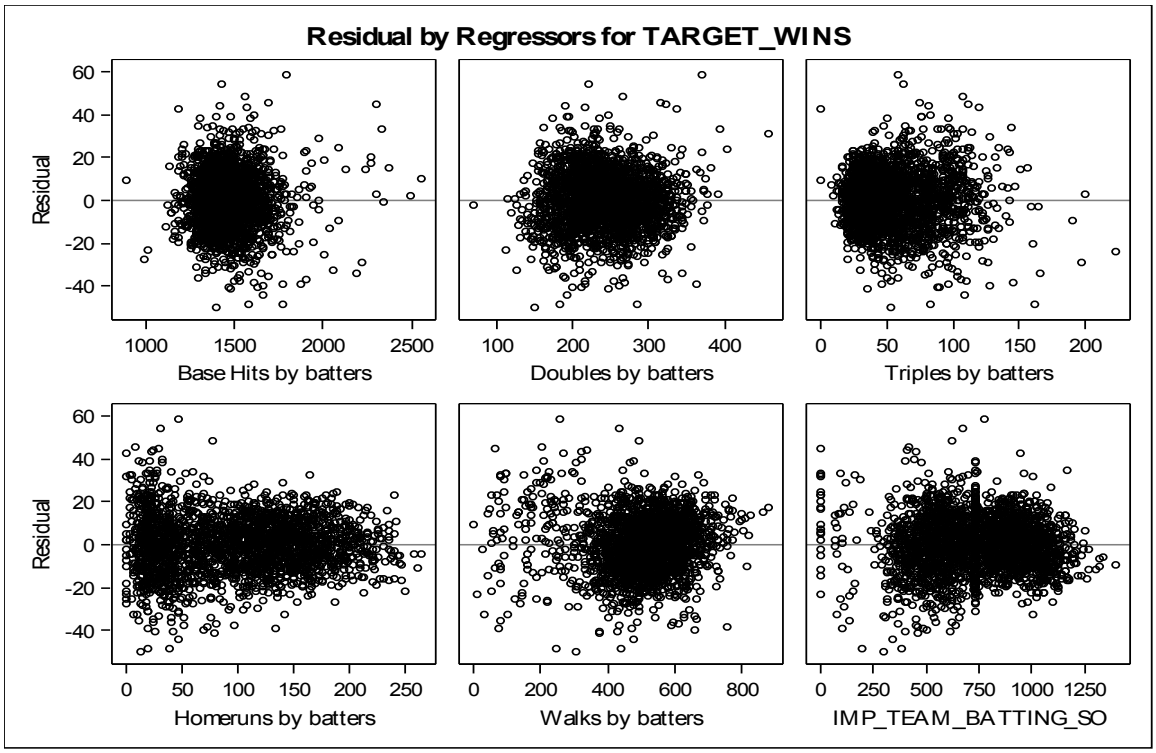
REGRESSION MODEL WITH IMPUTED MISSING VALUES ONLY

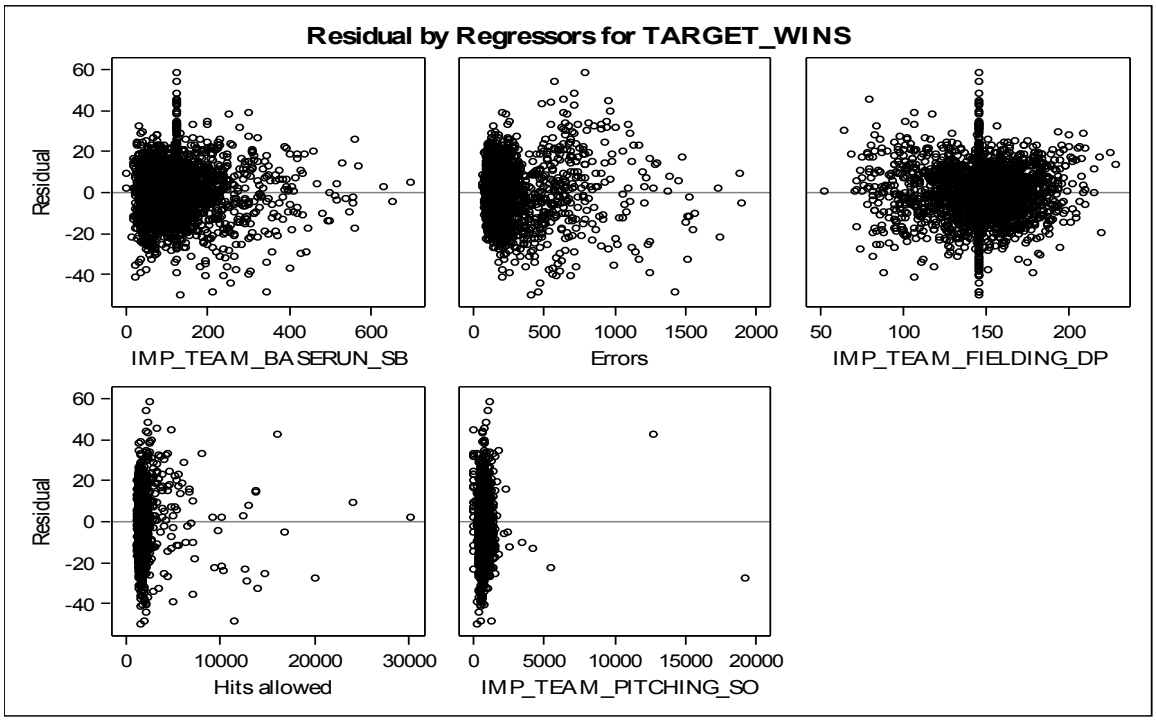
Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	11	179891	16354	96.27	<.0001
Error	2264	384605	169.87853		
Corrected Total	2275	564496			

Root MSE	13.03375	**R-Square**	0.3187
Dependent Mean	80.79086	**Adj R-Sq**	0.3154
Coeff Var	16.13270		

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	23.72283	5.22365	4.54	<.0001	0
TEAM_BATTING_H	Base Hits by batters	1	0.04845	0.00366	13.23	<.0001	3.75432
TEAM_BATTING_2B	Doubles by batters	1	-0.02049	0.00914	-2.24	0.0250	2.44800
TEAM_BATTING_3B	Triples by batters	1	0.06238	0.01658	3.76	0.0002	2.87495
TEAM_BATTING_HR	Homeruns by batters	1	0.06976	0.00962	7.25	<.0001	4.54796
TEAM_BATTING_BB	Walks by batters	1	0.01073	0.00335	3.20	0.0014	2.25939
IMP_TEAM_BATTING_SO		1	-0.00930	0.00246	-3.79	0.0002	4.76856
IMP_TEAM_BASERUN_SB		1	0.02875	0.00429	6.70	<.0001	1.79177
TEAM_FIELDING_E	Errors	1	-0.02067	0.00241	-8.57	<.0001	4.04198
IMP_TEAM_FIELDING_DP		1	-0.12118	0.01303	-9.30	<.0001	1.36640
TEAM_PITCHING_H	Hits allowed	1	-0.00068943	0.00032110	-2.15	0.0319	2.73286
IMP_TEAM_PITCHING_SO		1	0.00288	0.00067068	4.30	<.0001	1.76012


Fit Diagnostics for TARGET_WINS
Residual
Predicted Value
RStudent
Predicted Value
RStudent
Leverage
Residual
Quantile
TARGET_WINS
Predicted Value
Cook's D
Observation
Percent
Residual
Fit–Mean
Residual
Proportion Less
Observations 2276
Parameters 12
Error DF 2264
MSE 169.88
R-Square 0.3187
Adj R-Square 0.3154


Residual by Regressors for TARGET_WINS
Residual
Base Hits by batters
Doubles by batters
Triples by batters
Homeruns by batters
Walks by batters
IMP_TEAM_BATTING_SO


Residual by Regressors for TARGET_WINS
Residual
IMP_TEAM_BASERUN_SB
Errors
IMP_TEAM_FIELDING_DP
Hits allowed
IMP_TEAM_PITCHING_SO

AIC
11699.42

It achieved Adj-R^2 of .3154 and AIC of 11700 with all VIF values under 5. However, it needed 11 variables (not including intercept) and had a number of variables with inaccurate coefficients (positive or negative when should have been the opposite to predict game wins).

I used this model as a baseline for trial and error tests to determine how to trim the variables to improve performance.

Trimming variables:

After imputing the variables with missing variables, the decision was made to trim the low or top end of some variables. Where to trim was often made after doing domain research about the historical highest and lowest values ever obtained by these particular variables. For example, TEAM_BATTING_HITS was capped at 1800 because no team in the history of baseball has achieved more than 1783 hits in a single season.

Most of the trial and error testing was used to decide whether it was best to trim the variable to the high/low end of the variable's range or to simply delete the record if it exceeded either end of the range. In some instances, the model's Adj-R^2 improved by deleting the records outside of the range and in others it improved when setting their value equal to the high/low end of the variables range.

The variables that improved Adj-R^2 when trimmed were trimmed on the following criteria:

IMP_TEAM_BASERUN_SB > 480 then delete because averaging more than three stolen bases a game (which is more than 480 for a season) is unrealistic.

IMP_TEAM_BASERUN_CS > 160 then delete because getting caught stealing more than once a game on average is improbable.

TEAM_BATTING_H > 1800 then delete because this would mean a team would have to average over 10 hits a game which is unrealistic.

IMP_TEAM_PITCHING_SO <= 300 then delete and IMP_TEAM_PITCHING_SO > 1600 then delete. Deleting these values instead of having them equal the lower/upper limits improved ADJ-R^2.

IMP_TEAM_BATTING_SO <= 162 then delete and IMP_TEAM_BATTING_SO > 1600 then IMP_TEAM_BATTING_SO = 1600. Trimming values for IMP_TEAM_BATTING_SO to help VIF and by deleting lower end instead of having it equal the lower limit improved ADJ R-sqr and VIF.

TEAM_BATTING_3B > 134 then TEAM_BATTING_3B = 135
Trimmed low end values for TEAM_BATTING_3B instead of deleting because this ended up reducing Adj-R^2.

TEAM_PITCHING_H > 2300 then delete because giving up an average of 14 hits a game is very unrealistic.

TEAM_FIELDING_E > 650 then delete because this would mean a team would average over 4 errors a game over a season which, again, is extremely unrealistic.

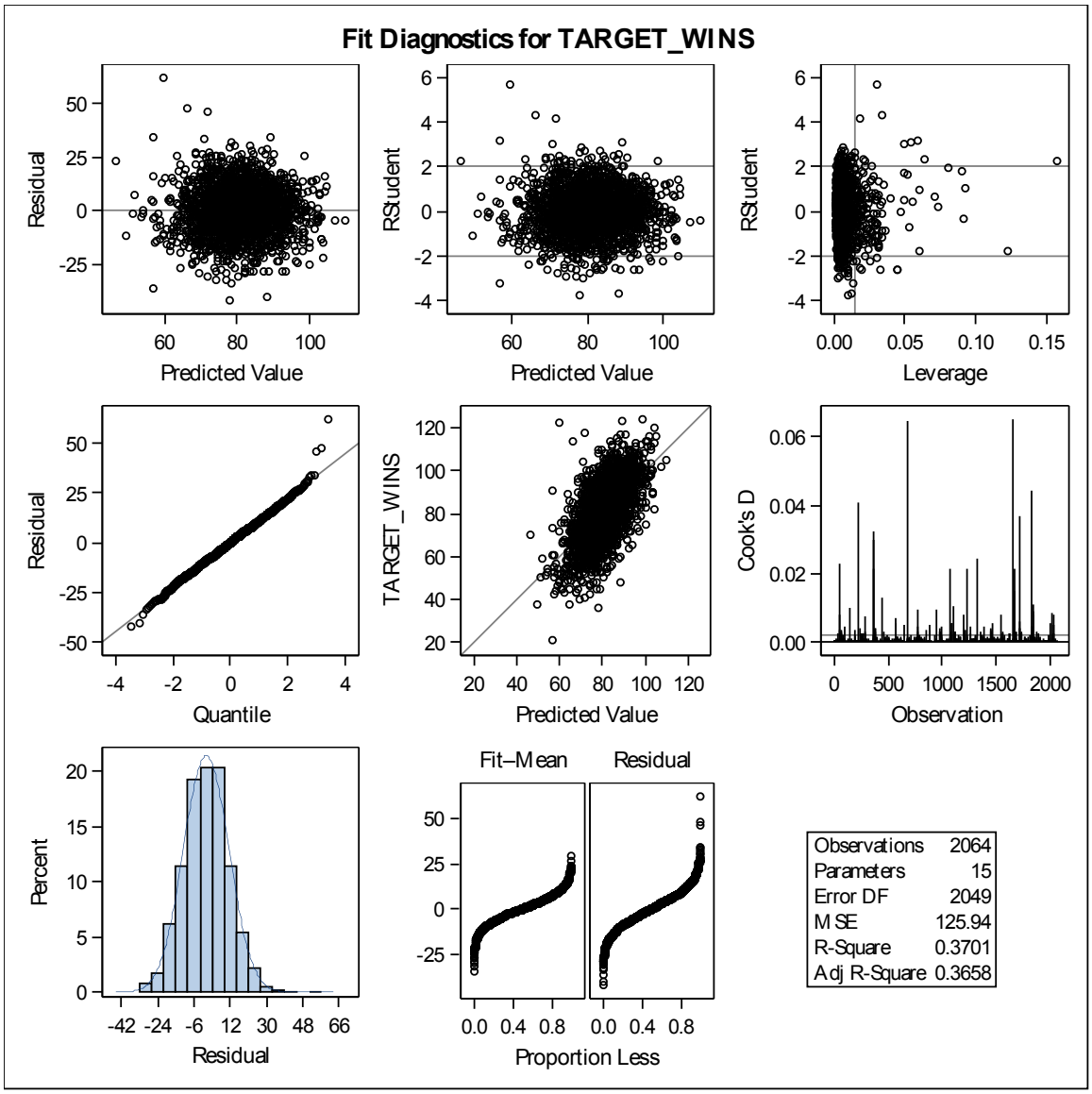
Finally, after completing the imputing and trimming of variables, I ran another regression model (TEMPFILE2 REGRESSION).

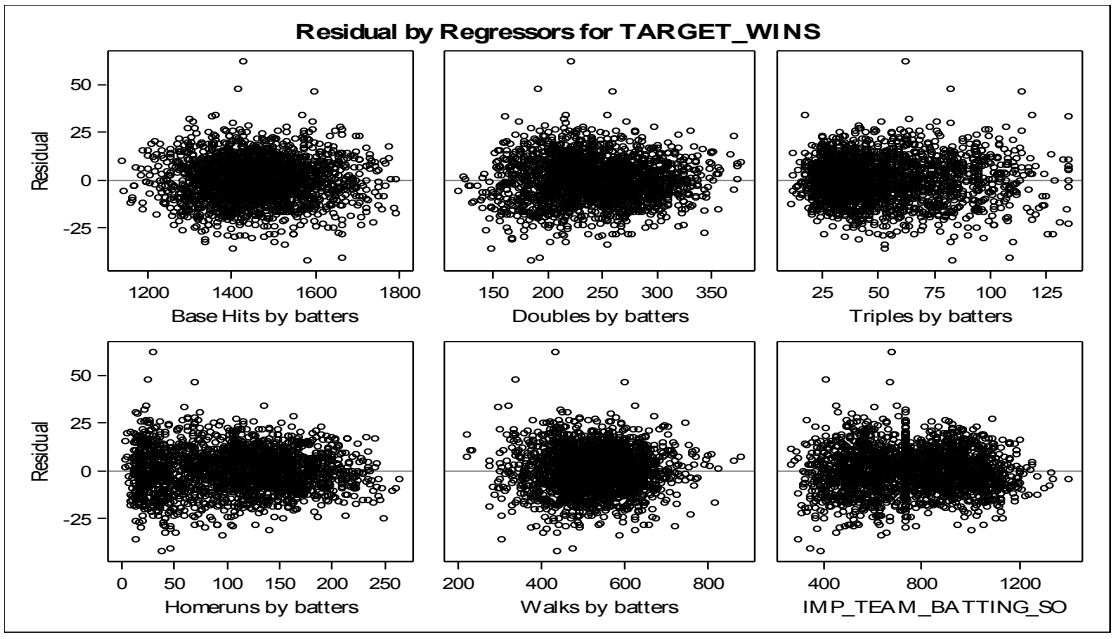
REGRESSION MODEL – IMPUTED MISSING VALUES & TRIMMED VARIABLES

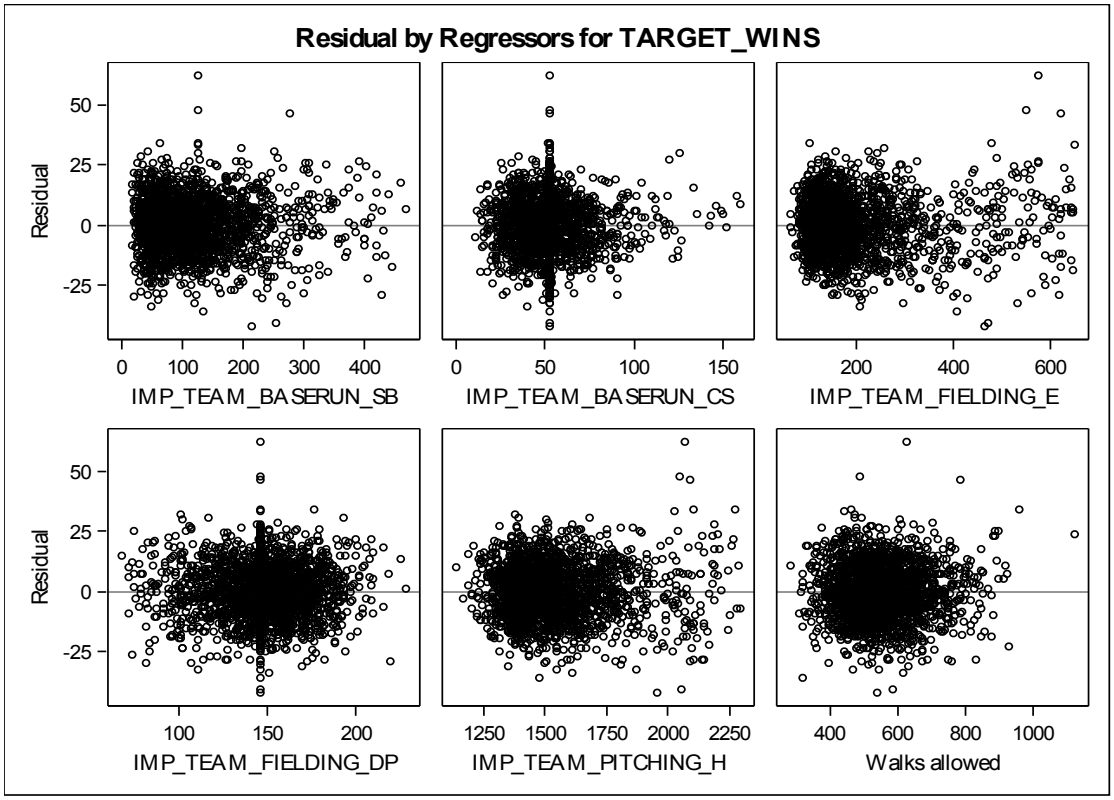
Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	14	151623	10830	85.99	<.0001
Error	2049	258053	125.94093		
Corrected Total	2063	409676			

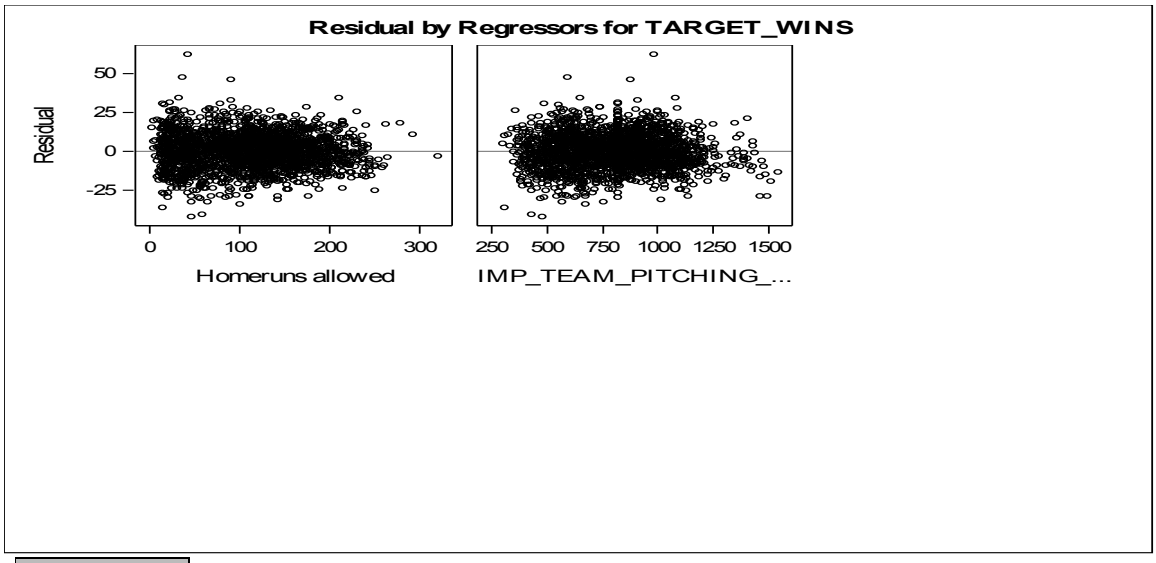
Root MSE	11.22234	**R-Square**	0.3701
Dependent Mean	80.91521	**Adj R-Sq**	0.3658
Coeff Var	13.86926		

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	46.77478	5.92885	7.89	<.0001	0
TEAM_BATTING_H	Base Hits by batters	1	-0.04511	0.01196	-3.77	0.0002	29.83752
TEAM_BATTING_2B	Doubles by batters	1	-0.03784	0.00901	-4.20	<.0001	2.67782
TEAM_BATTING_3B	Triples by batters	1	0.16609	0.01856	8.95	<.0001	3.48296
TEAM_BATTING_HR	Homeruns by batters	1	0.51667	0.07715	6.70	<.0001	330.12701
TEAM_BATTING_BB	Walks by batters	1	0.28761	0.02947	9.76	<.0001	118.06264
IMP_TEAM_BATTING_SO		1	-0.11375	0.01575	-7.22	<.0001	196.23724
IMP_TEAM_BASERUN_SB		1	0.08397	0.00543	15.47	<.0001	2.72514
IMP_TEAM_BASERUN_CS		1	-0.06938	0.01651	-4.20	<.0001	1.33504
IMP_TEAM_FIELDING_E		1	-0.08805	0.00496	-17.75	<.0001	4.93951
IMP_TEAM_FIELDING_DP		1	-0.09203	0.01213	-7.58	<.0001	1.41271
IMP_TEAM_PITCHING_H		1	0.07161	0.01022	7.01	<.0001	62.23545
TEAM_PITCHING_BB	Walks allowed	1	-0.24074	0.02725	-8.83	<.0001	127.71957
TEAM_PITCHING_HR	Homeruns allowed	1	-0.41278	0.07315	-5.64	<.0001	306.77224
IMP_TEAM_PITCHING_SO		1	0.09243	0.01459	6.33	<.0001	167.13165


Fit Diagnostics for TARGET_WINS
Residual
Predicted Value
RStudent
Predicted Value
RStudent
Leverage
Residual
Quantile
TARGET_WINS
Predicted Value
Cook's D
Observation
Percent
Residual
Fit–Mean
Residual
Proportion Less
Observations 2064
Parameters 15
Error DF 2049
MSE 125.94
R-Square 0.3701
Adj R-Square 0.3658


Residual by Regressors for TARGET_WINS
Residual
50
25
0
-25
1200
1400
1600
1800
Base Hits by batters
150
200
250
300
350
Doubles by batters
25
50
75
100
125
Triples by batters
0
50
100
150
200
250
Homeruns by batters
200
400
600
800
Walks by batters
400
800
1200
IMP_TEAM_BATTING_SO


Residual by Regressors for TARGET_WINS
Residual
50
25
0
-25
0
100
200
300
400
IMP_TEAM_BASERUN_SB
0
50
100
150
IMP_TEAM_BASERUN_CS
200
400
600
IMP_TEAM_FIELDING_E
100
150
200
IMP_TEAM_FIELDING_DP
1250
1500
1750
2000
2250
IMP_TEAM_PITCHING_H
400
600
800
1000
Walks allowed



AIC
9996.06

The Adj-R^2 had improved to .3658 and AIC improved to 9996, but now the model had extreme multicollinearity and improper sign (pos/neg) coefficients for many variables. Massive transformation and/or removing of variables was needed to create a reliable model.

Transforming variables:

Again, I ran PROC UNIVARIATE to focus on the distributions of the variables relative to their residuals.

By focusing on variables that were strongly skewed or with outliers, I was able to identify a few variables that could potentially improve the model if they were transformed using logarithmic or squared transformations. The variables identified for transformation and their results are listed below.

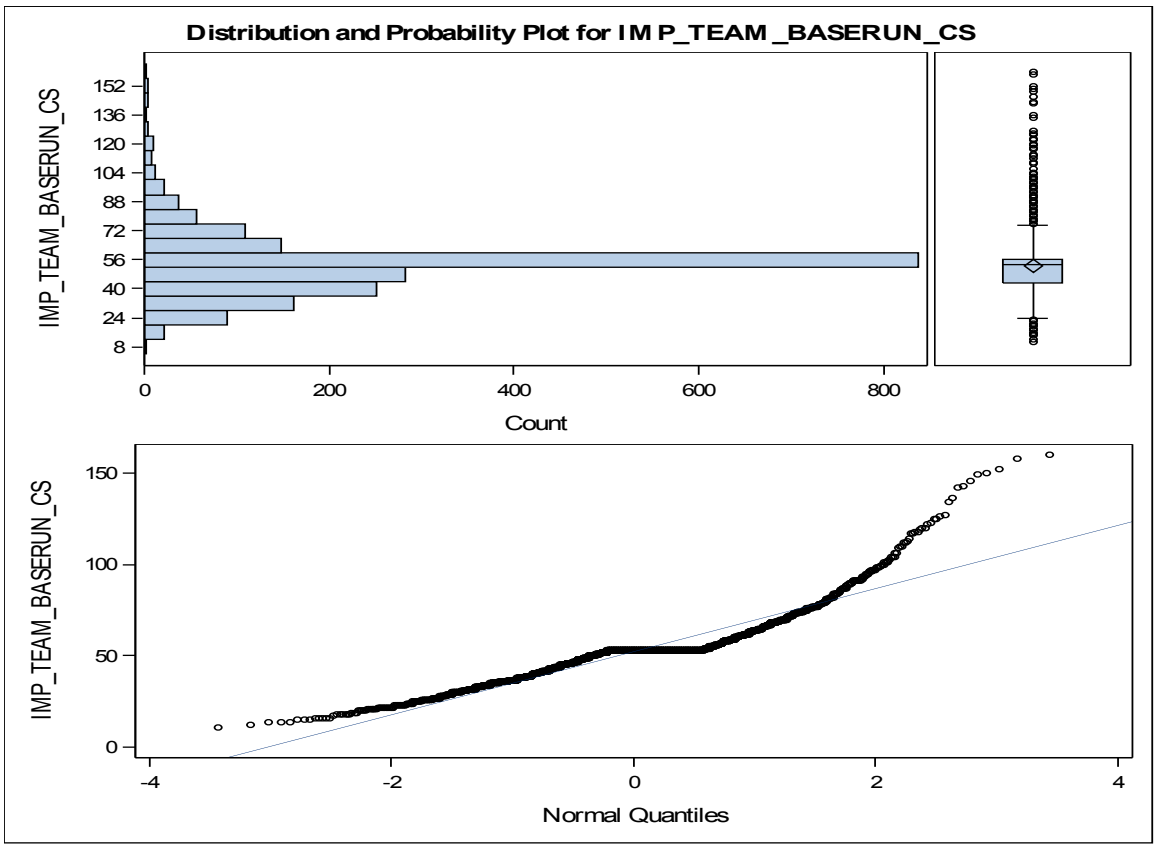
LOG10_TEAM_BASERUN_CS

Logarithmic transformation of IMP_TEAM_BASERUN_CS variable because skewed

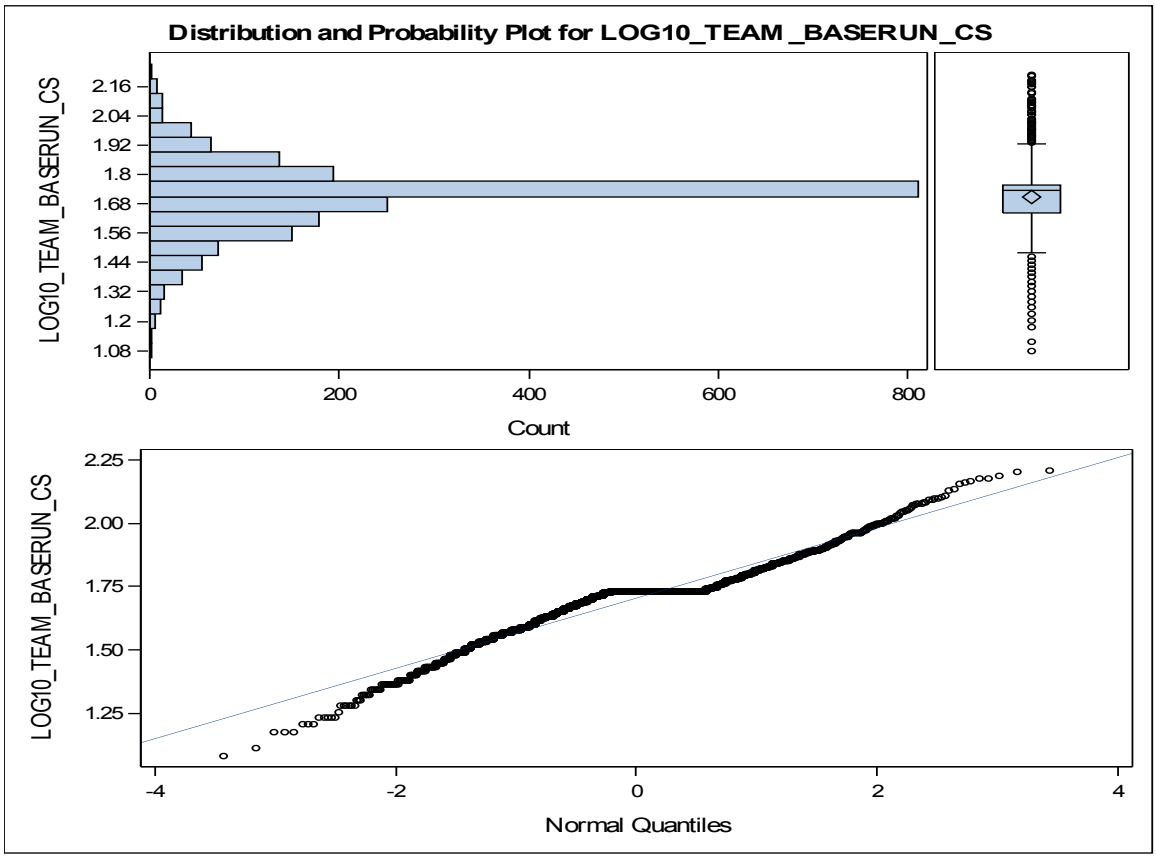
LOG10_TEAM_BASERUN_CS = sign(IMP_TEAM_BASERUN_CS) * log10(abs(IMP_TEAM_BASERUN_CS)+1)

Moments			
N	2064	**Sum Weights**	2064
Mean	1.70565227	**Sum Observations**	3520.46629
Std Deviation	0.13898932	**Variance**	0.01931803
Skewness	-0.4270763	**Kurtosis**	1.96582321
Uncorrected SS	6044.54441	**Corrected SS**	39.8530979
Coeff Variation	8.14874885	**Std Error Mean**	0.00305933

BEFORE TRANSFORMATION:



AFTER TRANSFORMATION:



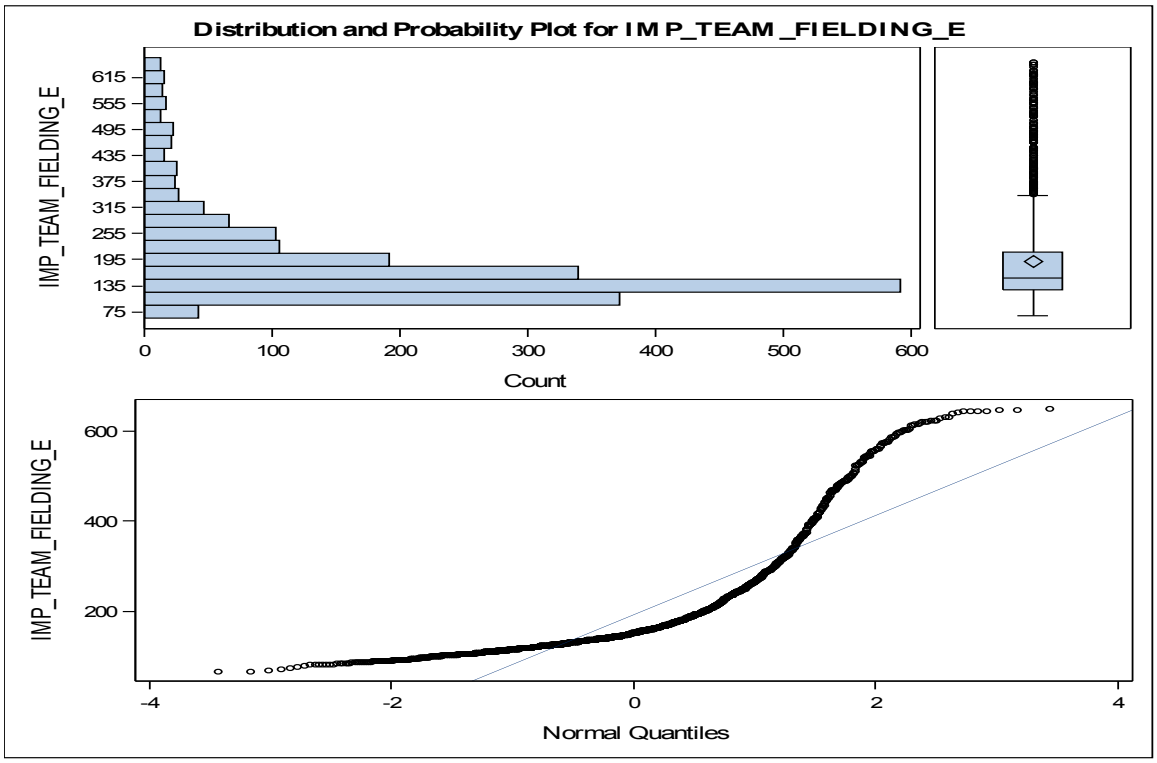
SQRT_IMP_TEAM_FIELDING_E

Square root transformation of IMP_TEAM_FIELDING_E variable because skewed. This raised ADJ-R^2 and lowered AIC compared to the logarithmic transformation.

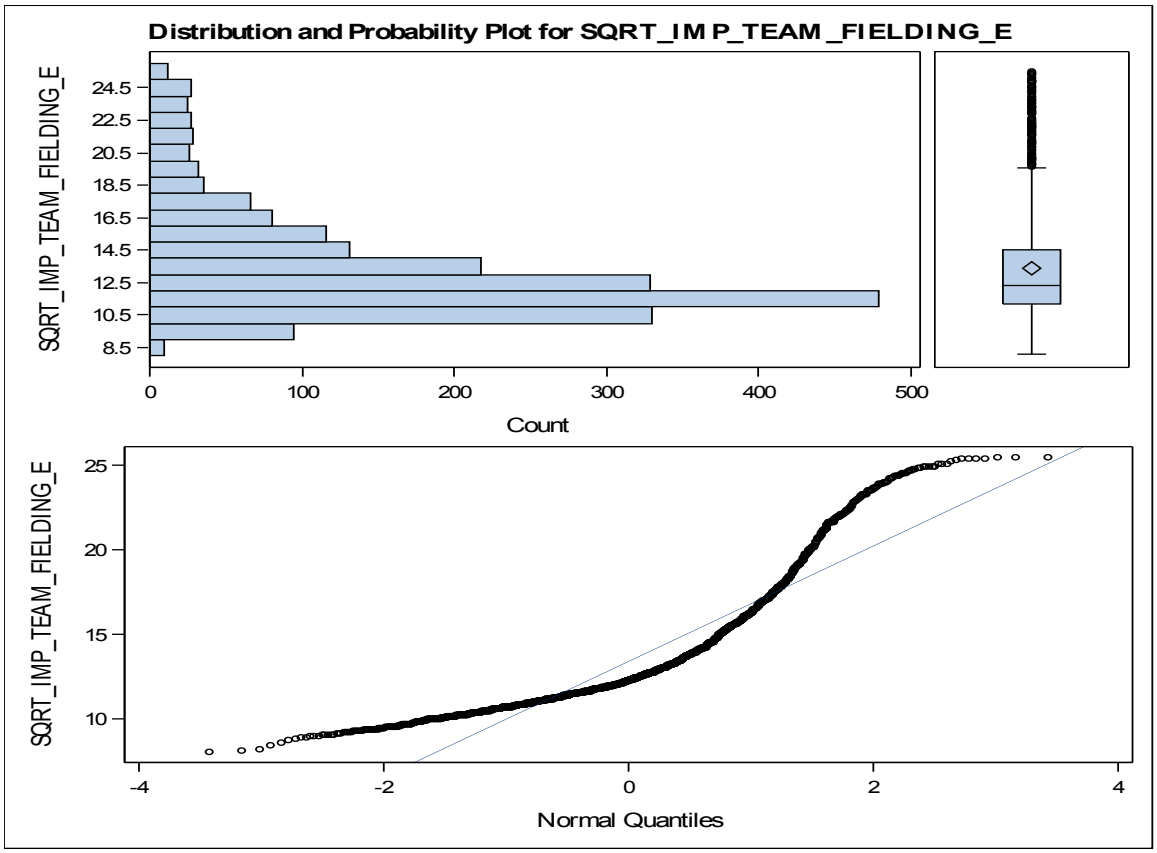
SQRT_IMP_TEAM_FIELDING_E = SQRT(IMP_TEAM_FIELDING_E)

Moments			
N	2064	**Sum Weights**	2064
Mean	13.4205658	**Sum Observations**	27700.0479
Std Deviation	3.41006495	**Variance**	11.628543
Skewness	1.55904606	**Kurtosis**	2.13038658
Uncorrected SS	395740	**Corrected SS**	23989.6841
Coeff Variation	25.4092487	**Std Error Mean**	0.07505987

BEFORE TRANSFORMATION:



AFTER TRANSFORMATION:



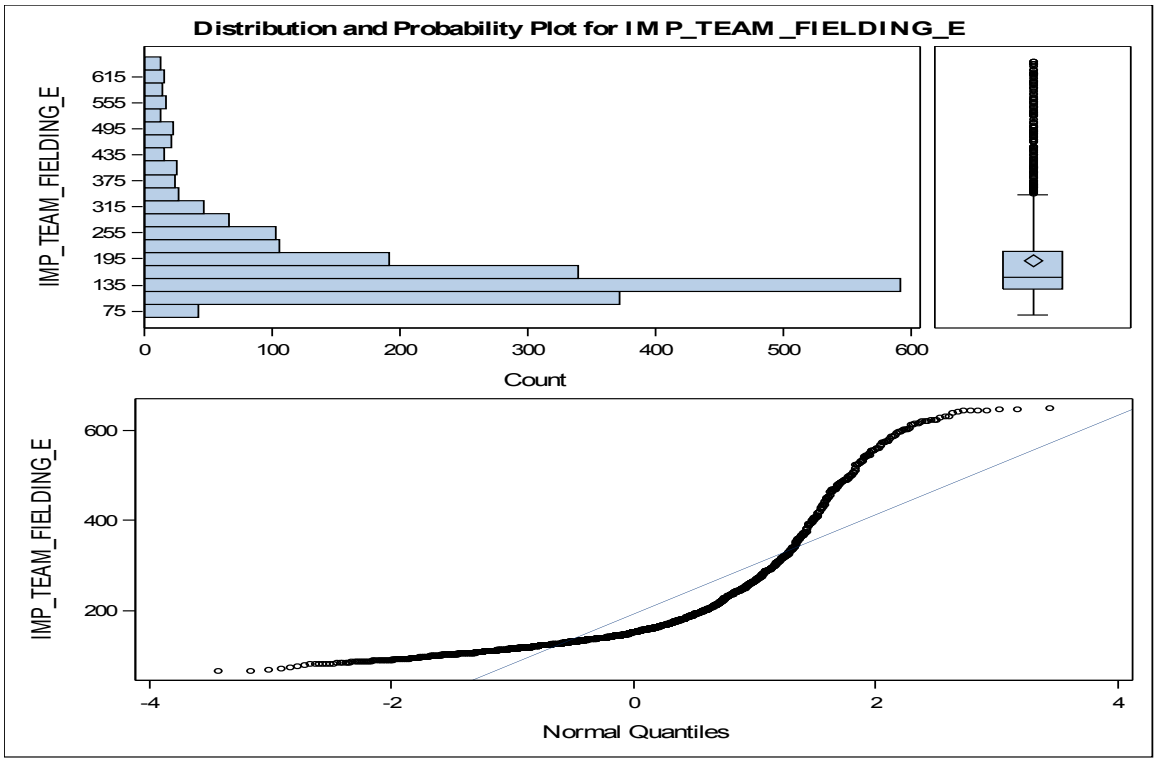
LOG10_IMP_TEAM_FIELDING_E

Logarithmic transformation of IMP_TEAM_FIELDING_E variable because skewed. This was better than no transformation (in terms of Adj-R^2), but not as good compared to squaring the variable.

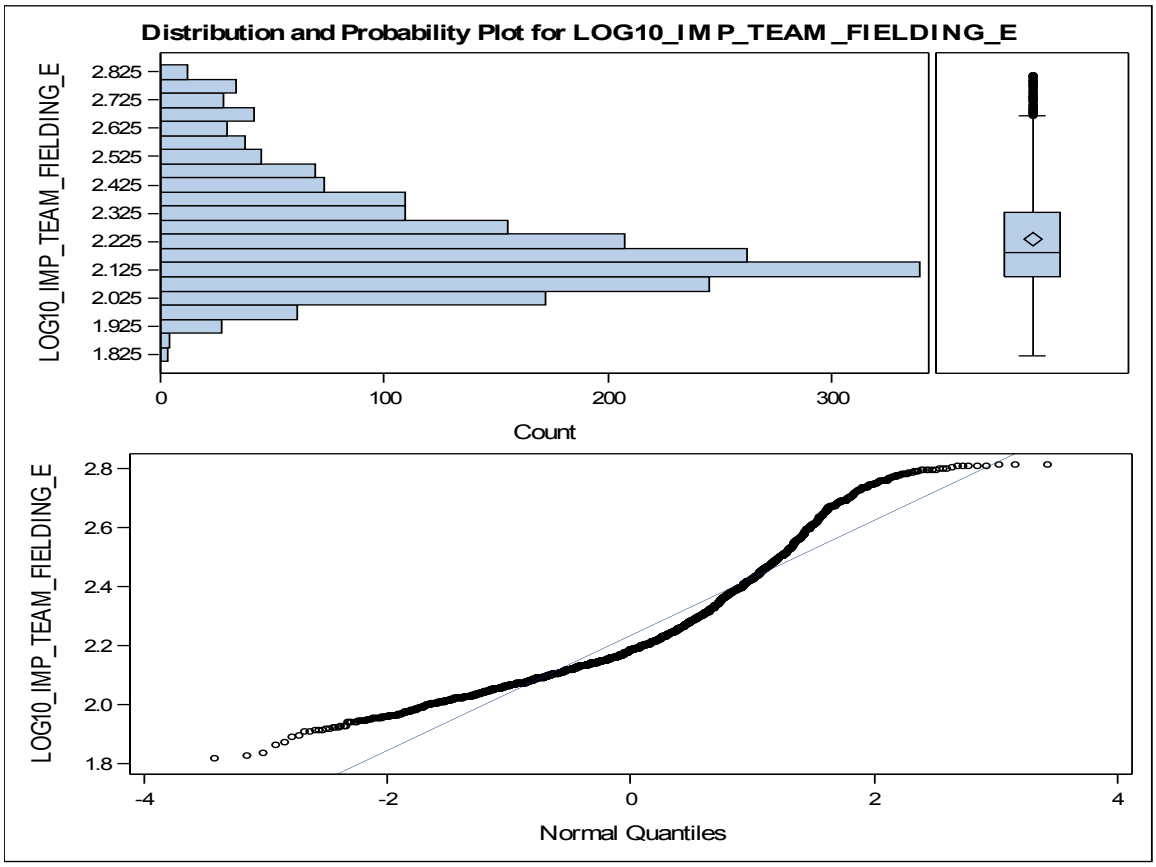
LOG10_IMP_TEAM_FIELDING_E = sign(IMP_TEAM_FIELDING_E) * log10(abs(IMP_TEAM_FIELDING_E)+1);

Moments			
N	2064	**Sum Weights**	2064
Mean	2.23461542	**Sum Observations**	4612.24623
Std Deviation	0.1944098	**Variance**	0.03779517
Skewness	1.02896486	**Kurtosis**	0.61152913
Uncorrected SS	10384.568	**Corrected SS**	77.9714392
Coeff Variation	8.69992222	**Std Error Mean**	0.00427921

BEFORE TRANSFORMATION:



AFTER TRANSFORMATION



Created/combined variables:

Using domain knowledge along with the correlations between variables (see DATA EXPLORATION), I combined a few variables to create some new variables that might improve the model. The variables created and the results are listed below. Furthermore,when the VIF value for a variable was greater than 9 or 10 I tried combining variables or dropping variables to see if it would reduce the multicollinearity issues.

The created variables and reasons are listed below:

Combining variables Batting_2B and Batting_HR b/c correlation .44. This combination did not improve the model.
BAT_2B_HR = TEAM_BATTING_2B * TEAM_BATTING_HR

Combining variables Hits*2B because correlation H-2B = .56 and H-3B = .42, but 2b-3b =1. This combination did not improve the model.
H_2B_3B = TEAM_BATTING_H * TEAM_BATTING_2B * TEAM_BATTING_3B

Combining variables Hits*2B*3B because correlation H-2B = .61. This was a fairly strong correlation but did not make it into the final model using stepwise selection process.
H_2B = TEAM_BATTING_H * TEAM_BATTING_2B

Creating variable SINGLE by combining variables of Hits-2B-3B-HR to measure if simply hitting a single (1B) has influence. This was an improvement to the model and was included.
SINGLES = TEAM_BATTING_H - TEAM_BATTING_2B - TEAM_BATTING_3B - TEAM_BATTING_HR

Combining variables of team pitching hits and walks in an attempt to reduce VIF of each variable and create a new variable PITCH_HITS_WALK. This made it into the model but was eventually manually removed because it had a positive correlation when it should have had a negative correlation;
PITCH_HITS_WALK = IMP_TEAM_PITCHING_H + TEAM_PITCHING_BB

It should be noted that one variable was eventually removed from the model, TEAM_BATTING_2B. The reason was that whenever this variable made it into a model it showed a negative correlation. This would mean whenever a team hit more doubles during a game they had a lower chance of winning! This is completely inappropriate as getting a hit or double in baseball always gives the batting team a better chance of scoring and winning, not a worst chance.

Finally, since some of the observations were normalized to equal a 162 game season, the number of TARGET_WINS has some outliers which are very unrealistic and have never occurred in the history of baseball. For instance, the minimum and maximum number of wins for a professional baseball team over an entire season is 25 and 123 wins respectively. Thus, the decision was made to trim the records to include only records within 25 to 123 wins in an entire season. This trimming of the TARGET_WINS is applied to our test model in our scored file (SCOREFILE).

BUILD MODEL

After imputing the missing values, trimming the variables, transforming the variables, and creating new variables, I created three different models to compare. Each model begins with different variables that I selected manually based on my experience with transforming and creating new variables. Each model is then run through original, forward, backward, and stepwise variable selection processes to produce the final variables. In each case, the results are the same if forward, backward, or stepwise variable selection was used. Each model is listed below along with the statistical results and reasoning why I selected Model #2 as the best model.

MODEL #1 for TEMPFILE3
This model has ADJ-Rsqr=.297, AIC=10204, Model DF=8.

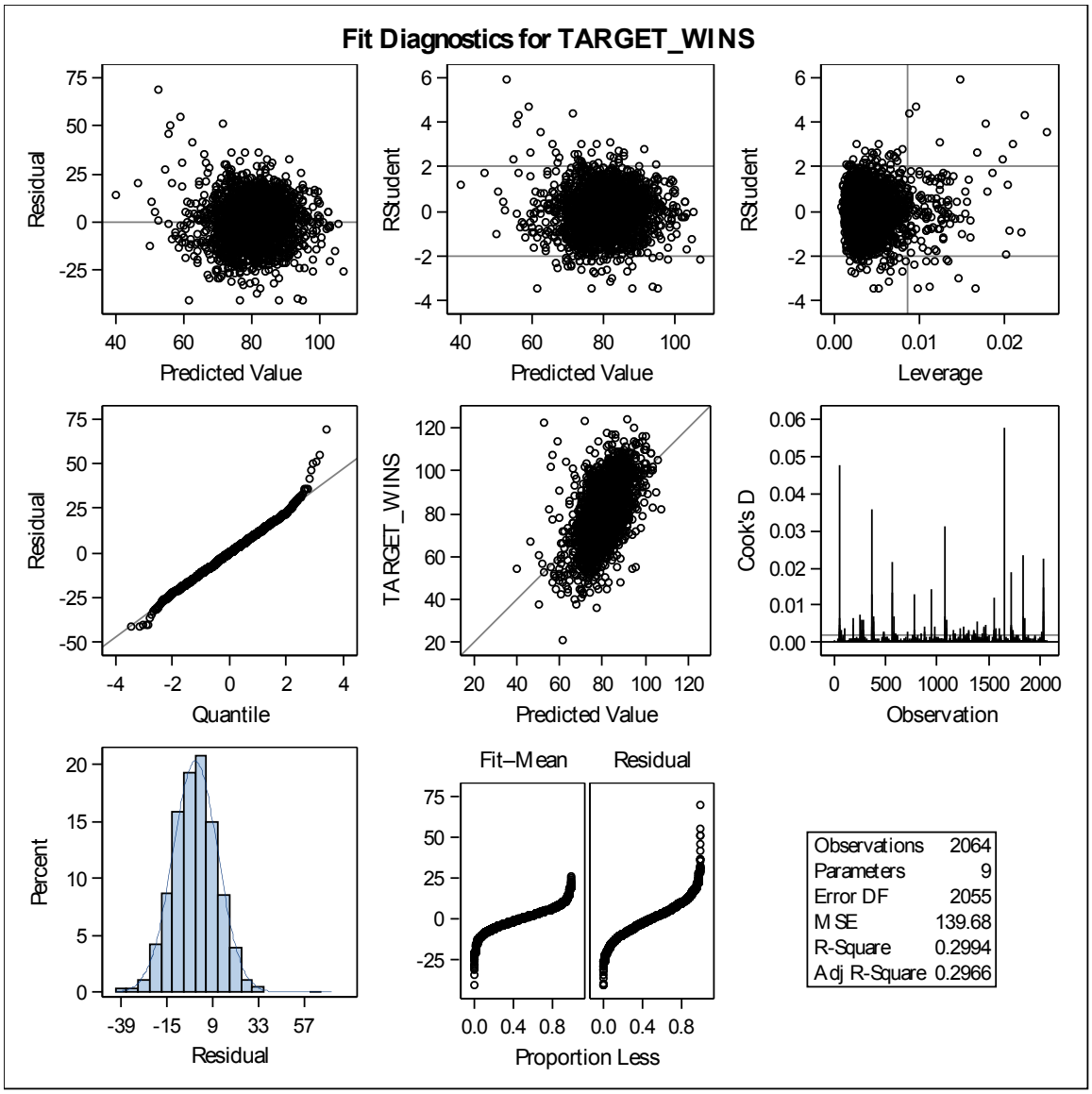
This model has lower ADJ-Rsqr and AIC than imputed and trimmed variable model, but all VIF values show low levels of multicollinearlity, every variable is significant, and, finally, every coefficient sign (pos/neg) is appropriate!

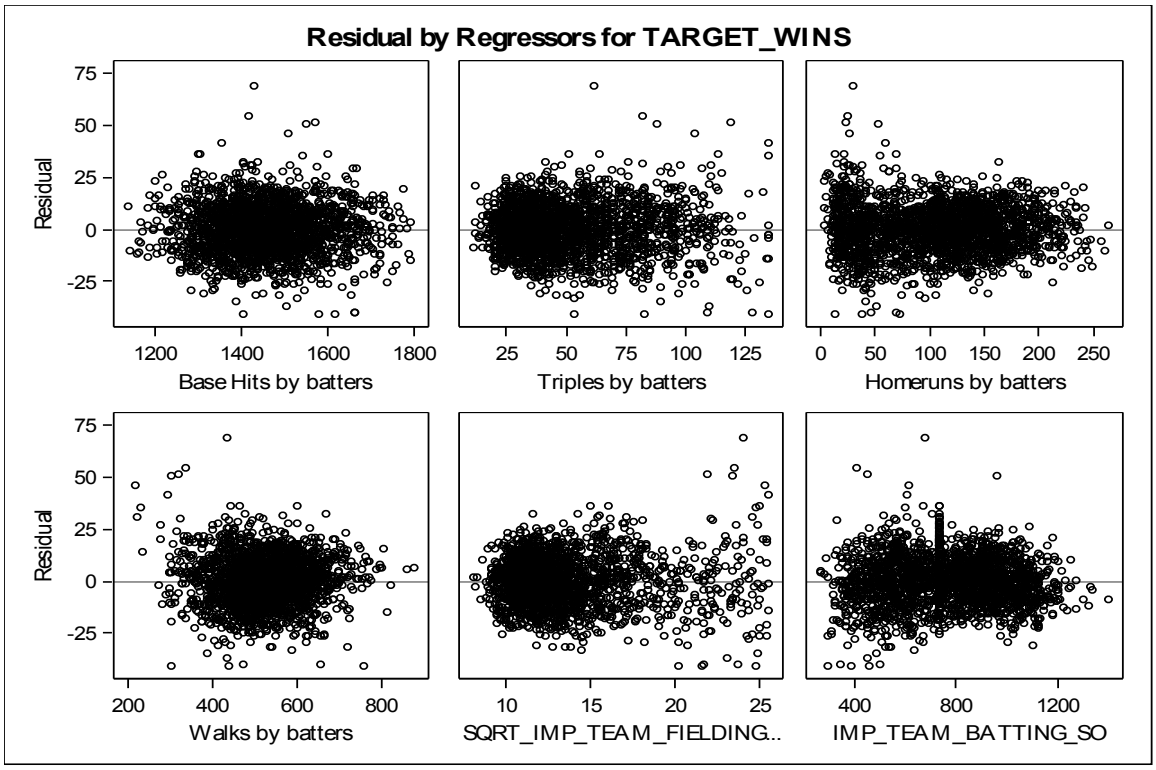
TEAM_BATTING_H
TEAM_BATTING_3B
TEAM_BATTING_HR
TEAM_BATTING_BB
SQRT_IMP_TEAM_FIELDING_E
IMP_TEAM_BATTING_SO
LOG10_TEAM_BASERUN_CS
IMP_TEAM_BASERUN_SB

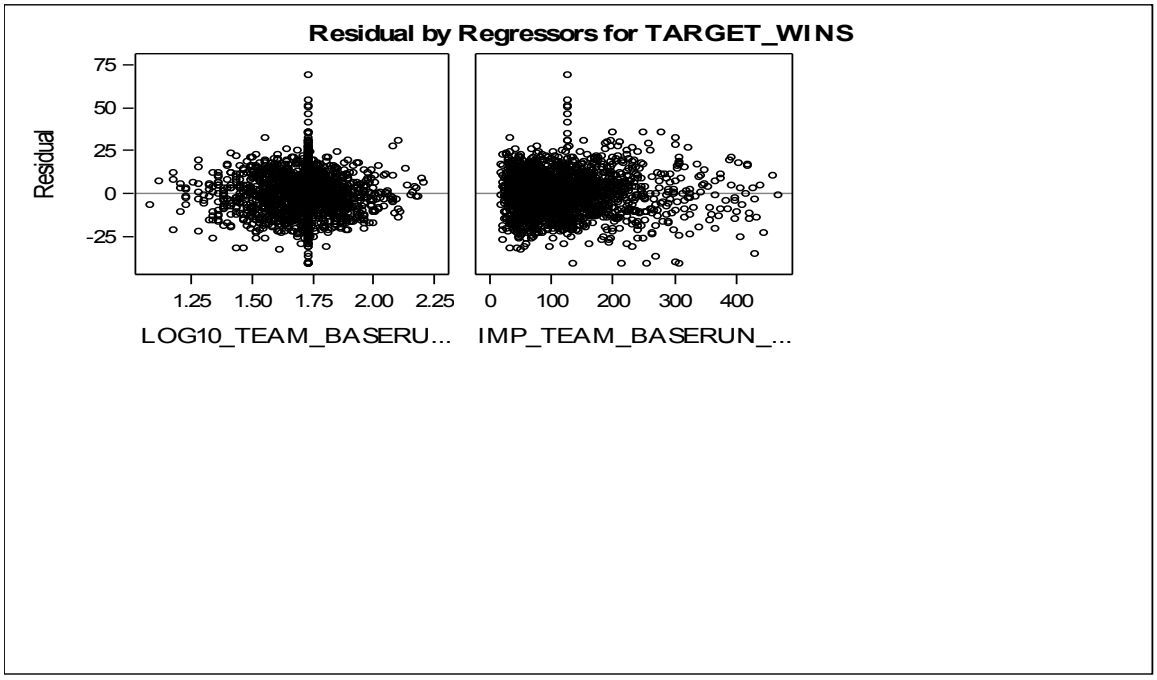
Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	8	122641	15330	109.75	<.0001
Error	2055	287035	139.67646		
Corrected Total	2063	409676			

Root MSE	11.81848	**R-Square**	0.2994
Dependent Mean	80.91521	**Adj R-Sq**	0.2966
Coeff Var	14.60600		

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	76.77134	7.27588	10.55	<.0001	0
TEAM_BATTING_H	Base Hits by batters	1	0.01682	0.00332	5.06	<.0001	2.07521
TEAM_BATTING_3B	Triples by batters	1	0.19781	0.01949	10.15	<.0001	3.46274
TEAM_BATTING_HR	Homeruns by batters	1	0.06354	0.00974	6.52	<.0001	4.74546
TEAM_BATTING_BB	Walks by batters	1	0.01994	0.00328	6.07	<.0001	1.32071
SQRT_IMP_TEAM_FIELDING_E		1	-2.31560	0.15036	-15.40	<.0001	3.88281
IMP_TEAM_BATTING_SO		1	-0.01392	0.00232	-6.00	<.0001	3.83623
LOG10_TEAM_BASERUN_CS		1	-9.14325	2.19778	-4.16	<.0001	1.37819
IMP_TEAM_BASERUN_SB		1	0.08017	0.00537	14.93	<.0001	2.40452


Fit Diagnostics for TARGET_WINS
Residual
Predicted Value
RStudent
Predicted Value
RStudent
Leverage
Residual
Quantile
TARGET_WINS
Predicted Value
Cook's D
Observation
Percent
Residual
Fit–Mean
Residual
Proportion Less
Observations 2064
Parameters 9
Error DF 2055
MSE 139.68
R-Square 0.2994
Adj R-Square 0.2966


Residual by Regressors for TARGET_WINS
Residual
75
50
25
0
-25
1200
1400
1600
1800
Base Hits by batters
25
50
75
100
125
Triples by batters
0
50
100
150
200
250
Homeruns by batters
200
400
600
800
Walks by batters
10
15
20
25
SQRT_IMP_TEAM_FIELDING..
400
800
1200
IMP_TEAM_BATTING_SO


Residual by Regressors for TARGET_WINS
Residual
75
50
25
0
-25
1.25
1.50
1.75
2.00
2.25
LOG10_TEAM_BASERU...
0
100
200
300
400
IMP_TEAM_BASERUN_...

AIC
10203.75

MODEL #2 for TEMPFILE3
This model has ADJ-Rsqr=.3021, AIC=10188, Model DF=8.

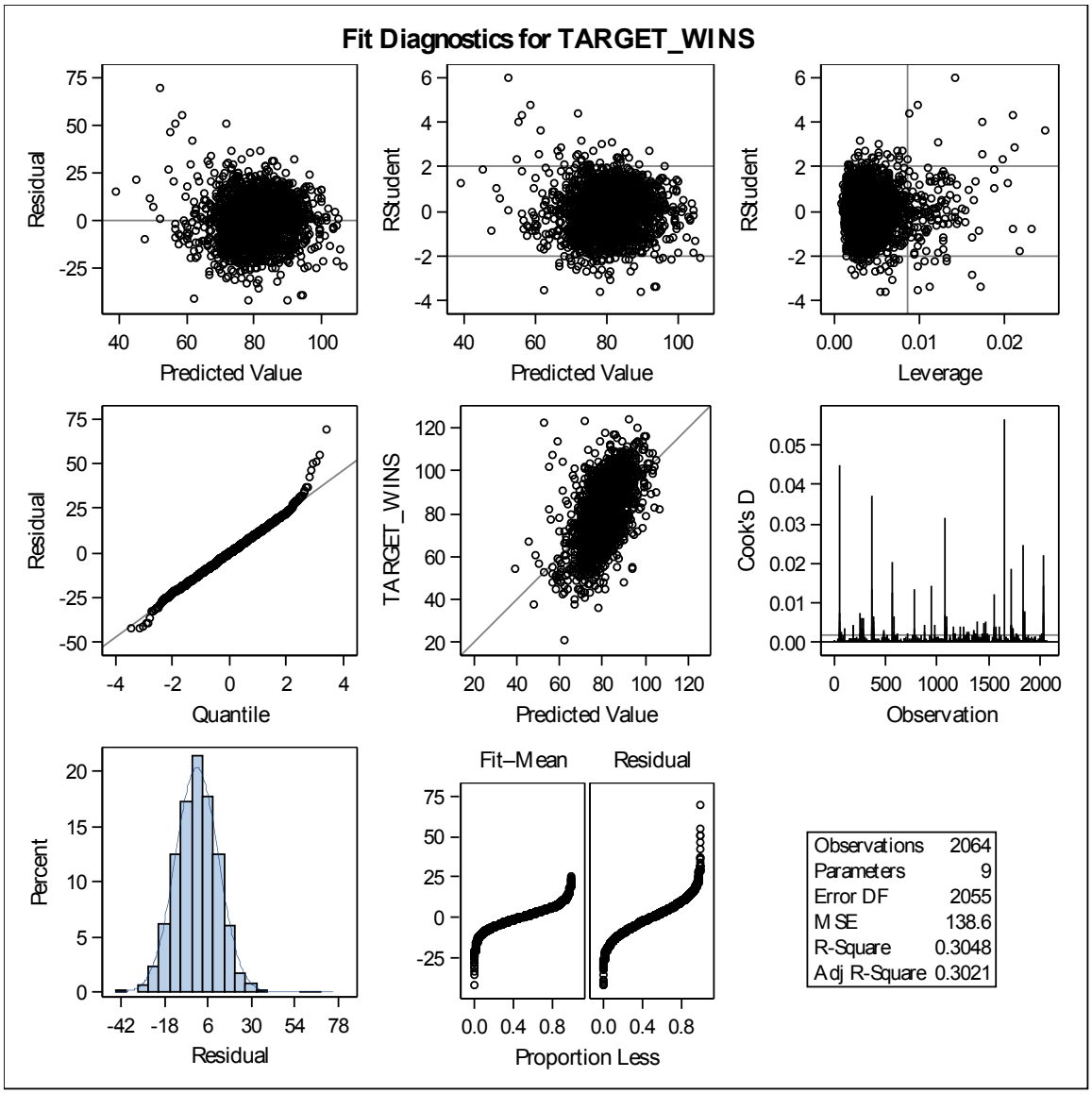
It improves upon Model #1 in terms of Adj R-Sqr and AIC while still maintaining the same number of variables in the model after forward, backward, or stepwise selection processes. The main difference in this model is the addition of two combined variables: H_2B and SINGLES. The variable H_2B does not make it into the model after the variable selection process, but every VIF number is below 5 and every variable has the appropriate coefficient signs.

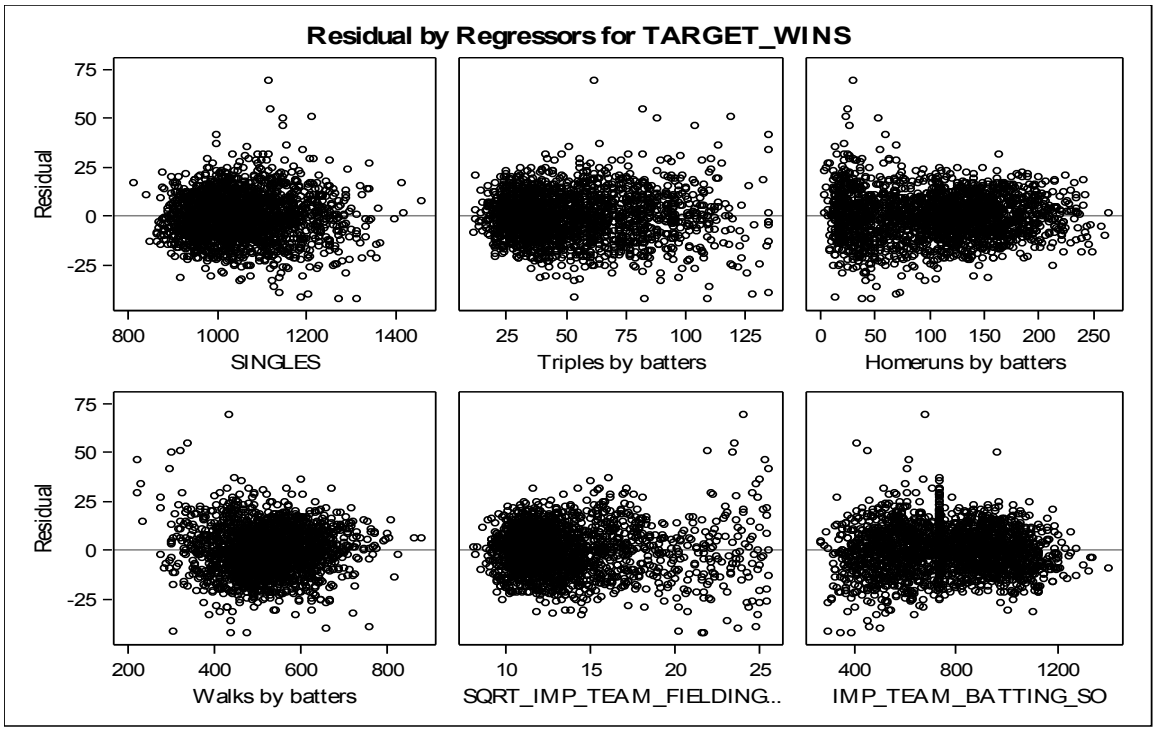
H_2B
SINGLES
TEAM_BATTING_3B
TEAM_BATTING_HR
TEAM_BATTING_BB
SQRT_IMP_TEAM_FIELDING_E
IMP_TEAM_BATTING_SO
LOG10_TEAM_BASERUN_CS
IMP_TEAM_BASERUN_SB

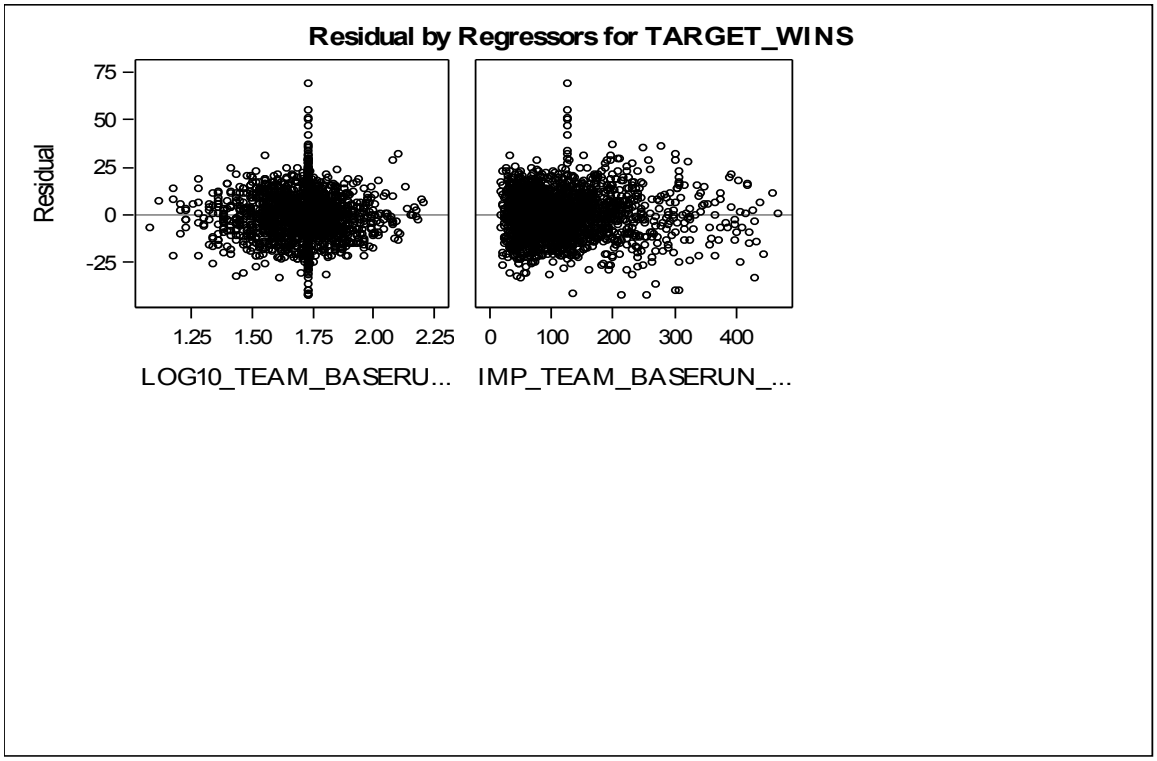
Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	8	124860	15608	112.61	<.0001
Error	2055	284816	138.59666		
Corrected Total	2063	409676			

Root MSE	11.77271	**R-Square**	0.3048
Dependent Mean	80.91521	**Adj R-Sq**	0.3021
Coeff Var	14.54944		

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	67.07738	7.59149	8.84	<.0001	0
SINGLES		1	0.02776	0.00429	6.47	<.0001	2.52492
TEAM_BATTING_3B	Triples by batters	1	0.21505	0.01808	11.89	<.0001	3.00290
TEAM_BATTING_HR	Homeruns by batters	1	0.08518	0.00801	10.63	<.0001	3.23481
TEAM_BATTING_BB	Walks by batters	1	0.02085	0.00327	6.37	<.0001	1.32121
SQRT_IMP_TEAM_FIELDING_E		1	-2.37560	0.14858	-15.99	<.0001	3.82121
IMP_TEAM_BATTING_SO		1	-0.01129	0.00239	-4.71	<.0001	4.12413
LOG10_TEAM_BASERUN_CS		1	-9.03036	2.18469	-4.13	<.0001	1.37242
IMP_TEAM_BASERUN_SB		1	0.07859	0.00536	14.65	<.0001	2.41743


Fit Diagnostics for TARGET_WINS
Residual
Predicted Value
RStudent
Predicted Value
RStudent
Leverage
Residual
Quantile
TARGET_WINS
Predicted Value
Cook's D
Observation
Percent
Residual
Fit–Mean
Residual
Proportion Less
Observations 2064
Parameters 9
Error DF 2055
MSE 138.6
R-Square 0.3048
Adj R-Square 0.3021


Residual by Regressors for TARGET_WINS
Residual
75
50
25
0
-25
800
1000
1200
1400
SINGLES
25
50
75
100
125
Triples by batters
0
50
100
150
200
250
Homeruns by batters
200
400
600
800
Walks by batters
10
15
20
25
SQRT_IMP_TEAM_FIELDING..
400
800
1200
IMP_TEAM_BATTING_SO


Residual by Regressors for TARGET_WINS
Residual
75
50
25
0
-25
1.25
1.50
1.75
2.00
2.25
LOG10_TEAM_BASERU...
0
100
200
300
400
IMP_TEAM_BASERUN_...

AIC
10187.74

MODEL #3 for TEMPFILE3
This model has ADJ-Rsqr=.299, AIC=10197, Model DF=8.

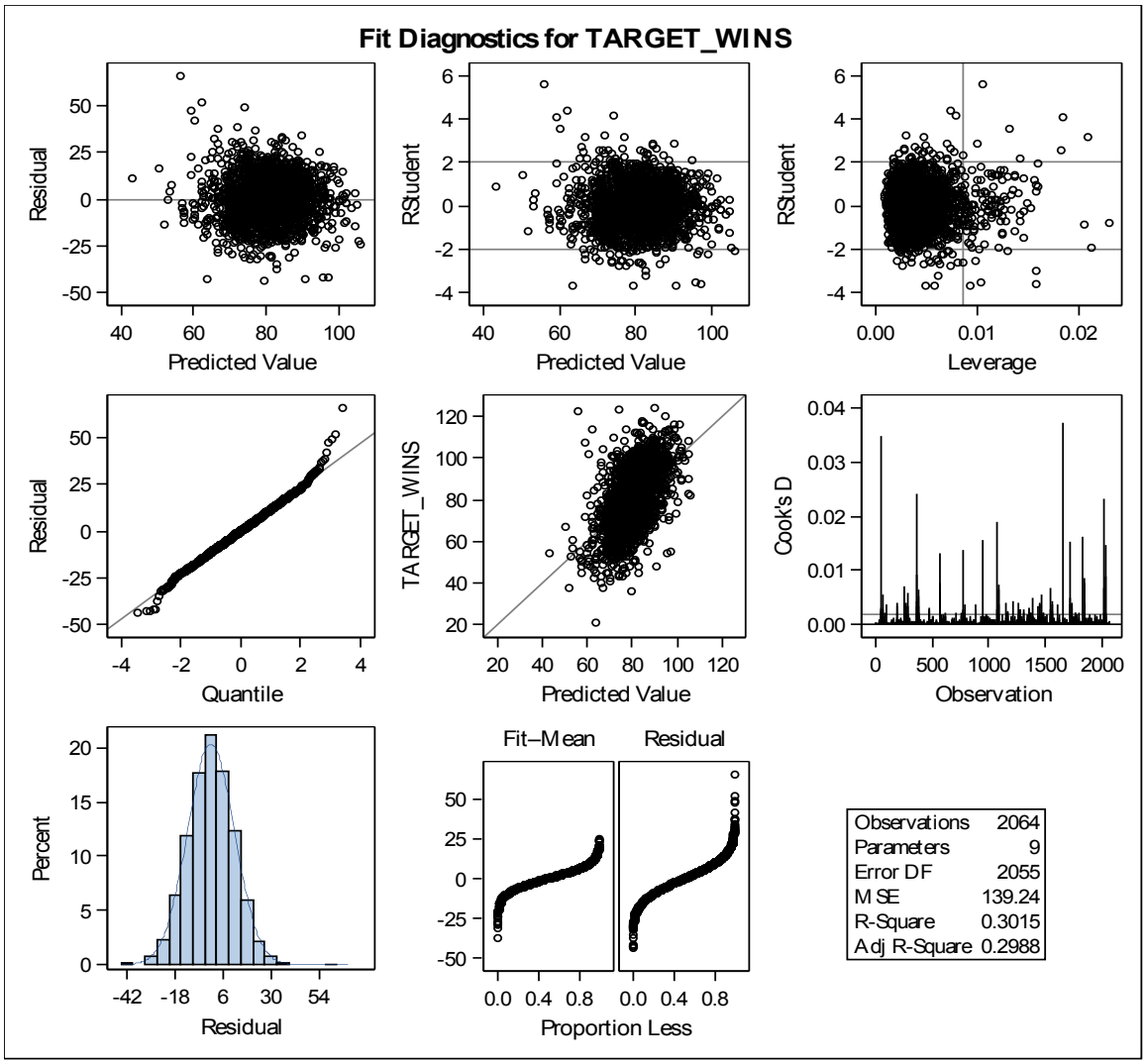
This model uses the logarithm of FIELDING_E (errors) instead of squaring the imputed variable. It also combines PITCH_HITS_WALK = PITCHING_H + PITCHING_BB to see if any additional information can be extracted into the model.

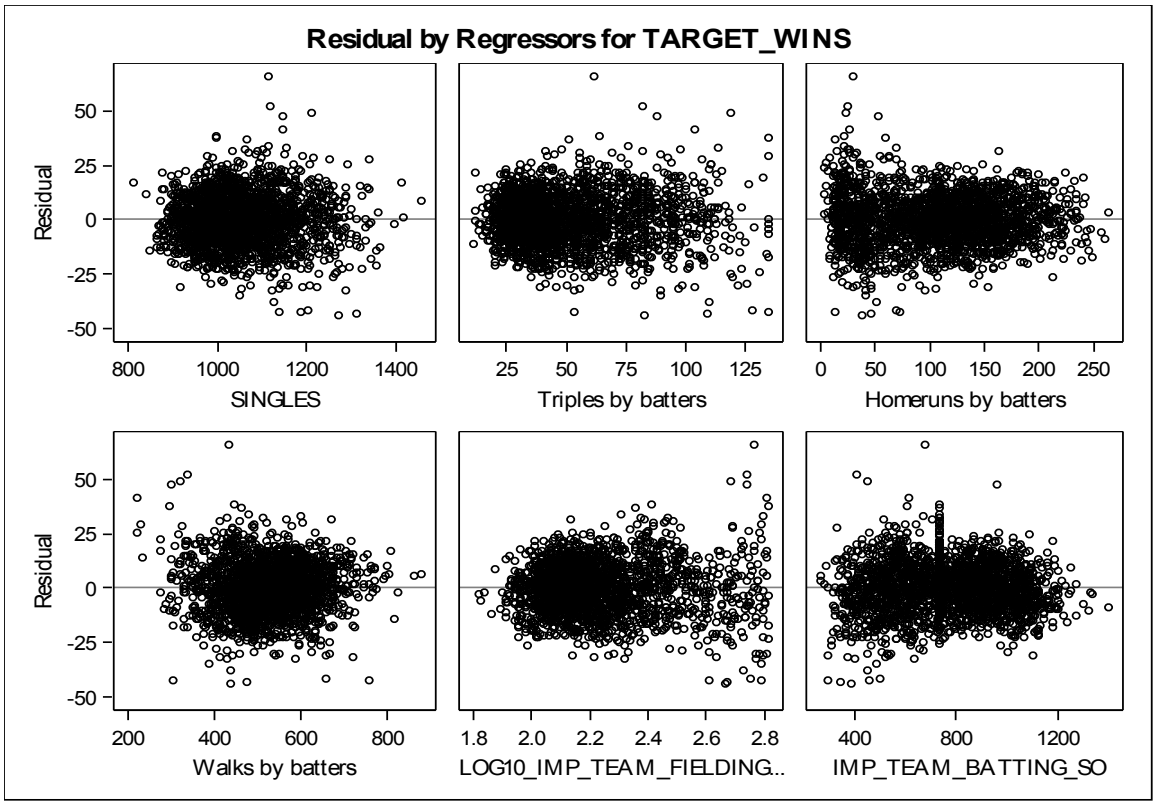
It is close to Model #2 in terms of Adj R-Sqr and AIC while still maintaining the same number of variables in the model, but isn't quite as good. The use of a logarithmic transformation of the FIELDING_E (errors) variable was still effective and used in the model, but the new combined variable, PITCH_HITS_WALK, was still ineffective at extracting more information into the model and did not make the final variable selection process. Still, every VIF number is below 5 and every variable has the appropriate coefficient signs.

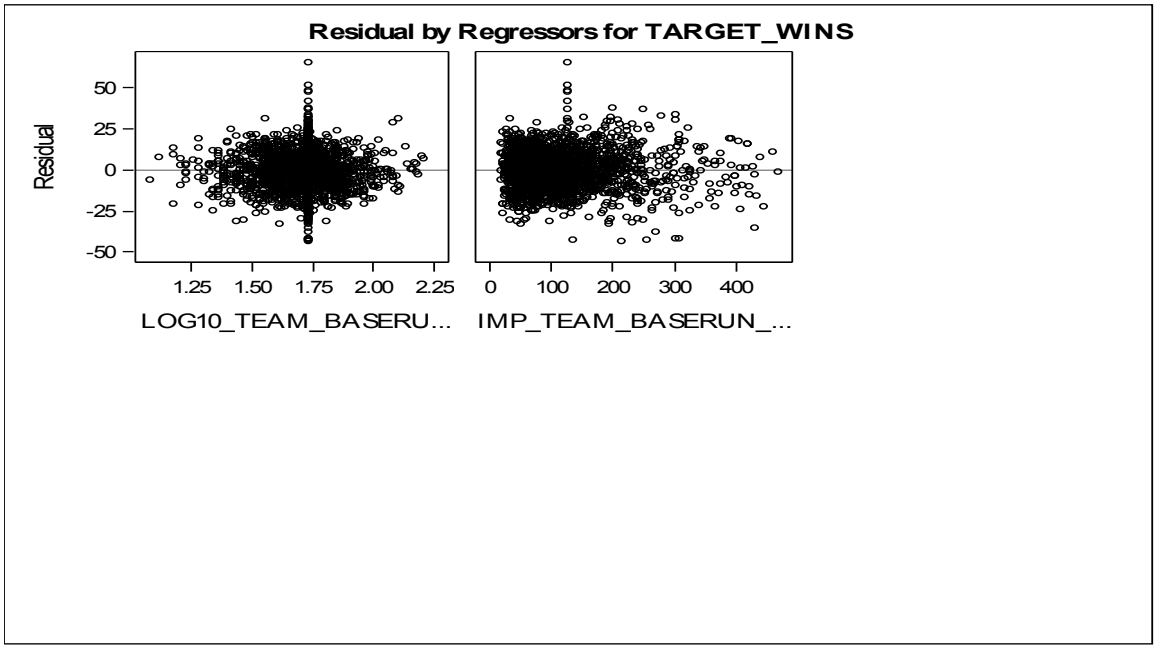
Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	8	123529	15441	110.89	<.0001
Error	2055	286148	139.24458		
Corrected Total	2063	409676			

Root MSE	11.80019	**R-Square**	0.3015
Dependent Mean	80.91521	**Adj R-Sq**	0.2988
Coeff Var	14.58341		

Parameter Estimates							
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	Intercept	1	128.03920	10.07982	12.70	<.0001	0
SINGLES		1	0.02765	0.00430	6.43	<.0001	2.52546
TEAM_BATTING_3B	Triples by batters	1	0.21615	0.01817	11.90	<.0001	3.01933
TEAM_BATTING_HR	Homeruns by batters	1	0.07693	0.00818	9.40	<.0001	3.35762
TEAM_BATTING_BB	Walks by batters	1	0.02128	0.00328	6.50	<.0001	1.31900
LOG10_IMP_TEAM_FIELDING_E		1	-41.82319	2.67263	-15.65	<.0001	3.99976
IMP_TEAM_BATTING_SO		1	-0.01181	0.00241	-4.90	<.0001	4.15834
LOG10_TEAM_BASERUN_CS		1	-7.44036	2.16550	-3.44	0.0006	1.34215
IMP_TEAM_BASERUN_SB		1	0.07023	0.00513	13.70	<.0001	2.19826


Fit Diagnostics for TARGET_WINS
Residual
Predicted Value
RStudent
Predicted Value
RStudent
Leverage
Residual
Quantile
TARGET_WINS
Predicted Value
Cook's D
Observation
Percent
Residual
Fit–Mean
Residual
Proportion Less
Observations 2064
Parameters 9
Error DF 2055
MSE 139.24
R-Square 0.3015
Adj R-Square 0.2988


Residual by Regressors for TARGET_WINS
Residual
50
25
0
-25
-50
800
1000
1200
1400
SINGLES
25
50
75
100
125
Triples by batters
0
50
100
150
200
250
Homeruns by batters
200
400
600
800
Walks by batters
1.8
2.0
2.2
2.4
2.6
2.8
LOG10_IMP_TEAM_FIELDING...
400
800
1200
IMP_TEAM_BATTING_SO


Residual by Regressors for TARGET_WINS
Residual
50
25
0
-25
-50
1.25
1.50
1.75
2.00
2.25
LOG10_TEAM_BASERU...
0
100
200
300
400
IMP_TEAM_BASERUN_...

AIC
10197.36

SELECT MODELS

The final three models were fairly close in terms of Adj- R^2 value and AIC values. I used both of these metrics to determine which model was better suited, as both metrics account for differences in the number of variables in each model and reward parsimony. I also looked at the number of variables in the model, the existence of multicollineariy through VIF values, the appropriateness of the coefficient signs, and anything that would allow one model to be easier to use than the others.

- In terms of parsimony, all three models used the same number of variables.
- Every model had VIF values below 5 for every variable indicating little multicollinearity between variables in each model
- Every coefficient has a positive or negative sign that is appropriate for the associated variable
- Models #2 and #3 both require two logarithmic or squared variable transformations and two created variables, while Model #1 does not require the two created variables. This would make Model #1 slightly easier to use.

Ultimately, I selected Model #2. It has the best Adj- R^2 and AIC values of the three while maintaining parsimony, the absence of multicollinearity, appropriate coefficients, and is only slightly harder to use than Model #1.

CONCLUSION

According to the OLS Regression model, the dataset, and the variables that we used, a professional baseball general or team manager should pay particular attention to these variables when making decisions to improve the team's chances of achieving a win.

Number of singles
Number of triples
Number of home runs
Number of walks achieved
Number of fielding errors
Number of strikeouts by batters
Number of times caught stealing
Number of bases stolen

All of these variables make sense intuitively. Singles, triples, homeruns, walks achieved, and stolen bases all help a team get close to scoring a run, which is the point of the game.

Additionally, striking out a lot as a hitter, getting caught stealing, and having a poor fielding team (errors) gives the opposing team advantages.

My main concern with this model is the lack of defense related variables that made it into the model, specifically those related to pitching. For example, the model shows that having a good offensive hitting team is important (note the variables singles, triples, home runs, walks, and their positive associations with wins). But it is well known that having a good pitching staff (the defensive side of a baseball game) will reduce those same offensive variables. Thus, I was surprised that none of the pitching variables (BB, H, HR, SO) made it into the final model.

Overall, the model is relative and appropriate for baseball predictions. However, for future models, adding in the most common pitching metric used in baseball games, Earned Run Average, might further improve the model as more defensive statistics will be included as well.

**********BINGO BONUS**********

PROG GLM and GENMOD (20 points)

The PROC GLM regression on the best model produces the exact same results as PROC REG.

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	124860.0179	15607.5022	112.61	<.0001
Error	2055	284816.1444	138.5967		
Corrected Total	2063	409676.1623			

R-Square	Coeff Var	Root MSE	TARGET_WINS Mean
0.304777	14.54944	11.77271	80.91521

Parameter	Estimate	Standard Error	t Value	Pr > |t|
Intercept	67.07738200	7.59149248	8.84	<.0001
SINGLES	0.02775563	0.00429032	6.47	<.0001
TEAM_BATTING_3B	0.21504589	0.01807871	11.89	<.0001
TEAM_BATTING_HR	0.08518180	0.00801132	10.63	<.0001
TEAM_BATTING_BB	0.02084614	0.00327091	6.37	<.0001
SQRT_IMP_TEAM_FIELDI	-2.37560444	0.14858143	-15.99	<.0001
IMP_TEAM_BATTING_SO	-0.01128880	0.00239498	-4.71	<.0001

Parameter	Estimate	Standard Error	t Value	Pr > |t|
LOG10_TEAM_BASERUN_C	-9.03036389	2.18468603	-4.13	<.0001
IMP_TEAM_BASERUN_SB	0.07858711	0.00536472	14.65	<.0001

PROC GENMOD

The PROC GENMOD regression on the best model yielded similar results in terms of coefficients, their significance, and the appropriate sign for the coefficients. I did not see an Adj R^2 value but the AIC value was 16047, which is much higher than that produced with PROC REG, and the only real glaring difference.

Model Information	
Data Set	WORK.TEMPFILE3
Distribution	Normal
Link Function	Identity
Dependent Variable	TARGET_WINS

Criteria For Assessing Goodness Of Fit			
Criterion	**DF**	**Value**	**Value/DF**
Deviance	2055	284816.1444	138.5967
Scaled Deviance	2055	2064.0000	1.0044
Pearson Chi-Square	2055	284816.1444	138.5967
Scaled Pearson X2	2055	2064.0000	1.0044

Criteria For Assessing Goodness Of Fit			
Criterion	**DF**	**Value**	**Value/DF**
Log Likelihood		-8013.5575	
Full Log Likelihood		-8013.5575	
AIC (smaller is better)		16047.1150	
AICC (smaller is better)		16047.2221	
BIC (smaller is better)		16103.4390	

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	**DF**	**Estimate**	**Standard Error**	**Wald 95% Confidence Limits**		**Wald Chi-Square**	**Pr > ChiSq**
Intercept	1	67.0774	7.5749	52.2308	81.9240	78.41	<.0001
SINGLES	1	0.0278	0.0043	0.0194	0.0361	42.04	<.0001
TEAM_BATTING_3B	1	0.2150	0.0180	0.1797	0.2504	142.11	<.0001
TEAM_BATTING_HR	1	0.0852	0.0080	0.0695	0.1008	113.55	<.0001
TEAM_BATTING_BB	1	0.0208	0.0033	0.0144	0.0272	40.80	<.0001
SQRT_IMP_TEAM_FIELDI	1	-2.3756	0.1483	-2.6662	-2.0850	256.75	<.0001
IMP_TEAM_BATTING_SO	1	-0.0113	0.0024	-0.0160	-0.0066	22.31	<.0001
LOG10_TEAM_BASERUN_C	1	-9.0304	2.1799	-13.3029	-4.7578	17.16	<.0001
IMP_TEAM_BASERUN_SB	1	0.0786	0.0054	0.0681	0.0891	215.53	<.0001
Scale	1	11.7470	0.1828	11.3941	12.1109		

MACROS (10 points)

```
* creating the macro for data set;
%let PATH = /home/derekhughes2014/DATAFILES/;
%let NAME = mydata;
%let LIB = &NAME..;

libname &NAME. "&PATH." access=readonly;

%let INFILE = &LIB.moneyball;
%let INFILE2 = &LIB.moneyball_test;

%let TEMPFILE1 = TEMPFILE1;
%let TEMPFILE2 = TEMPFILE2;
%let TEMPFILE3 = TEMPFILE3;
```

SCORED DATASET as SAS file (10 points)

```
* code to store scored code into my SAS folder Assignments

libname scorelib "/home/derekhughes2014/Assignments";
data scorelib.DEREK_FILE_moneyball_test;
                set SCOREFILE;
run;
```

SAS CODE FOR ASSIGNMENT:

```
* creating the macro for data set;
%let PATH = /home/derekhughes2014/DATAFILES/;
%let NAME = mydata;
%let LIB = &NAME..;

libname &NAME. "&PATH." access=readonly;

%let INFILE = &LIB.moneyball;
%let INFILE2 = &LIB.moneyball_test;

%let TEMPFILE1 = TEMPFILE1;
%let TEMPFILE2 = TEMPFILE2;
%let TEMPFILE3 = TEMPFILE3;
```

```
********** EDA - discovering means, medians, and variance ***********;

* displaying a few observations for reference;
	proc print data=&INFILE.(obs=10);
	TITLE 'Moneyball Data Output';
	run;

* searching contents of data for EDA;
	proc contents data=&INFILE.(obs=10);
	run;

* investigating correlation between target_wins and other variables;
ods graphics on;
	PROC CORR DATA=&INFILE. plot=matrix(histogram nvar=all);
	TITLE2 "PROC CORR: Investigating variable correlation vs target variable";
	run;
ods graphics off;

	TITLE ; * clear title;
	TITLE2 ; * clear title;

* exploring data with proc means;
	proc means data=&INFILE. nmiss mean mode p1 p50 p99 stddev min max;
	TITLE3 "Means - Full Model (no modifications)";
	run;

	Title3 ; * clear title;

* change to variables in &INFILE;
	proc univariate data=&INFILE. plot;
	title4 "Univariate Full Model - analyzing all variables";
	histogram TARGET_WINS
			TEAM_BATTING_H
			TEAM_BATTING_2B
			TEAM_BATTING_3B
			TEAM_BATTING_HR
			TEAM_BATTING_BB
			TEAM_BATTING_HBP
			TEAM_BATTING_SO
			TEAM_BASERUN_SB
			TEAM_BASERUN_CS
			TEAM_FIELDING_E
			TEAM_FIELDING_DP
```

```
                        TEAM_PITCHING_BB
                        TEAM_PITCHING_H
                        TEAM_PITCHING_HR
                        TEAM_PITCHING_SO ;
        run;

        TITLE4 ; * clear title;

* OLS REGRESSION on full model with no adjustments***************
*       This model has ADJ-Rsqr=.53, AIC=821, Model DF=7....
*       However, it only uses 191 of 2276 observations!
*       This is mainly due to variable TEAM_BATTING_HBP only having values in
10% of total observations;

        proc reg data=&INFILE. outset=results;
        model TARGET_WINS = TEAM_BATTING_H
                                        TEAM_BATTING_2B
                                        TEAM_BATTING_3B
                                        TEAM_BATTING_HR
                                        TEAM_BATTING_BB
                                        TEAM_BATTING_HBP
                                        TEAM_BATTING_SO
                                        TEAM_BASERUN_SB
                                        TEAM_BASERUN_CS
                                        TEAM_FIELDING_E
                                        TEAM_FIELDING_DP
                                        TEAM_PITCHING_BB
                                        TEAM_PITCHING_H
                                        TEAM_PITCHING_HR
                                        TEAM_PITCHING_SO /
selection=stepwise vif aic;
        title5 "INFILE REGRESSION - FULL MODEL #1";
        run;

proc print data=results;
run;

        title5 ; *clear title;


* BEGIN TO FIX MISSING VALUES USING NEW DATA FILE;

*********************************************;
*       TEMPFILE1 - data set with missing
*       values imputed
*********************************************;
```

```
data &TEMPFILE1.;
	set &INFILE.;

*remove hits by pitch for batting variable because only;
	* has values in approximately 10% of total observations!;
	drop TEAM_BATTING_HBP;

* using the median for TEAM_PITCHING_SO missing values;
	IMP_TEAM_PITCHING_SO = TEAM_PITCHING_SO;
	if missing(IMP_TEAM_PITCHING_SO) then IMP_TEAM_PITCHING_SO =
818;
	drop TEAM_PITCHING_SO;

* using the mean for missing values for TEAM_BATTING_SO;
	IMP_TEAM_BATTING_SO = TEAM_BATTING_SO;
	if missing(TEAM_BATTING_SO) then IMP_TEAM_BATTING_SO = 736;
	drop TEAM_BATTING_SO;

* using the mean for missing values for TEAM_BASERUN_CS;
	IMP_TEAM_BASERUN_CS = TEAM_BASERUN_CS;
	if missing(TEAM_BASERUN_CS) then IMP_TEAM_BASERUN_CS = 53;
	drop TEAM_BASERUN_CS;

* using the mean for missing values for TEAM_BASERUN_SB;
	IMP_TEAM_BASERUN_SB = TEAM_BASERUN_SB;
	if missing(TEAM_BASERUN_SB) then IMP_TEAM_BASERUN_SB = 125;
	drop TEAM_BASERUN_SB;

* using the mean for missing values for TTEAM_FIELDING_DP;
	IMP_TEAM_FIELDING_DP = TEAM_FIELDING_DP;
	if missing(TEAM_FIELDING_DP) then IMP_TEAM_FIELDING_DP = 146;
	drop TEAM_FIELDING_DP;

run;



* OLS REGRESSION on TEMPFILE1 - which includes imputed missing values and
remove HBP....**********
*	This model has ADJ-Rsqr=.31, AIC=11505, Model DF=11....
*	However, two values show high correlation with other variables due to high VIF
values.
```

```
*       Furthermore, IMP_TEAM_BATTING_SO, IMP_TEAM_BATTING_2B,
IMP_TEAM_FIELDING_DP, TEAM_PITCHING_BB
*       have improper coefficients in the model (should have opposite positive/negative
correlations);

        ods graphics on;
                proc reg data=&TEMPFILE1. outest=results1;
                model TARGET_WINS = TEAM_BATTING_H
                                    TEAM_BATTING_2B
                                    TEAM_BATTING_3B
                                    TEAM_BATTING_HR
                                    TEAM_BATTING_BB
                                    IMP_TEAM_BATTING_SO

        IMP_TEAM_BASERUN_SB

        IMP_TEAM_BASERUN_CS
                                    TEAM_FIELDING_E
                                    IMP_TEAM_FIELDING_DP

                                    TEAM_PITCHING_H
                                    TEAM_PITCHING_BB
                                    TEAM_PITCHING_HR

        IMP_TEAM_PITCHING_SO / selection=stepwise vif aic;
                title6 "TEMPFILE1 REGRESSION - Missing Values Imputed,
HBP removed";
            run;
        ods graphics off;

        proc print data=results1;
        run;

        title6 ; * clear title;


************************************************************
*       TEMPFILE2 - uses imputed data set with missing values
*           with trimmed variables to handle outliers
************************************************************;

data &TEMPFILE2.;
        set &TEMPFILE1.;

****    Trimming varibles   & removing outliers    ****;
```

```
* trimming top end outliers for TEAM_PITCHING_H;
        IMP_TEAM_PITCHING_H = TEAM_PITCHING_H;
        if TEAM_PITCHING_H > 2300 then delete;
        drop TEAM_PITCHING_H;

* trimming top end outliers if TEAM_FIELDING_E over 650;
        IMP_TEAM_FIELDING_E = TEAM_FIELDING_E;
        if TEAM_FIELDING_E > 650 then delete;
        drop TEAM_FIELDING_E;

* trimming low end values for TEAM_BATTING_3B...THIS ENDED UP REDUCING
ADJ-R_SQR;
        if TEAM_BATTING_3B > 134 then TEAM_BATTING_3B = 135;

* trimming values for IMP_TEAM_BATTING_SO to help VIF...
*               by deleting lower end, it improved ADJ Rsqr and VIF;
        if IMP_TEAM_BATTING_SO <= 162 then delete;
        else if IMP_TEAM_BATTING_SO > 1600 then IMP_TEAM_BATTING_SO =
1600;

* trimming values for IMP_TEAM_PITCHING_SO to help VIF...deleting these values
*               instead of having them equal the lower/upper limits improved ADJ-Rsqr;
        if IMP_TEAM_PITCHING_SO <= 300 then delete;
        else if IMP_TEAM_PITCHING_SO > 1600 then delete;

* trimming top end values for TEAM_BATTING_H;
        if TEAM_BATTING_H > 1800 then delete;

* trimming topend values for IMP_TEAM_BASERUN_CS;
        if IMP_TEAM_BASERUN_CS > 160 then delete;

* trimming top end values for IMP_TEAM_BASERUN_SB;
        if IMP_TEAM_BASERUN_SB > 480 then delete;

* trimming values for TEAM_PITCHING_HR...this had no impact bc variable is
*               first to be removed in backward selection;
*       if TEAM_PITCHING_HR > 264 then TEAM_PITCHING_HR = 265;

* trimming values for TEAM_PITCHING_BB...no real impact because
*               does not make it into stepwise selection;
*       if TEAM_PITCHING_BB < 250 then TEAM_PITCHING_BB = 250;
*       if TEAM_PITCHING_BB > 850 then TEAM_PITCHING_BB = 850;

* trimming values for TEAM_BATTING_3B...THIS TOO ENDED UP REDUCING
ADJ-R_SQR
*       if TEAM_BATTING_3B > 134 then delete;
```

```
* Logarithmic transformation of IMP_TEAM_BASERUN_SB variable;
*	LN_TEAM_BASERUN_SB = sign(IMP_TEAM_BASERUN_SB) *
log(abs(IMP_TEAM_BASERUN_SB)+1);
*	LOG10_TEAM_BASERUN_SB = sign(IMP_TEAM_BASERUN_SB) *
log10(abs(IMP_TEAM_BASERUN_SB)+1);
*	drop IMP_TEAM_BASERUN_SB;

* Logarithmic transformation of TEAM_BATTING_HR variable;
*	LOG10_TEAM_BATTING_3B = sign(TEAM_BATTING_3B) *
log10(abs(TEAM_BATTING_3B)+1);
*	drop TEAM_BATTING_3B;

run;



* OLS REGRESSION on TEMPFILE1 - which includes imputed missing
values....**********
*	This model has ADJ-Rsqr=.37, AIC=9996, Model DF=14....
*	Improved ADJ-Rsqr and AIC but extreme multicollinearity and improper sign
(pos/neg) coefficients
*		for many variables.
*	Massive transformation and/or removing of variables needed to create an a
reliable model;

	ods graphics on;
		proc reg data=&TEMPFILE2. outest=results2;
		model TARGET_WINS = TEAM_BATTING_H
							TEAM_BATTING_2B
							TEAM_BATTING_3B
							TEAM_BATTING_HR
							TEAM_BATTING_BB
							IMP_TEAM_BATTING_SO

	IMP_TEAM_BASERUN_SB

	IMP_TEAM_BASERUN_CS
							IMP_TEAM_FIELDING_E
							IMP_TEAM_FIELDING_DP

							IMP_TEAM_PITCHING_H
							TEAM_PITCHING_BB
							TEAM_PITCHING_HR

	IMP_TEAM_PITCHING_SO / selection=stepwise vif aic;
```

```
                        title7 "TEMPFILE2 REGRESSION - Missing Values Imputed,
Trimming Values, Outliers Removed";
                run;
        ods graphics off;

        proc print data=results2;
        run;

        title7 ; * clear title;



*****************************************************************
*       TEMPFILE3 - data set with missing values imputed, variables
*               trimmed, and variables combined and transformed
*****************************************************************;

data &TEMPFILE3.;
        set &TEMPFILE2.;

* attempt to test combine variables batting_2b and batting_hr b/c correlation .44;
        BAT_2B_HR = TEAM_BATTING_2B * TEAM_BATTING_HR;

* attempt to test combine variables hits*2b b/c correlation H-2b = .56 and H-3B = .42,
but 2b-3b = -.1;
        H_2B_3B = TEAM_BATTING_H * TEAM_BATTING_2B *
TEAM_BATTING_3B;

* attempt to test combine variables hits*2b*3b b/c correlation H-2b = .61;
        H_2B = TEAM_BATTING_H * TEAM_BATTING_2B;

* creating variable single: combining variables of hits-2b-3b-hr
*        to measure if hitting a single (1B) has influence;
        SINGLES = TEAM_BATTING_H - TEAM_BATTING_2B -
TEAM_BATTING_3B - TEAM_BATTING_HR;

* transforming log10 of hits*2b*3b*hr because highly skewed;
        LOG10_H_2B_3B = sign(H_2B_3B) * log10(abs(H_2B_3B)+1);

* Logarithmic transformation of IMP_TEAM_BASERUN_CS variable because skewed;
        LOG10_TEAM_BASERUN_CS = sign(IMP_TEAM_BASERUN_CS) *
log10(abs(IMP_TEAM_BASERUN_CS)+1);
        drop IMP_TEAM_BASERUN_CS;

* Square root transformation of IMP_TEAM_FIELDING_E variable because skewed
```

```
*       ...this raised ADJ Rsqr and lowered AIC vs. Logarithmic transformation;
        SQRT_IMP_TEAM_FIELDING_E = SQRT(IMP_TEAM_FIELDING_E);
*       drop IMP_TEAM_FIELDING_E;

* Logarithmic transformation of IMP_TEAM_FIELDING_E variable because skewed
*       ...good but not as good compared to squaring variable;
        LOG10_IMP_TEAM_FIELDING_E = sign(IMP_TEAM_FIELDING_E) *
log10(abs(IMP_TEAM_FIELDING_E)+1);
        drop IMP_TEAM_FIELDING_E;

* Logarithmic transformation of IMP_TEAM_BASERUN_SB variable...;
*       ...this lowered ADJ Rsqr and raised AIC;
*       LOG10_IMP_TEAM_BASERUN_SB = sign(IMP_TEAM_BASERUN_SB) *
log10(abs(IMP_TEAM_BASERUN_SB)+1);
*       drop IMP_TEAM_BASERUN_SB;

* combining variables pitching hits and walks to attempt to reduce VIF of each...
        * this was eventually removed from model because it had a positive correlation
        * when it should have been a negative correlation;
        PITCH_HITS_WALK = IMP_TEAM_PITCHING_H + TEAM_PITCHING_BB;
        drop IMP_TEAM_PITCHING_H;
        drop TEAM_PITCHING_BB;

* droping IMP_TEAM_FIELDING_DP b/c has negative coefficient when should be
positive;
*       drop IMP_TEAM_FIELDING_DP;

* droping TEAM_BATTING_2B b/c has negative coefficient when should be positive;
        drop TEAM_BATTING_2B;



run;


* OLS REGRESSION on TEMPFILE3 - which includes imputed missing values,
trimmed variables,     **********
*                                                       transformed variables,
dropped variables, and combined variables    ******
*               Results are the same for original, forward, backward, or stepwise variable
selection       ******;


* MODEL #1 - proc reg for TEMPFILE3;
        *       This model has ADJ-Rsqr=.297, AIC=10204, Model DF=8....
```

```
		*	This model has lower ADJ-Rsqr and AIC but all VIF values show low levels of multicollinearlity,
		*	every variable is significant, and every coefficient sign (pos/neg) is appropriate! ;
	ods graphics on;
		proc reg data=&TEMPFILE3. outest=results3;
			*original: model TARGET_WINS = TEAM_BATTING_H TEAM_BATTING_3B TEAM_BATTING_HR TEAM_BATTING_BB SQRT_IMP_TEAM_FIELDING_E IMP_TEAM_BATTING_SO LOG10_TEAM_BASERUN_CS IMP_TEAM_BASERUN_SB / vif;
			*forward:  model TARGET_WINS = TEAM_BATTING_H TEAM_BATTING_3B TEAM_BATTING_HR TEAM_BATTING_BB SQRT_IMP_TEAM_FIELDING_E IMP_TEAM_BATTING_SO LOG10_TEAM_BASERUN_CS IMP_TEAM_BASERUN_SB / selection=forward vif aic;
			*backward: model TARGET_WINS = TEAM_BATTING_H TEAM_BATTING_3B TEAM_BATTING_HR TEAM_BATTING_BB SQRT_IMP_TEAM_FIELDING_E IMP_TEAM_BATTING_SO LOG10_TEAM_BASERUN_CS IMP_TEAM_BASERUN_SB / selection=backward vif aic;
			stepwise: model TARGET_WINS = TEAM_BATTING_H

	TEAM_BATTING_3B

	TEAM_BATTING_HR

	TEAM_BATTING_BB

	SQRT_IMP_TEAM_FIELDING_E

	IMP_TEAM_BATTING_SO

	LOG10_TEAM_BASERUN_CS

	IMP_TEAM_BASERUN_SB / selection=stepwise aic vif;
			title8 "TEMPFILE3 REGRESSION - Model #1 - Variables Transformed and Inappropriate Variables Dropped";
		run;

		title8 ; * clear title;

* MODEL #2 - combining variables H_2B = H*2B and SINGLES = H-2B-3B-HR;
*		This model has ADJ-Rsqr=.302, AIC=10188, Model DF=8....;
		proc reg data=&TEMPFILE3. outest=results4;
			stepwise: model TARGET_WINS = H_2B
```

```
	SINGLES

	TEAM_BATTING_3B

	TEAM_BATTING_HR

	TEAM_BATTING_BB

	SQRT_IMP_TEAM_FIELDING_E

	IMP_TEAM_BATTING_SO

	LOG10_TEAM_BASERUN_CS

	IMP_TEAM_BASERUN_SB / selection=stepwise aic vif;
			title9 "TEMPFILE3 REGRESSION - Model#2 (BEST of the 3) -
with Variables Transformed and Inappropriate Variables Dropped";
		run;



* MODEL #3 - LOG instead of SQRT_FIELDING_Errors, combine
PITCH_HITS_WALK = PITCHING_H + PITCHING_BB;
*		This model has ADJ-Rsqr=.299, AIC=10197, Model DF=8....;
		proc reg data=&TEMPFILE3. outest=results5;
			stepwise: model TARGET_WINS = SINGLES

	TEAM_BATTING_3B

	TEAM_BATTING_HR

	TEAM_BATTING_BB

	LOG10_IMP_TEAM_FIELDING_E

	PITCH_HITS_WALK

	IMP_TEAM_BATTING_SO

	LOG10_TEAM_BASERUN_CS

	IMP_TEAM_BASERUN_SB / selection=stepwise aic vif;
			title9 "TEMPFILE3 REGRESSION - Model#3 - with Variables
Transformed and Variables Created";
		run;
```

```
	ods graphics off;

	proc print data=results3;
	title8 "TEMPFILE3 REGRESSION - Model #1 - Variables Transformed and
Inappropriate Variables Dropped";
	run;
	title8 ; * clear title;

	proc print data=results4;
	title9 "TEMPFILE3 REGRESSION - Model#2 (BEST of the 3) - with Variables
Transformed and Inappropriate Variables Dropped";
	run;
	title9 ; * clear title;

	proc print data=results5;
	title9 "TEMPFILE3 REGRESSION - Model#3 - with Variables Transformed and
Variables Created";

	run;
	title9 ; * clear title;



/*
* investigating correlation between target_wins and other variables;
	ods graphics on;
		PROC CORR DATA=&TEMPFILE3. plot=matrix(histogram nvar=all);
		TITLE6 "PROC CORR: TEMPFILE3 Correlations";
		run;
	ods graphics off;

	Title5 ; * clear title;
	Title6 ; * clear title;


* proc univariate for best model;
	proc univariate data=&TEMPFILE3. plot;
		title7 "UNIVARIATE - After missing value and transformations";
	run;

	title7 ;
*/
```

```
*****************************************************;
**********        SCORING DATA          ***************;
*****************************************************;

***** SCORING data - use &INFILE for testing model and checking out error
(moneyball - dataset);
***** SCORING data - use &INFILE2 to test model on moneyball_test dataset;

data SCOREFILE;
        set &INFILE2.;

*remove hits by pitch for batting variable b/c only has < 10% of total observations;
        drop TEAM_BATTING_HBP;

* trimming top end outliers for TEAM_PITCHING_H;
        IMP_TEAM_PITCHING_H = TEAM_PITCHING_H;
        if TEAM_PITCHING_H > 2300 then delete;
        drop TEAM_PITCHING_H;

* using the median for TEAM_PITCHING_SO missing values;
        IMP_TEAM_PITCHING_SO = TEAM_PITCHING_SO;
*       if IMP_TEAM_PITCHING_SO > 2300 then delete;
        if missing(IMP_TEAM_PITCHING_SO) then IMP_TEAM_PITCHING_SO =
818;
        drop TEAM_PITCHING_SO;

* using the mean for missing values for TEAM_BATTING_SO;
        IMP_TEAM_BATTING_SO = TEAM_BATTING_SO;
        if missing(TEAM_BATTING_SO) then IMP_TEAM_BATTING_SO = 736;
        drop TEAM_BATTING_SO;

* using the mean for missing values for TEAM_BASERUN_CS and removing
observation if over 160;
        IMP_TEAM_BASERUN_CS = TEAM_BASERUN_CS;
        if missing(TEAM_BASERUN_CS) then IMP_TEAM_BASERUN_CS = 53;
        if IMP_TEAM_BASERUN_CS > 160 then delete;
        drop TEAM_BASERUN_CS;

* using the mean for missing values for TEAM_BASERUN_SB and removing
observation if over 480;
        IMP_TEAM_BASERUN_SB = TEAM_BASERUN_SB;
        if missing(TEAM_BASERUN_SB) then IMP_TEAM_BASERUN_SB = 125;
        if IMP_TEAM_BASERUN_SB > 480 then delete;
        drop TEAM_BASERUN_SB;
```

```
* using the mean for missing values for TTEAM_FIELDING_DP;
	IMP_TEAM_FIELDING_DP = TEAM_FIELDING_DP;
	if missing(TEAM_FIELDING_DP) then IMP_TEAM_FIELDING_DP = 146;
	drop TEAM_FIELDING_DP;

* using the mean for missing values for TEAM_FIELDING_E and removing observation
if over 650;
	IMP_TEAM_FIELDING_E = TEAM_FIELDING_E;
	if TEAM_FIELDING_E > 650 then delete;
	drop TEAM_FIELDING_E;

* using the median for TEAM_PITCHING_SO missing values;
	IMP_TEAM_PITCHING_SO = TEAM_PITCHING_SO;
	if missing(IMP_TEAM_PITCHING_SO) then IMP_TEAM_PITCHING_SO =
818;
	drop TEAM_PITCHING_SO;

* using the mean for missing values for TEAM_BATTING_SO;
	IMP_TEAM_BATTING_SO = TEAM_BATTING_SO;
	if missing(TEAM_BATTING_SO) then IMP_TEAM_BATTING_SO = 736;
	drop TEAM_BATTING_SO;

* trimming values for IMP_TEAM_BATTING_SO...
		* by deleting lower end, it improved ADJ Rsqr and VIF;
	if IMP_TEAM_BATTING_SO <= 162 then delete;
	else if IMP_TEAM_BATTING_SO > 1600 then IMP_TEAM_BATTING_SO =
1600;

* trimming values for IMP_TEAM_PITCHING_SO...deleting these values instead;
		* having them equal the lower/upper limits improved ADJ-Rsqr;
	if IMP_TEAM_PITCHING_SO <= 300 then delete;
	else if IMP_TEAM_PITCHING_SO > 1600 then delete;

* trimming values for TEAM_BATTING_H;
	if TEAM_BATTING_H > 1800 then delete;

* trimming values for TEAM_BATTING_3B;
	if TEAM_BATTING_3B > 134 then delete;

* Square root transformation of IMP_TEAM_FIELDING_E variable...;
		* this yielded higher Adj-Rsqr, lower AIC than log transformation;
	SQRT_IMP_TEAM_FIELDING_E = SQRT(IMP_TEAM_FIELDING_E);
	drop IMP_TEAM_FIELDING_E;

* Logarithmic transformation of IMP_TEAM_BASERUN_CS variable;
```

```
        LOG10_TEAM_BASERUN_CS = sign(IMP_TEAM_BASERUN_CS) *
log10(abs(IMP_TEAM_BASERUN_CS)+1);
        drop IMP_TEAM_BASERUN_CS;

* attempt to test combine variables hits*2b*3b b/c correlation H-2b = .61;
        H_2B = TEAM_BATTING_H * TEAM_BATTING_2B;

* creating variable single: combining variables of hits-2b-3b-hr
*         to measure if hitting a single (1B) has influence;
        SINGLES = TEAM_BATTING_H - TEAM_BATTING_2B -
TEAM_BATTING_3B - TEAM_BATTING_HR;


P_TARGET_WINS =
                                67.0774
                        +  0.027756 * SINGLES
                        +  0.21505 * TEAM_BATTING_3B
                        +  0.085182 * TEAM_BATTING_HR
                        +  0.020846 * TEAM_BATTING_BB
                        +  -2.37560 * SQRT_IMP_TEAM_FIELDING_E
                        +  -0.011289 * IMP_TEAM_BATTING_SO
                        +  -9.03036 * LOG10_TEAM_BASERUN_CS
                        +  0.078587 * IMP_TEAM_BASERUN_SB
                                ;

        if P_TARGET_WINS < 25 then P_TARGET_WINS = 25;
        if P_TARGET_WINS > 123 then P_TARGET_WINS = 123;

        keep INDEX;
        keep P_TARGET_WINS;
*       keep TARGET_WINS   - removed when not checking error amounts on
moneyball dataset;

run;


/*
* Testing model - testing error amounts on a file where dependent variable value
*                                 is known (Note: &INFILE - this must be changed in initial
SCOREFILE)
*                                Must put "keep TARGET_WINS" back into SCOREFILE
for Error scoring to work;

data SCOREFILE;
        set SCOREFILE;
        ERROR = TARGET_WINS - P_TARGET_WINS;
```

```
        ERROR = ERROR**2;
run;

* testing model- view the amount of error^2 and average mean error ;
*                                   for file with target value known (INFILE);
        proc means data=SCOREFILE MEAN SUM;
                var ERROR;
                TITLE8 "Model#2 Error on Moneyball dataset - Mean and total error";
        run;

        TITLE8 ; * clear title;

* view a few observations with predicted target value;
        proc print data=SCOREFILE(obs=10);
                title9 "Model#2 Error on Moneyball dataset - Predictions";
        run;

        TITLE9 ; * clear title;
*/




*********************************************************;
****      CREATE FILE TO STORE SCORED DATA         ****;
*********************************************************;

* print a few observations to ensure can access the dataset (moneyball_test);
        proc print data=&INFILE2. (obs=5);
        title10 "Testing Access to Moneyball_test - dataset";
        run;

        title10 ;

* code to store scored code into my SAS folder Assignments;
libname scorelib "/home/derekhughes2014/Assignments";
        data scorelib.DEREK_HUGHES_FILE_moneyball_test;
                set SCOREFILE;
        run;

* view scored data on Moneyball_test - click "download" button
        * in Folders to get this file on local CPU;
        proc print data=scorelib.DEREK_HUGHES_FILE_moneyball_test (obs=10);
                title10 "Model#2 vs Dataset in SCOREfILE that's saved to CPU -
```

```
(SCOREFILE currently set to Moneyball_test dataset with error measurements";
        run;

                title10 ;



**********************************************;
******          BINGO BONUS          **********;
**********************************************;

(20 points)

* PROC GLM;

proc glm data=&TEMPFILE3. ;
                        model TARGET_WINS = SINGLES

        TEAM_BATTING_3B

        TEAM_BATTING_HR

        TEAM_BATTING_BB

        SQRT_IMP_TEAM_FIELDING_E

        IMP_TEAM_BATTING_SO

        LOG10_TEAM_BASERUN_CS

        IMP_TEAM_BASERUN_SB ;
                        title9 "GLM REGRESSION - Model#2 (BEST of the 3) - with
Variables Transformed and Inappropriate Variables Dropped";
                run;

                title9 ;

* PROC GENMOD;

proc genmod data=&TEMPFILE3.;
                        model TARGET_WINS = SINGLES

        TEAM_BATTING_3B

        TEAM_BATTING_HR
```

```
        TEAM_BATTING_BB

        SQRT_IMP_TEAM_FIELDING_E

        IMP_TEAM_BATTING_SO

        LOG10_TEAM_BASERUN_CS

        IMP_TEAM_BASERUN_SB / link=identity;
                        title9 "GENMOD REGRESSION - Model#2 (BEST of the 3) -
with Variables Transformed and Inappropriate Variables Dropped";
                run;
```