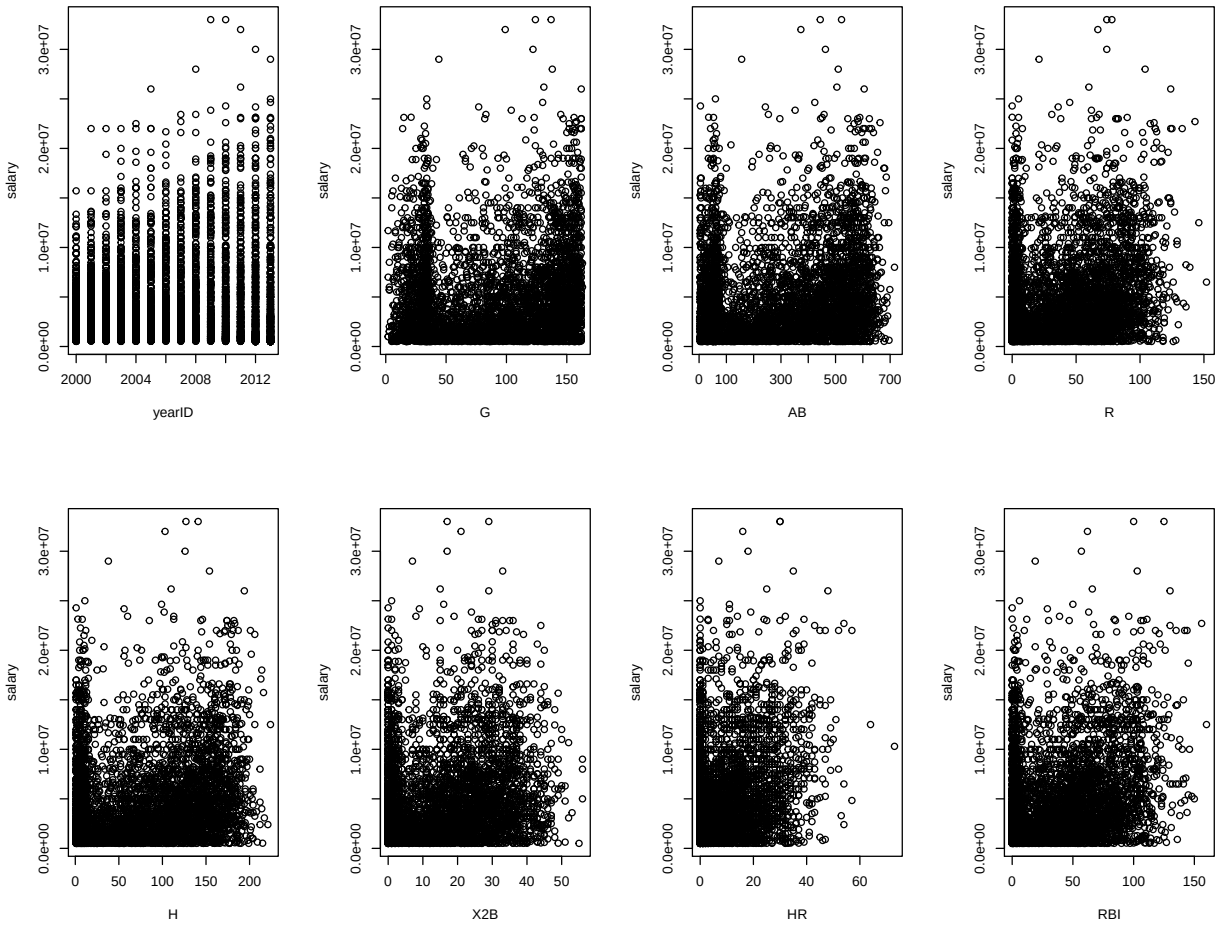
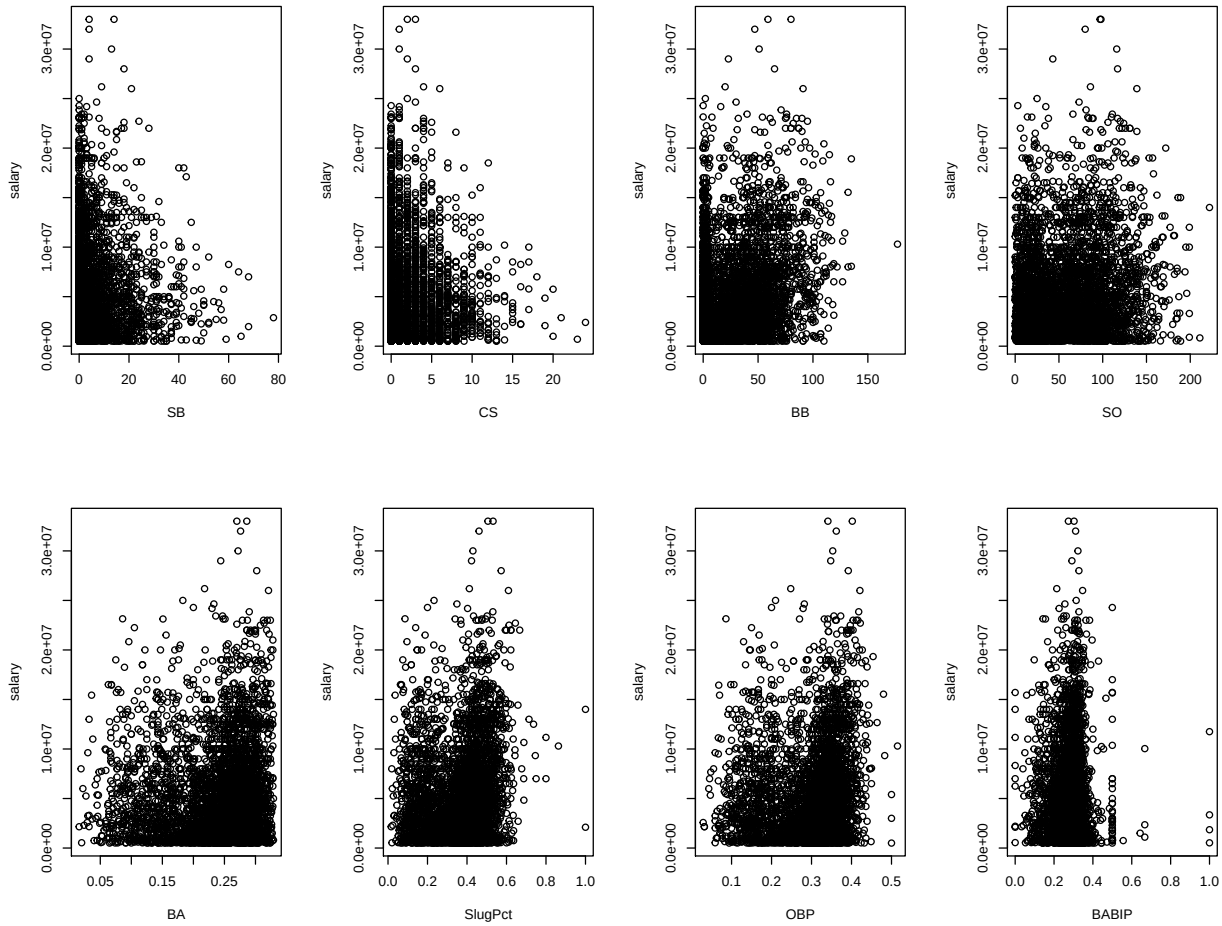
Stat 506 Assignment 5

Leslie Gains-Germain

February 20, 2015

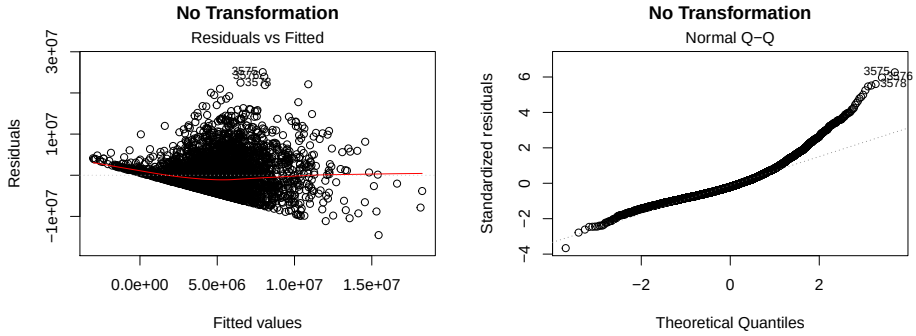
I first looked at the following exploratory plots. Salaries appear to increase across batting average (BA), slugging percentage, on base percentage (OBP), and batting average on ball in play (BABIP). Salaries appear to decrease across stolen bases (SB), caught stealing (CS), base on balls (BB), and strikeouts (SO).

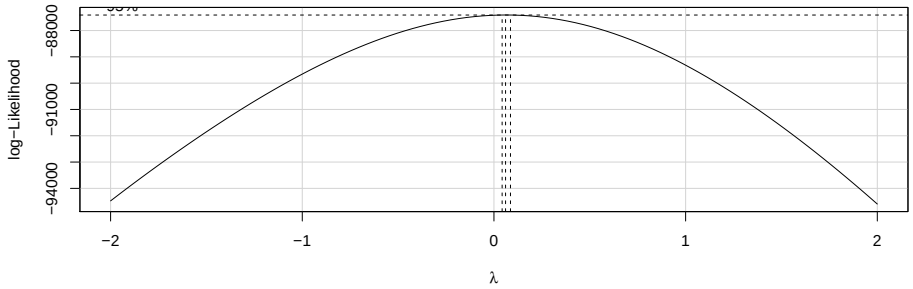


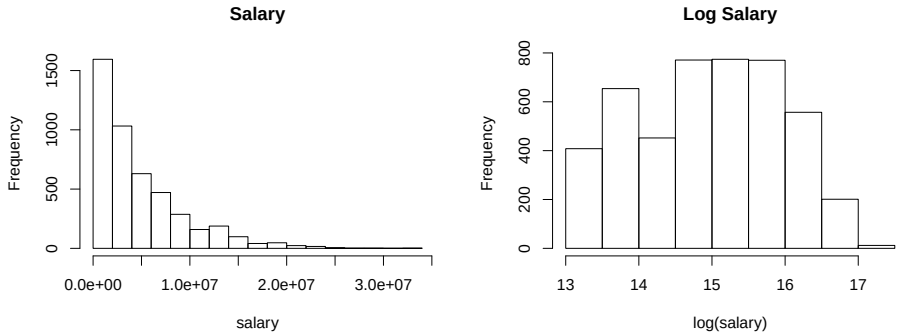


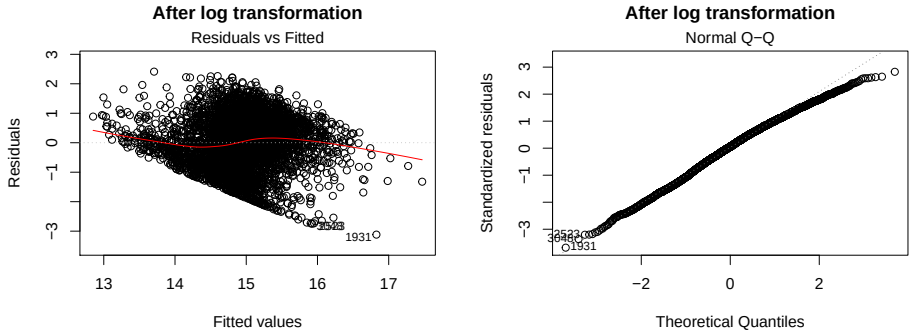
1. Fit salary using all other columns except playerID and teamID and check to see if you want to transform the response variable.

   I fit salary on all other columns except playerID and teamID. I saw problems with constant variance and normality assumptions in the residual plots. The boxcox procedure suggested a log transformation, so I fit a model with log salary as the response variable. The histogram below shows that salaries exhibit a highly skewed distribution, but log salaries are more symmetric. The residual plots also look better after the transformation.









2. What multicollinearity issues does the full model have?

   I first looked at the variance inflation factors and tolerances. There were some large variance inflation factors and small tolerances, which suggests that a multicollinearity issue is present.

```
> round(vif(lm.log),2)
(Intercept)      yearID           G          AB           R           H
       0.00        1.10       16.27      109.90       33.80      114.23
        X2B         X3B          HR         RBI          SB          CS
      10.83        2.22       17.46       29.95        3.36        2.96
        BB          SO         IBB         HBP          SH          SF
       9.85        7.58        2.20        1.68        1.41        2.75
      GIDP          BA     SlugPct         OBP       BABIP
       3.69       23.09       11.51       12.09        3.08
```

```
> round(tol(lm.log),5)
(Intercept)      yearID           G          AB           R           H
         NA     0.91223     0.06145     0.00910     0.02959     0.00875
        X2B         X3B          HR         RBI          SB          CS
    0.09235     0.45112     0.05727     0.03339     0.29779     0.33838
         BB          SO         IBB         HBP          SH          SF
    0.10150     0.13200     0.45537     0.59352     0.70841     0.36428
        GIDP          BA     SlugPct         OBP       BABIP
    0.27134     0.04332     0.08691     0.08268     0.32500
```

I then investigated further. I played around with using Jim's `collin` *function, but eventually I settled on the* `colldiag` *function in the* `perturb` *package because it has the ability to center and scale predictor variables.*

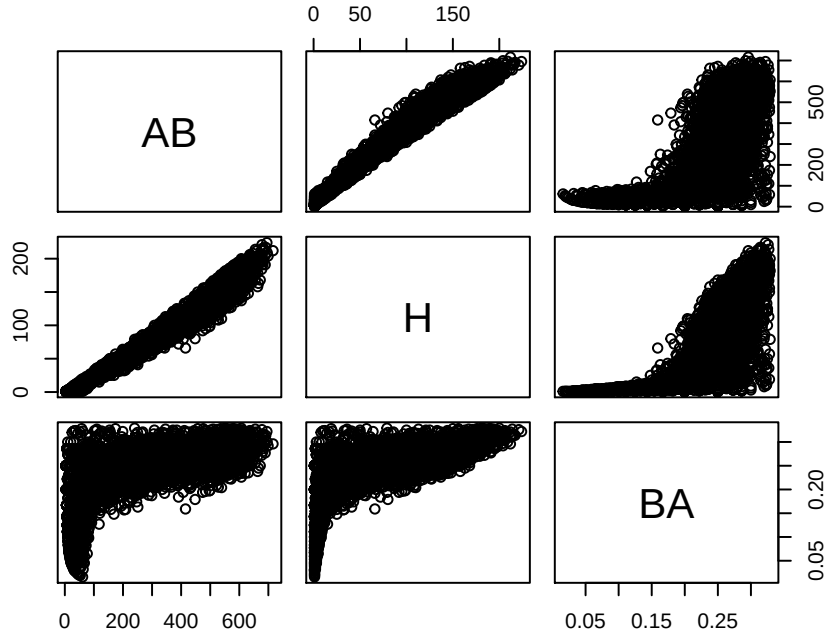
The `colldiag` *function outputs a* $23x25$ *matrix. This is too large to display here, so I just pulled out the important pieces. Below, I show the rows that had condition indexes larger than* 30*, and I show only the columns that appeared to have multicollinearity issues.*

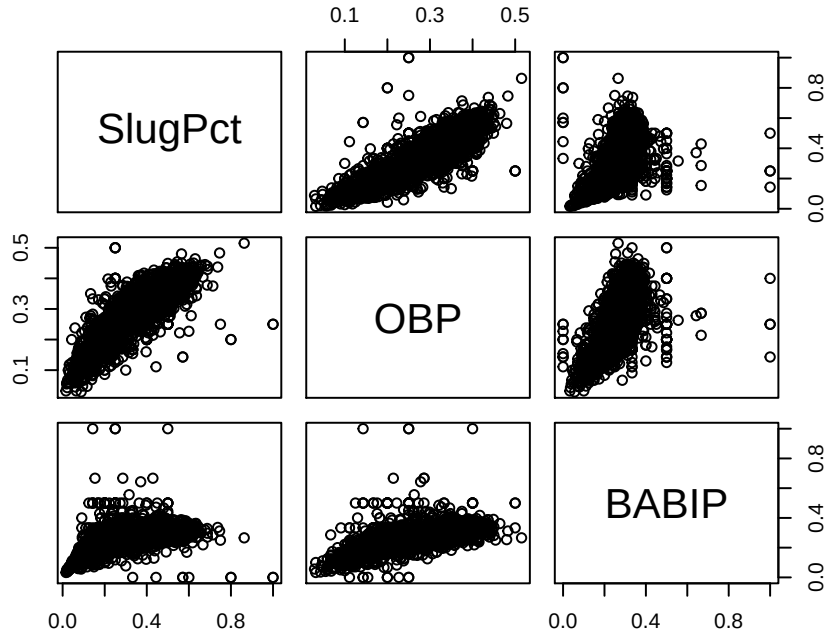
In the bottom row, we see that year is highly collinear with the intercept. This makes sense because we haven't centered year yet. In rows 5 *and* 6*, we see that at bat (AB), hits (H), and batting average (BA) are collinear. In rows* 4 *and* 5*, we see that slugging percentage, on base percentage (OBP), and batting average on balls in play (BABIP) are collinear.*

	condindx	intercept	yearID	G	AB	R	H	HR	RBI
1	35.15	0.000	0.000	0.161	0.002	0.336	0.005	0.000	0.261
2	38.84	0.000	0.000	0.283	0.001	0.000	0.009	0.393	0.503
3	50.13	0.000	0.000	0.249	0.064	0.553	0.082	0.308	0.168
4	58.41	0.000	0.000	0.005	0.007	0.012	0.011	0.105	0.001
5	92.76	0.000	0.000	0.043	0.312	0.012	0.104	0.025	0.015
6	108.08	0.000	0.000	0.107	0.599	0.023	0.767	0.005	0.020
7	3074.41	1.000	1.000	0.001	0.000	0.014	0.005	0.000	0.007

	condindx	BB	SO	BA	SlugPct	OBP	BABIP
1	35.15	0.063	0.081	0.000	0.009	0.000	0.034
2	38.84	0.001	0.087	0.001	0.001	0.006	0.085
3	50.13	0.100	0.011	0.002	0.021	0.005	0.000
4	58.41	0.241	0.026	0.008	0.517	0.515	0.116
5	92.76	0.118	0.200	0.588	0.201	0.301	0.349
6	108.08	0.027	0.070	0.390	0.109	0.158	0.009
7	3074.41	0.000	0.019	0.002	0.000	0.000	0.002

Here are two pairs plots that show the variables we identified as being collinear. There does appear to be strong relationships in each of the plots shown.





(a) Does centering make them better or worse? Explain the impact of centering.
I centered and scaled the predictors, and part of the `colldiag` *output is below. There is now only one condition number larger than* 30 *due to a collinearity issue between at bats and number of hits. This makes sense because in the plots above we saw a*

very strong linear relationship between at bats and hits.

	condindx	yearID	G	AB	R	H	HR	RBI
1	20.36	0.000	0.025	0.000	0.239	0.000	0.108	0.449
2	23.16	0.001	0.046	0.029	0.002	0.024	0.338	0.172
3	26.17	0.017	0.133	0.100	0.578	0.055	0.349	0.248
4	50.56	0.001	0.124	0.855	0.049	0.896	0.001	0.041

	condindx	BB	SO	BA	SlugPct	OBP	BABIP
1	20.36	0.250	0.007	0.163	0.010	0.172	0.045
2	23.16	0.063	0.025	0.626	0.419	0.149	0.198
3	26.17	0.105	0.056	0.012	0.002	0.003	0.026
4	50.56	0.001	0.191	0.046	0.010	0.011	0.026

(b) Which variables seem to be most highly multicollinear? *See my explanations above.*

3. Try the step variable selection technique available in R. Does this fix the multicollinearity problem?

   I used stepwise selection in R to choose a model. The `step` *function kept* 17 *of the* 22 *predictors, removing runs (R), hits (H), stolen bases (SB), stolen flies (SF), and batting average (BA).*

   I examined the `colldiag` *output with the model that R chose. Note that I centered and rescaled the predictors before looking at the collinearity output. Because hits (H) was removed, we do not see any multicollinearity issues (see partial output below). The largest condition number was* 20.5.

	condindx	yearID	G	AB	X2B	X3B	HR	RBI
1	1.00	0.000	0.001	0.000	0.001	0.002	0.001	0.000
2	2.44	0.000	0.001	0.000	0.000	0.073	0.002	0.000
3	2.79	0.002	0.000	0.000	0.000	0.015	0.003	0.001
4	3.01	0.859	0.000	0.000	0.000	0.000	0.000	0.000
5	3.66	0.000	0.000	0.000	0.001	0.131	0.001	0.000
6	3.79	0.015	0.000	0.000	0.000	0.036	0.000	0.000
7	3.94	0.013	0.003	0.001	0.007	0.010	0.001	0.001
8	4.60	0.003	0.000	0.000	0.000	0.093	0.019	0.002
9	4.76	0.006	0.000	0.000	0.001	0.439	0.006	0.000
10	5.09	0.048	0.000	0.000	0.000	0.026	0.000	0.000
11	6.00	0.010	0.009	0.000	0.002	0.001	0.042	0.007
12	7.41	0.000	0.022	0.008	0.270	0.085	0.017	0.002
13	8.25	0.034	0.037	0.001	0.008	0.014	0.060	0.026
14	10.09	0.002	0.280	0.015	0.344	0.007	0.031	0.018
15	12.71	0.000	0.102	0.001	0.021	0.011	0.022	0.056
16	17.71	0.000	0.110	0.022	0.224	0.017	0.764	0.712
17	20.50	0.007	0.435	0.949	0.119	0.040	0.031	0.175

	condindx	CS	BB	SO	IBB	HBP	SH	GIDP	SlugPct	OBP	BABIP
1	1.00	0.002	0.001	0.002	0.002	0.003	0.001	0.002	0.001	0.001	0.001
2	2.44	0.084	0.000	0.000	0.017	0.000	0.141	0.000	0.003	0.001	0.001
3	2.79	0.009	0.001	0.004	0.014	0.004	0.014	0.005	0.007	0.017	0.226
4	3.01	0.000	0.000	0.003	0.004	0.000	0.007	0.000	0.000	0.000	0.003
5	3.66	0.033	0.004	0.000	0.096	0.357	0.094	0.026	0.000	0.001	0.030
6	3.79	0.028	0.004	0.000	0.186	0.085	0.488	0.000	0.000	0.001	0.088
7	3.94	0.015	0.005	0.000	0.107	0.411	0.002	0.139	0.000	0.001	0.002
8	4.60	0.247	0.000	0.058	0.179	0.012	0.077	0.094	0.014	0.003	0.005
9	4.76	0.340	0.009	0.058	0.061	0.053	0.000	0.048	0.001	0.001	0.016
10	5.09	0.012	0.002	0.038	0.045	0.017	0.125	0.001	0.106	0.079	0.390
11	6.00	0.137	0.185	0.032	0.041	0.008	0.013	0.008	0.064	0.066	0.028
12	7.41	0.019	0.010	0.070	0.002	0.005	0.012	0.501	0.000	0.003	0.002
13	8.25	0.003	0.240	0.411	0.221	0.002	0.000	0.002	0.080	0.002	0.069
14	10.09	0.004	0.047	0.175	0.002	0.005	0.010	0.084	0.040	0.010	0.008
15	12.71	0.002	0.461	0.093	0.012	0.024	0.006	0.008	0.545	0.734	0.115
16	17.71	0.000	0.024	0.025	0.001	0.009	0.000	0.038	0.121	0.069	0.000
17	20.50	0.065	0.004	0.031	0.011	0.004	0.009	0.045	0.017	0.009	0.015

4. Export the data and import it into SAS. Use PROC REG.

   (a) Does the collin output agree (non-centered and centered) with the R version?

   Yes, the collin output from SAS does agree with the R version. I've included part of the collin output below. We can see that the condition numbers at the proportion of variations are the same as the `colldiag` *output in R. The centered output is shown below on the right.*

Condition Index	Intercept	year	G	AB
1.00000	9.459265E-9	9.467143E-9	0.00004196	0.00000893
3.40365	2.382875E-7	2.385924E-7	9.428233E-7	0.00000909
3.72616	8.340718E-8	8.350123E-8	0.00000338	0.00000342
6.22032	1.379824E-7	1.385091E-7	1.676041E-8	0.00000644
6.36267	3.962917E-8	3.96262E-8	0.00008653	0.00005775
6.93238	4.286743E-8	4.289737E-8	0.00020271	0.00011134
7.62250	1.128381E-9	1.14249E-9	0.00001543	0.00000785
9.12238	7.28172E-11	8.59123E-11	0.00000761	2.075628E-7
9.63841	5.459121E-8	5.440175E-8	0.00050881	0.00023566
10.86639	7.66824E-10	6.79874E-10	0.00000115	0.00004587
13.48639	1.259481E-8	1.173946E-8	0.00361	0.00024809
14.92102	0.00000149	0.00000151	0.00036245	0.00002983
15.95885	4.21977E-11	2.27696E-10	0.00563	0.00164
22.15130	0.00002463	0.00002462	0.00981	0.00146
24.58992	0.00000189	0.00000189	0.11092	0.01111
26.99068	2.536855E-7	2.539135E-7	0.02057	0.00004642
35.15055	9.718295E-7	9.762402E-7	0.16058	0.00176
38.84222	0.00000135	0.00000135	0.28272	0.00095603
50.12886	1.736886E-7	2.357361E-7	0.24948	0.06409
58.41278	0.00000280	0.00000286	0.00452	0.00673
92.75914	0.00000276	0.00000263	0.04253	0.31196
108.08333	0.00000310	0.00000352	0.10723	0.59914
3074.41380	0.99996	0.99996	0.00115	0.00033678

Condition Index	year	G	AB
1.00000	0.00002021	0.00034581	0.00005354
2.46966	0.00004895	0.00012079	0.00003382
2.84766	0.00090522	0.00030989	0.00006330
3.48426	0.83101	0.00004222	0.00000926
3.99473	0.00366	0.00143	0.00019295
4.33515	0.00028521	0.00016669	0.00002959
4.39219	0.02222	0.00019666	0.00008936
5.16997	0.00111	0.00001624	0.00000523
5.34410	0.03053	0.00008677	0.00001225
5.61031	0.00893	0.00167	0.00028100
5.68764	0.02772	0.00005860	0.00000594
6.88026	0.00323	0.00865	0.00007117
7.56684	0.00303	0.00000668	0.00016097
8.50418	0.00001454	0.01748	0.00195
9.16081	0.04479	0.03842	0.00057826
11.39232	0.00148	0.14564	0.00359
14.52082	0.00114	0.24225	0.00062647
16.46780	0.00025367	0.21661	0.00728
20.35780	0.00000952	0.02460	0.00032667
23.16314	0.00104	0.04569	0.02925
26.16861	0.01729	0.13264	0.10031
50.55797	0.00129	0.12357	0.85509

   (b) Try some of the PROC REG selection methods to see if they fix the multicollinearity problem.

   I used the PROC REG selection procedure in SAS, and a the chosen model contained 18 *of the* 22 *variables. All variables were chosen except runs (R), hits (H),*

stolen bases (SB), and stolen flies (SF). After centering and rescaling, there are no multicollinearity issues in this model. Note that this is nearly the same model that R chose, but the stepwise procedure in R removed batting average (BA) while SAS left it in. In PROC REG, the default selection criteria is R^2 while in `step` *the default selection criteria is AICC.*

Number	Eigenvalue	Condition Index	HR	SlugPct	BB	G	AB	OBP
1	9.90262	1.00000	0.00051443	0.00065200	0.00098863	0.00057190	0.00023872	0.00057314
2	1.59606	2.49087	0.00026315	0.00369	0.00007547	0.00108	0.00057425	0.00329
3	1.46509	2.59982	0.00421	0.00090917	0.00198	0.00004923	0.00002722	0.00420
4	1.02618	3.10644	0.00015799	0.00014072	0.00017373	0.00007403	0.00004058	0.00004913
5	0.70260	3.75423	0.00074042	0.00041680	0.00357	0.00058515	0.00020123	0.00006507
6	0.64940	3.90498	0.00004067	0.00014969	0.00344	0.00000424	0.00001502	0.00021580
7	0.60348	4.05083	0.00056953	0.00001293	0.00500	0.00245	0.00128	0.00168
8	0.44311	4.72736	0.01838	0.00356	0.00012107	0.00008457	0.00003764	0.00342
9	0.41965	4.85769	0.00036153	0.01135	0.00630	0.00002205	4.197237E-8	0.00157
10	0.38358	5.08098	0.00446	0.04612	0.00267	0.00040725	0.00019427	0.02050
11	0.25996	6.17190	0.04022	0.02900	0.16571	0.00827	0.00028775	0.05910
12	0.17209	7.58581	0.01878	0.00301	0.01208	0.02705	0.00780	0.00699
13	0.13975	8.41774	0.04404	0.02416	0.18329	0.03379	0.00073309	0.00001626
14	0.09487	10.21671	0.04161	0.07612	0.01524	0.20204	0.01148	0.00016098
15	0.05832	13.03080	0.01878	0.42161	0.30665	0.15956	0.00126	0.36031
16	0.03734	16.28512	0.10491	0.01363	0.25168	0.26287	0.03804	0.44366
17	0.02721	19.07677	0.68156	0.14168	0.00979	0.00806	0.04107	0.01661
18	0.01868	23.02374	0.02039	0.22380	0.03124	0.29303	0.89670	0.07758

5. In SAS or R, what terms do we have to remove (or combine) to make the condition numbers all less than 30? Suppose that for some reason OBP has to be in the model. Does removing terms (other than BA) from the model change the estimated BA effect on salaries?

   After centering and rescaling, the only term we need to remove from the model to ensure that all condition numbers are less than 30 *is hits (H). I looked at the coefficient estimates for the full model, and then I removed hits and looked at the coefficient estimates for this reduced model. The output is shown below. We can see that the coefficient for BA in the full model is* -1.3265*, and the coefficient for BA in the model without hits is* -1.1466*. So the effect of batting average on salaries is estimated to be less extreme when hits is removed.*

Parameter	Estimate	Standard Error	t Value	Pr > |t|
year	0.007852122	0.00004139	189.72	<.0001
G	-0.024747411	0.00105034	-23.56	<.0001
AB	0.005792184	0.00064993	8.91	<.0001
R	-0.001897791	0.00217834	-0.87	0.3837
H	0.001384080	0.00225376	0.61	0.5392
X2B	-0.008270891	0.00322644	-2.56	0.0104
X3B	-0.033391632	0.00891650	-3.74	0.0002
HR	0.007641766	0.00495784	1.54	0.1233
RBI	0.004196120	0.00202556	2.07	0.0384
SB	0.002726993	0.00270326	1.01	0.3131
CS	-0.025877850	0.00781848	-3.31	0.0009
BB	0.016857026	0.00149538	11.27	<.0001
SO	-0.004156019	0.00086140	-4.82	<.0001
IBB	0.018792486	0.00469137	4.01	<.0001
HBP	0.009485964	0.00439342	2.16	0.0309
SH	0.022241996	0.00453776	4.90	<.0001
SF	-0.004539088	0.00814595	-0.56	0.5774
GIDP	0.008979265	0.00399081	2.25	0.0245
BA	-1.326530554	1.01302509	-1.31	0.1904
SlugPct	0.902000432	0.34656066	2.60	0.0093
OBP	-3.927493027	0.58262452	-6.74	<.0001
BABIP	1.347265746	0.33954343	3.97	<.0001

Parameter	Estimate	Standard Error	t Value	Pr > |t|
year	0.007842182	0.00003809	205.89	<.0001
G	-0.024847270	0.00103761	-23.95	<.0001
AB	0.006095690	0.00042206	14.44	<.0001
R	-0.001450517	0.00205284	-0.71	0.4799
X2B	-0.007781123	0.00312611	-2.49	0.0128
X3B	-0.032972229	0.00888971	-3.71	0.0002
HR	0.007505834	0.00495256	1.52	0.1297
RBI	0.004495887	0.00196573	2.29	0.0222
SB	0.002742916	0.00270296	1.01	0.3103
CS	-0.025794231	0.00781676	-3.30	0.0010
BB	0.016801419	0.00149253	11.26	<.0001
SO	-0.004341321	0.00080677	-5.38	<.0001
IBB	0.019251284	0.00463119	4.16	<.0001
HBP	0.009492433	0.00439311	2.16	0.0308
SH	0.022346787	0.00453424	4.93	<.0001
SF	-0.005064351	0.00810037	-0.63	0.5319
GIDP	0.009088576	0.00398656	2.28	0.0227
BA	-1.146643587	0.96968602	-1.18	0.2371
SlugPct	0.866244496	0.34161162	2.54	0.0113
OBP	-3.971334373	0.57819502	-6.87	<.0001
BABIP	1.366128337	0.33812836	4.04	<.0001

6. Look again at the full model and the process you used in (e) to reduce the number of variables included. Did any coefficient estimates change markedly when another variable was removed?

   We can see in the output above that the coefficient estimates are slightly different when hits is removed, but none of them change dramatically. All estimates keep the same sign, and most of the coefficient estimates change by less than 0.01*. The estimate of the batting average effect shows the largest change. Also notice that the standard errors all decrease when hits is removed.*

7. Last week we saw that NFL teams have almost no variation in average salary. The cap on total team salary is enforced in football, but not so much in baseball. After reducing the full model try fitting a random team effect and see if the other estimates change. Explain your favorite model and summarize how OBP is related to salary in these data.

   I chose the model selected by SAS in PROC REG. The output on the left (below) shows the model with team as a fixed effect. The output on the right is the model output with team as a random effect. The random effect model has an an AIC of 11537.6*. The team random effect is estimated to be* 0.03048*, which suggests that there is more team to team variability in salaries than we saw in the NFL.*

   The coefficient estimate for OBP changes from -3.937 *to* -3.957 *when team is included as a random effect. This means that for a one point increase in on base percentage (OBP), salary is estimated to change by a multiplicative factor of* 0.019 *given that all other variables accounted for in the model.*

Covariance Parameter Estimates	
Cov Parm	Estimate
team	0.03048
Residual	0.7036

Fit Statistics	
-2 Log Likelihood	11497.5
AIC (Smaller is Better)	11537.5
AICC (Smaller is Better)	11537.6
BIC (Smaller is Better)	11567.4

Parameter	Estimate	Standard Error	t Value	Pr > |t|
HR	0.006907720	0.00446661	1.55	0.1220
SlugPct	0.879818384	0.34042457	2.58	0.0098
BB	0.016380370	0.00134539	12.18	<.0001
G	-0.024871027	0.00101975	-24.39	<.0001
AB	0.006039875	0.00038033	15.88	<.0001
OBP	-3.937234357	0.57646570	-6.83	<.0001
IBB	0.019873880	0.00456427	4.35	<.0001
year	0.007843214	0.00003782	207.40	<.0001
SO	-0.004228808	0.00078482	-5.39	<.0001
X3B	-0.032722766	0.00835128	-3.92	<.0001
SH	0.022230767	0.00450619	4.93	<.0001
BABIP	1.376301271	0.33714958	4.08	<.0001
CS	-0.022131916	0.00643559	-3.44	0.0006
GIDP	0.008857551	0.00389968	2.27	0.0232
HBP	0.008962779	0.00432273	2.07	0.0382
X2B	-0.008349593	0.00303441	-2.75	0.0060
RBI	0.003883962	0.00181106	2.14	0.0320
BA	-1.230450308	0.96326735	-1.28	0.2015

Solution for Fixed Effects					
Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
HR	0.005064	0.004435	4548	1.14	0.2536
SlugPct	0.8764	0.3338	4548	2.63	0.0087
BB	0.01614	0.001321	4548	12.22	<.0001
G	-0.02553	0.001008	4548	-25.34	<.0001
AB	0.006233	0.000378	4548	16.48	<.0001
OBP	-3.9566	0.5653	4548	-7.00	<.0001
IBB	0.01891	0.004531	4548	4.17	<.0001
year	0.007833	0.000040	4548	194.59	<.0001
SO	-0.00375	0.000773	4548	-4.85	<.0001
X3B	-0.03329	0.008278	4548	-4.02	<.0001
SH	0.01873	0.004476	4548	4.18	<.0001
BABIP	1.3304	0.3305	4548	4.03	<.0001
CS	-0.02049	0.006346	4548	-3.23	0.0013
GIDP	0.008482	0.003828	4548	2.22	0.0268
HBP	0.008926	0.004260	4548	2.10	0.0362
X2B	-0.00847	0.002996	4548	-2.83	0.0047
RBI	0.003848	0.001795	4548	2.14	0.0321
BA	-1.0824	0.9439	4548	-1.15	0.2516

So now we've seen that two measures of player consistency (batting average and on base percentage) have negative coefficient estimates. This is crazy! Given that all of the other variables are fixed, players who have higher batting averages and higher on base percentages are earning less money. But, BABIP and SlugPct have relatively large positive coefficient estimates.

Just to explore, I removed BABIP and Slugpct from the model, and my coefficient estimate for batting average became positive (see below). The coefficient estimate for OBP didn't change much. The overall lesson I'm taking from this is that I would not trust this model to explain relationships. Even though mathematically we've removed the multicollinearity in the X matrix, a model with this many predictors is too difficult too understand. Practically, it doesn't make sense to describe the relationship between salary and OBP when G, HR, R, X2B, RBI, and 12 other variables are fixed. I think that a much better way to approach model building is to first define the relationship that you want to describe and then build a model from there while thinking about which terms should be included in the model from a practical standpoint.

Solution for Fixed Effects					
Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
HR	0.008451	0.003929	4550	2.15	0.0315
BB	0.01608	0.001316	4550	12.22	<.0001
G	-0.02556	0.001008	4550	-25.35	<.0001
AB	0.005728	0.000357	4550	16.06	<.0001
OBP	-4.0051	0.5558	4550	-7.21	<.0001
IBB	0.01884	0.004535	4550	4.15	<.0001
year	0.007868	0.000039	4550	201.01	<.0001
SO	-0.00245	0.000708	4550	-3.46	0.0006
X3B	-0.02856	0.008150	4550	-3.50	0.0005
SH	0.01835	0.004467	4550	4.11	<.0001
CS	-0.02018	0.006355	4550	-3.18	0.0015
GIDP	0.008428	0.003835	4550	2.20	0.0280
HBP	0.008892	0.004266	4550	2.08	0.0372
X2B	-0.00595	0.002909	4550	-2.05	0.0407
RBI	0.003610	0.001793	4550	2.01	0.0442
BA	1.6597	0.6578	4550	2.52	0.0117

Code Appendix

```
baseball <- read.csv("~/Documents/Stat506/Homework/HW5/BaseballStats2000.csv", head=T)
```

```
par(mfrow=c(1,4))
with(baseball, plot(yearID, salary))
with(baseball, plot(G, salary))
with(baseball, plot(AB, salary))
with(baseball, plot(R, salary))
with(baseball, plot(H, salary))
with(baseball, plot(X2B, salary))
with(baseball, plot(HR, salary))
with(baseball, plot(RBI, salary))
with(baseball, plot(SB, salary))
with(baseball, plot(CS, salary))
with(baseball, plot(BB, salary))
with(baseball, plot(SO, salary))
with(baseball, plot(BA, salary))
with(baseball, plot(SlugPct, salary))
with(baseball, plot(OBP, salary))
with(baseball, plot(BABIP, salary))
```

```
lm.ball <- lm(salary~., data=baseball[,c(2,4:25)])
par(mfrow=c(1,2))
plot(lm.ball, which=c(1:2), main="No Transformation")
require(car)
par(mfrow=c(1,1))
boxCox(lm.ball)
par(mfrow=c(1,2))
#Yes, let's try a transformation-boxcox suggests log
lm.log <- lm(log(salary)~., data=baseball[,c(2,4:25)])
with(baseball, hist(salary, main="Salary"))
with(baseball, hist(log(salary), main="Log Salary"))
plot(lm.log, which=c(1:2), main="After log transformation")
```

```
#check collinearity measures
#round(vif(lm.log),2)
#round(tol(lm.log),5)
round(collin(lm.log),3)
#slightly different condition numbers when we use Jims function
#Use colldiag so we can center later on
require(perturb)
colin <- as.data.frame(print(colldiag(baseball[,c(2,4:24)])))
```

```
names(colin)[1] <- "condindx"
colin$condindx <- as.numeric(as.character(colin$condindx))
require(dplyr)
colin.more <- colin %>%
  select(condindx, intercept, yearID, G, AB, R, H, HR, RBI) %>%
  filter(condindx > 30)
colin.next <- colin %>%
  select(condindx, BB, SO, BA, SlugPct, OBP, BABIP) %>%
  filter(condindx > 30)
require(xtable)
xtable(colin.more)
xtable(colin.next)
```

```
pairs(select(baseball, c(AB, H, BA)))
pairs(select(baseball, c(SlugPct, OBP, BABIP)))
```

```
require(perturb)
colin.ctr <- as.data.frame(print(colldiag(baseball[,c(2,4:24)], scale=TRUE, center=TRUE)))
```

```
names(colin.ctr)[1] <- "condindx"
colin.ctr$condindx <- as.numeric(as.character(colin.ctr$condindx))
#condition numbers all went less than thirty except the last
colin.more <- colin.ctr %>%
  select(condindx, yearID, G, AB, R, H, HR, RBI) %>%
  filter(condindx > 20)
colin.next <- colin.ctr %>%
  select(condindx, BB, SO, BA, SlugPct, OBP, BABIP) %>%
  filter(condindx > 20)
require(xtable)
print(xtable(colin.more), table.placement="h")
print(xtable(colin.next), table.placement="h")
```

```
step.fix <- step(lm.log, direction="both")
step.choice <- baseball %>%
  select(yearID, G, AB, X2B, X3B, HR, RBI, CS, BB, SO, IBB, HBP, SH, GIDP, SlugPct, OBP, BABIP)
```

```
require(perturb)
colin.step <- as.data.frame(print(colldiag(step.choice, scale=TRUE, center=TRUE)))
```

```
#vifs are still large
#round(vif(step.fix),5)
names(colin.step)[1] <- "condindx"
colin.step$condindx <- as.numeric(as.character(colin.step$condindx))
#condition numbers all went less than thirty except the last
colin.more <- colin.step %>%
  select(condindx, yearID, G, AB, X2B, X3B, HR, RBI)
colin.next <- colin.step %>%
  select(condindx, CS, BB, SO, IBB, HBP, SH, GIDP, SlugPct, OBP, BABIP)
require(xtable)
xtable(colin.more)
print(xtable(colin.next), table.placement="h")
```

```
DATA baseball;
  INFILE "/folders/myfolders/BaseballStats2000.csv" firstobs=2 delimiter =',';
  INPUT player $ year team $ G AB R H X2B X3B HR RBI SB CS BB SO IBB HBP SH
      SF GIDP BA SlugPct OBP BABIP salary;
```

```
  ;
RUN;

DATA baseball;
  SET baseball;
log = log(salary);
;
RUN;

PROC REG DATA=baseball;
MODEL log = year G AB R H X2B X3B HR RBI SB CS BB SO
                    IBB HBP SH SF GIDP BA SlugPct OBP BABIP / COLLIN;
RUN;

*/Center and scale;

PROC REG DATA=baseball;
MODEL log = year G AB R H X2B X3B HR RBI SB CS BB SO
               IBB HBP SH SF GIDP BA SlugPct OBP BABIP / COLLINoint;
RUN;

PROC REG DATA=baseball PLOTS=criteria;
MODEL log = year G AB R H X2B X3B HR RBI SB CS BB SO
               IBB HBP SH SF GIDP BA SlugPct OBP BABIP / selection=forward ALL;
RUN;

*check for multicollinearity in model SAS chose with forward selection;

PROC REG DATA=baseball;
MODEL log = HR SlugPct BB G AB OBP IBB year SO X3B SH BABIP CS GIDP
HBP X2B RBI BA / COLLINoint;
RUN;

*Look at coefficient estimates for both models;

PROC GLM data=baseball;
MODEL log = year G AB R H X2B X3B HR RBI SB CS BB SO
               IBB HBP SH SF GIDP BA SlugPct OBP BABIP / NOINT SOLUTION;
RUN;

PROC GLM data=baseball;
MODEL log = year G AB R X2B X3B HR RBI SB CS BB SO
               IBB HBP SH SF GIDP BA SlugPct OBP BABIP / NOINT SOLUTION;
RUN;

*Now choose favorite model and fit with a random team effect;
```

```
PROC MIXED data=baseball method=ml;
CLASS team;
MODEL log = HR SlugPct BB G AB OBP IBB year SO X3B SH BABIP CS GIDP
HBP X2B RBI BA / NOINT SOLUTION;
RANDOM team / s;
RUN;

*compare to results when team was not a random effect;

PROC GLM DATA=baseball;
MODEL log = HR SlugPct BB G AB OBP IBB year SO X3B SH BABIP CS GIDP
HBP X2B RBI BA / NOINT SOLUTION;
RUN;
```