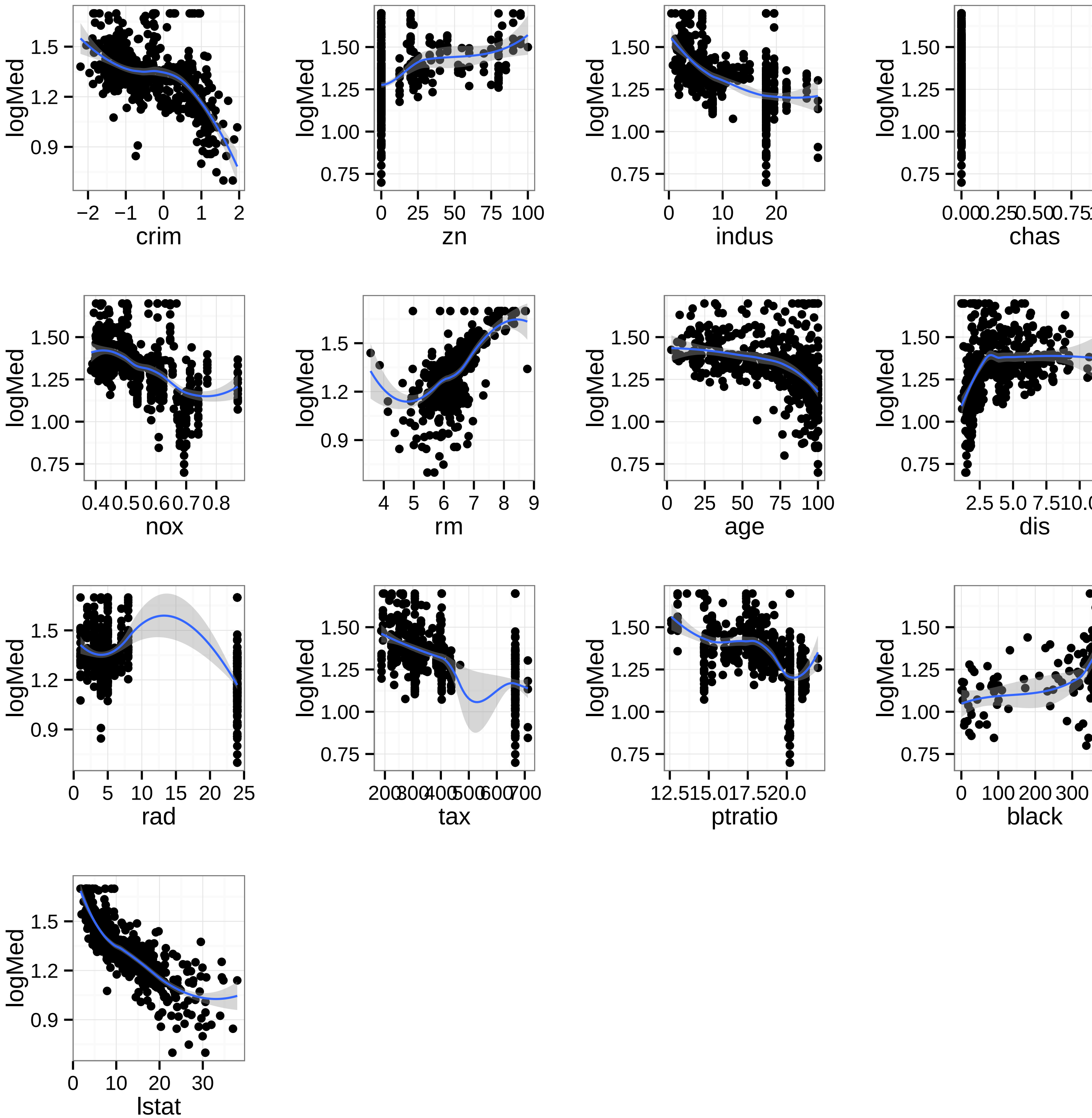
Stat 506 Spring 2015 Assignment 5

Due: February 25, 4pm

Grab the `collin` function R code from the Stat 506 website using the code below or use `colldiag` in the `perturb` package.

Choose one of these two situations in order to explore multicollinearity issues.

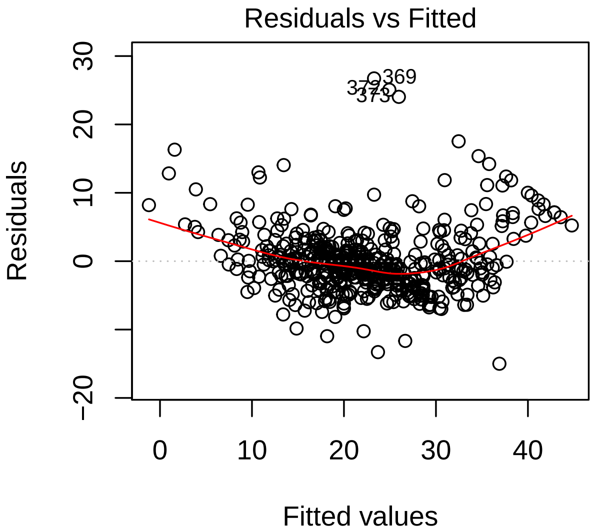
Use the Boston data set in the MASS package of R. These data were collected on 506 census tracts in the Boston area with the intent of modeling how air pollution might affect housing prices. See Harrison and Rubinfeld (1978), Hedonic prices and the demand for clean air. *J. Environ. Economics and Management* **5**, 81-102 for more details. Section 6 of the *Stat 505 2014 Notes* uses these data, but ignores the purpose of the study which was to look at the NOX relationship.
Start with exploratory plots.

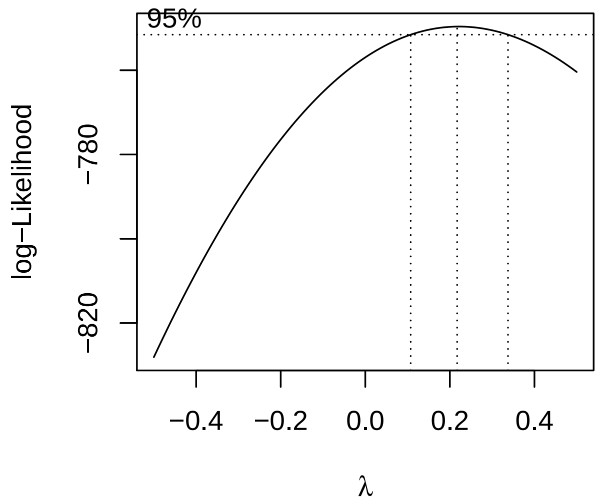


Based on the plots, I logged crime rate to get a "sort of" linear association with log median value. Other variables which appear to be helpfull are rooms, tax, industry, and lstat.

Fit `medv` using all other columns and check to see if you want to transform the response variable.

```
## Loading required package: knitr
```





Judging by residual curvature and by the Box-Cox plot, yes, we do want to work with log median value, not raw median value.

What multicollinearity issues does the full model have?
i. Does transformation make them better or worse?
Explain the impact of transformation.

	Eignvls	cond	(Intercept)	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black
7	0.03	19.70	0	0.04	0.01	0.02	0.02	0.00	0.12	0.00	0.00	0.03	0.03	0.00	0.00
8	0.01	30.01	0	0.05	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.79
9	0.01	33.93	0	0.05	0.02	0.02	0.01	0.00	0.36	0.03	0.07	0.00	0.00	0.00	0.17
10	0.01	45.95	0	0.04	0.15	0.08	0.00	0.01	0.37	0.03	0.45	0.02	0.00	0.00	0.02
11	0.00	71.78	0	0.01	0.00	0.00	0.00	0.01	0.03	0.84	0.34	0.01	0.00	0.00	0.01
12	0.00	95.58	0	0.08	0.08	0.01	0.01	0.58	0.00	0.05	0.14	0.02	0.00	0.15	0.00
13	0.00	131.95	0	0.00	0.06	0.04	0.00	0.41	0.03	0.05	0.00	0.03	0.01	0.85	0.00

We have no VIFs over 10. The highest is 9.7 for radius. The collin function without centering shows highest condition numbers of 116, 49, 40, and 33. I prefer the centered version which finds these issues:

Line	Condition	Term	Collinear with
13	132	ptratio	nox
12	96	nox	ptratio
11	72	age	distance
10	46	rm	distance
9	34	lsat	distance
8	30	black	ptratio, distance?

No change in collinearity if we transform because it depends only on $\mathbf{X}$, *not* $\mathbf{y}$.

ii. Which variables seem to be most highly multicollinear?
NOx and ptratio are strongly correlated, then age, distance, rooms, and lsat seem to form a collinear grouping. I can't tell what `black` is correlated with from this output.
At this point, I rescaled the columns so each has variance 1.

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

	Eignvls	cond	(Intercept)	chas	nox	rm	dis	rad	tax	ptratio	black	lstat
5	0.01	37.00	0	0.00	0.09	0.00	0.00	0	0.19	0.02	0.10	0.05
6	0.00	49.49	0	0.01	0.08	0.71	0.11	0	0.00	0.01	0.02	0.38
7	0.00	57.24	0	0.00	0.18	0.11	0.79	0	0.01	0.04	0.01	0.12
8	0.00	78.82	0	0.00	0.17	0.11	0.10	0	0.09	0.90	0.00	0.00
9	0.00	189.63	0	0.01	0.00	0.04	0.00	1	0.70	0.03	0.01	0.00

Step removed two variables: age, crime, industry, and zone. Again, `vif` finds no issues with `tax` having the highest `vif` of 7 on tax. The `collin` function still finds multicollinearity as the largest condition number is 190 and rows 5 through 9 have large condition numbers. The terms involved have really changed to:

Line	Condition	Term	Collinear with
9	190	rad	tax
8	79	ptratio	nox, tax
7	57	distance	ptratio, nox
6	50	rm	distance, lstat
5	37	tax	ptratio

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

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

Not at all. Using PROC REG, the vif's agree, but with collin, there are three high condition numbers (87, 37, 30.4) which indicate collinearities between 1) intercept with rm, ptratio, 2) room with nox, ptratio, and perhaps lstat, 3) nox with ptratio. Using the `collinoint` option, the highest condition number is 10, so there is no problem.

ii. Try some of the PROC REG selection methods to see if they "fix" the "multicollinearity problem".
Doing plain old 'backward' selection removes the same variables in the same order: age, crime, industry, zone. It leaves a model with adjusted R^2 of 0.758 which is the same as the original adjusted R^2, so this seems to be a good model.

In SAS or R, what terms do we have to remove (or combine) to make the condition numbers all less than 30? Does removing terms (other than NOX) from the model change the estimated NOX effect on prices? The appendix of the original article includes expected signs (+ or -) of relationships between each predictor and house price. The authors say they always saw agreement with their prior beliefs. Did you?

In SAS the reduced model is said to have no collinearity issues, so we're done. In R I still had a problem, so I removed these terms in order: ptratio, radius, distance, rooms, and tax. At this point, the highest condition number is 23, but I feel like I've gutted the model.

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?

Looking at the NOx coefficient estimates from R, I see:

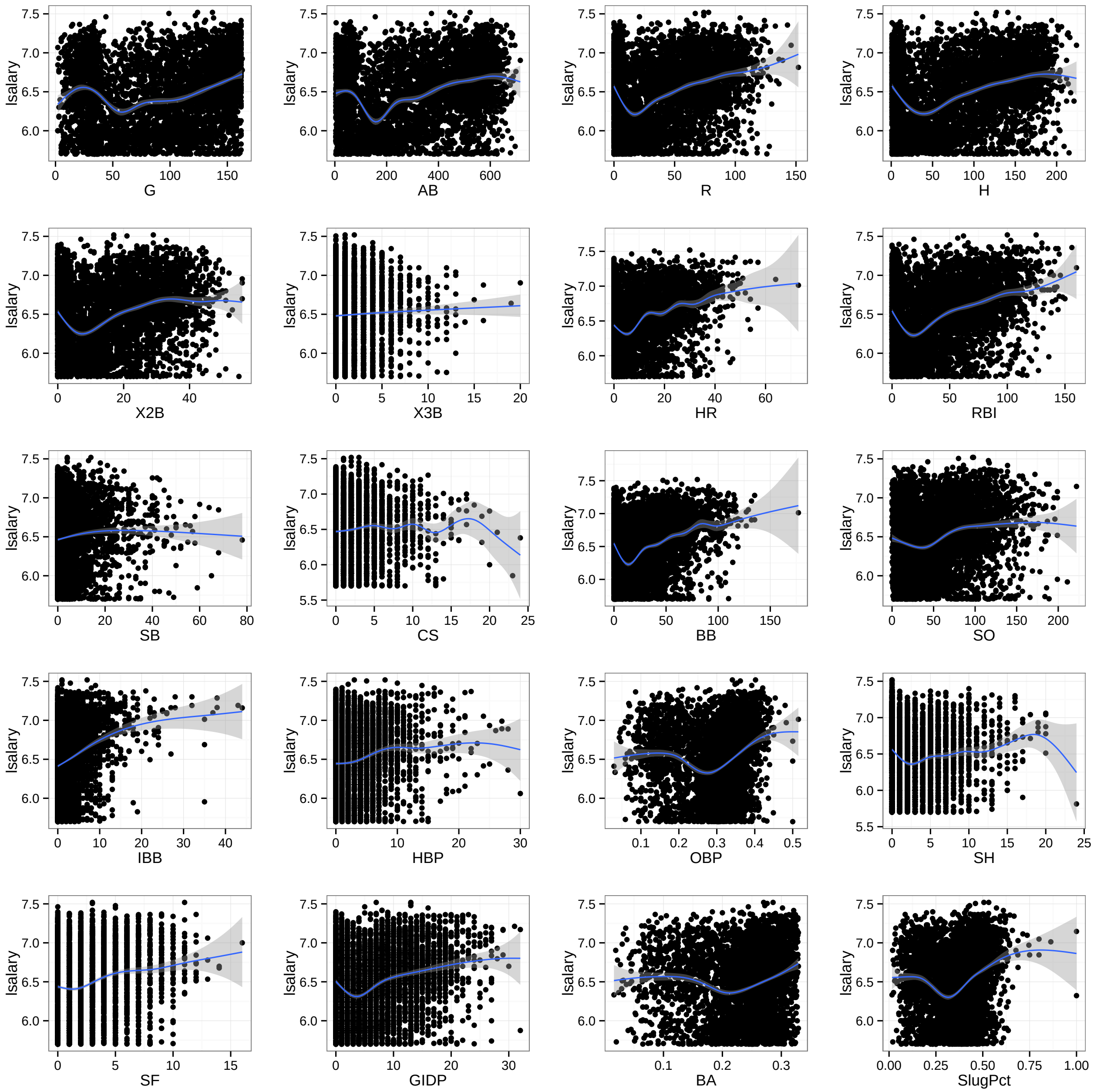
	Estimate	Std. Error
Full	-0.0338820	0.0084196
age crim zone indust	-0.0332102	0.0074885
ptratio	-0.0175137	0.0075843
radius	-0.0176384	0.0075933
distance	0.0033670	0.0061713
rm	0.0058720	0.0065265
tax	-0.0085155	0.0058183

The Full and stepwise models are in close agreement that the nox coefficient is about -0.36 (.007). Removing ptratio and/or radius changes it to -0.020 (0.008), and removing distance changes its sign to +0.003 (0.006) and makes the t-ratio small.

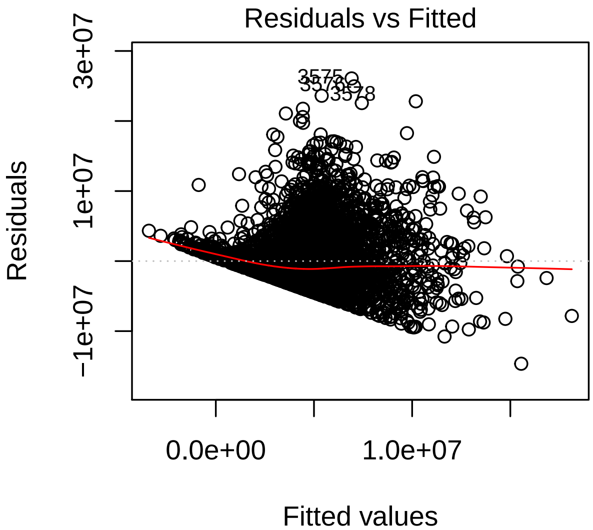
Explain your favorite model and summarize how NOX is related to median house price in these data.

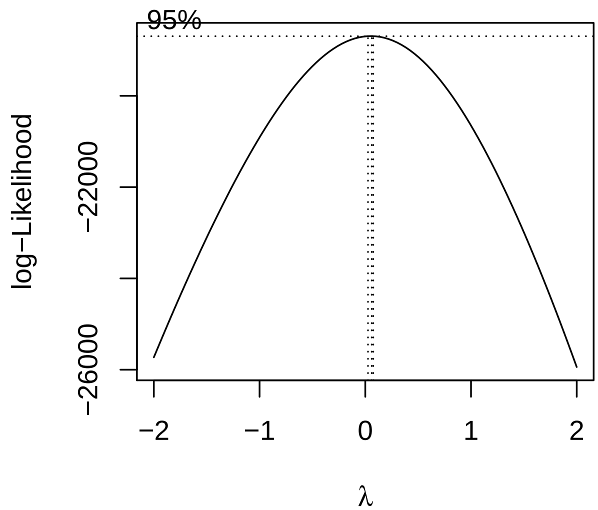
I like the model chosen by the stepwise (AIC in R, F test in SAS) methods. Comparing to the original article, I see that signs do agree, but they logged radius and distance (not crime) and added $(B - .52)^2$ as another predictor, so actual values differ. Their estimated coefficient for NOx was -0.0064 (t ratio = -5.64), so results are roughly similar in that both are negative. For my model, a one unit increase in NOx, keeping other variables fixed, is associated with a decrease in the log10 scale of (-0.048, -0.019) and a multiplicative decrease in median home value of from 0.896 to 0.958.

Use these baseball data which I extracted from the `Lahman` package in R. For more info what is in the columns, see the Batting and battingStats help pages in that package. Start with exploratory plots.
Disclaimer: I know very little about which Baseball stats are most informative about a player's worth. Of course it would be better to work with an expert who can answer these questions. I am treating this as an exercise in using model selection techniques on a model with very little prior knowledge.



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





The residuals show fanning and the Box-Cox "best transform" seems close to 0, meaning log transformation, so I will use log base 10.

What multicollinearity issues does the full model have?

VIFs are over 10 for games, AtBats, runs, hits, doubles, homers, RBI's, Batting Average, Slug% and OBP. Using uncentered collin does not show strong correlations with the intercept column, but I prefer the centered version. The variance proportions (collin with centering) routine flags these terms:

Line	Condition	Term	Collinear.with
21	132	BA	OBP
20	119	AB	H, BA
19	78	SlugPct	BA, OBP
18	39	HR	BABIP
17	36	R	H, BABIP

For more details, see the appendix.

i. Does transformation make them better or worse? Explain the impact of transformation.
   No change. The collinearity depends only on X, not y.

ii. Which variables seem to be most highly multicollinear?

BA, OBP, and Slug% seem highly collinear, and BABIP is also inter-related.

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

The step procedure dropped SF, Hits, Runs, SB, and batting average. The resulting model still has VIFs over 10 for games, atBats, homers and RBIs, and only one line is flagged by collin, indicating a collinearity in OBP and BABIP.

Line	Condition	Term	CollinearWith
16	75	OBP	AB
15	58	AB	OBP

Line	Condition	Term	CollinearWith
14	35	BABIP	HR?

Export the data and import it into SAS. Use PROC REG.
i. Does the collin output agree (non-centered and centered) with the R version?

No. The vifs agree, regular collin shows 6 large condition numbers which indicate these collinearities:

Line	Condition	Term	CollinearWith
22	106	AB	H
21	91	BA	AB, OBP, BABIP
20	58	OBP	SlugPct
19	49	R	HR
18	38	RBI	HR,G
17	35	R	all Above

Using the `collinoint` option, SAS gives a maximum condition number of 50 and only one collinearity issue – between atBats and Hits. That seems much more reasonable

```
ii. Try some of the  PROC REG selection methods to see if they "fix" the "multicollinearity problem".
```

Backwards selection removed SF, H, R, SB, BA in that order – just like R's step function. R^2 decreased from .259 to .258 and the collinearity with collinoint option shows a highest condition number of 17, so the problem is "solved".

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 OPB effect on salaries?

In R I had to remove BABIP, BB, RBI, SH, IBB, and HR and still had one condition number of 50 for OBP by itself. As soon as BABIP is removed from the model, OPB has just one large entry in its column, which makes me think that it is not all that collinear with the others. In addition, VIFs are all less than three, so I like the model with triples, stolen bases, CS, stikeouts, HBP, SF, GIDP and OBP.

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?

Removals	Estimate	Std..Error
None	-1.6921981	0.2537819
SF H R SB BA	-1.8966190	0.2047007
BABIP	-1.5051116	0.1832857
BB	-0.3550480	0.1612036
RBI	-0.3276618	0.1609337
SH	-0.4405901	0.1598016

Removals	Estimate	Std..Error
IBB	-0.1325849	0.1576462
HR	-0.7579841	0.1533047

First of all, it seems odd that OBP has a negative coefficient in all the models. However, all the models contain triples (a rare event), stolen bases, caught stealing, sacrifice flies, HBP and GIDP so it is very hard to interpret any of the coefficients individually – they are highly inter-related. Yes, we do see some changes in th OBP coefficient estimate as we remove other terms. The full model and the model without doubles, BA and SlugPct are very similar in OBP estimates. Also the first three models without BABIP are similar with OBP at about -1. The model which first dropped BABIP is intermediate between the fuller models and this grouping. Dropping HR changed the OBP coefficient estimate again to -.75.

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 don't trust any of these models to actually make any sense. Using the tools we have, I would pick these predictors: triples, homers, RBI, steals, caughtStealing, base on balls, strikeouts, and intentional walks, HBP, SH GIDP and OBP.

*When I fit that model in `lme` with random team effects, nothing changes, as again, teams explain very little variation. Adding random player effects does change the coefficient estimate – **a lot** –and really reduces the residual error variance estimate.*

What have we learned? First of all, current year stats do not do a good job of explaining baseball players' log salaries. Salaries are based on the history of a player's career and predictions of how well they might do in the future. The variables we were given to work with are highly multicollinear, and all models have interpretation issues. I see little utility in using the collinearity as justification for selection of variables. The table of coefficient estimates shows that estimates for one coefficient (OBP) are heavily dependent on the other terms selected for the model. To do a good job with these data, we need to talk to someone who has a good understanding of how salaries are set and we need a clearer vision of how a model might be used. Without that further information, I am very hesitant to say that any one model is 'adequate', or to compare two models.

R Code for #1

```
data(Boston, package="MASS")
Boston$logMed <- log10(Boston$medv)
Boston$crim <- log10(Boston$crim)
write.csv(Boston, file="bostonHouses.csv",quote=F,row.names=F)
require(ggplot2)
require(gridExtra)
bplot1 <- qplot(x= crim, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot2 <- qplot(x= zn, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot3 <- qplot(x= indus, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot4 <- qplot(x= chas, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot5 <- qplot(x= nox, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot6 <- qplot(x= rm, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot7 <- qplot(x= age, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot8 <- qplot(x= dis, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot9 <- qplot(x= rad, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot10 <- qplot(x= tax, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
```

```
bplot11 <- qplot(x= ptratio, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot12 <- qplot(x= black, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()
bplot13 <- qplot(x= lstat, y = logMed, data=Boston, geom=c("point","smooth"))+theme_bw()

grid.arrange(bplot1, bplot2,bplot3,bplot4,bplot5,bplot6,bplot7,bplot8,bplot9,bplot10,bplot11,bplot12,bpl
```

```
require(knitr)
par(mfrow=c(1,2))
source("http://www.math.montana.edu/~jimrc/classes/stat506/Rcode/collin.r")
houseFitFull.raw <- lm(medv ~ .,data = Boston[,1:14])
plot(houseFitFull.raw, which = 1)
MASS::boxcox(houseFitFull.raw,lam=seq(-.5,.5,length=10))
houseFitFull <- update(houseFitFull.raw, logMed ~ ., data = Boston)
```

```
  ## vif(houseFitFull)
  ## kable(round(collin(houseFitFull)[7:14,],2))
  kable(round(collin(houseFitFull, center=TRUE)[7:13,],2))
```

```
BostScale <- Boston
BostScale[,1:13] <-  scale(Boston[,1:13],T,T)
houseFitFull <- update(houseFitFull, data = BostScale)
houseStepDn <- step(houseFitFull, trace = FALSE)
## houseStepDn$call
kable(round( collin(houseStepDn, center=TRUE)[5:9,],2))
```

```
  houseStep2 <- update(houseStepDn, .~. - ptratio)
  kable(round(collin(houseStep2,center=TRUE),2))
```

Eignvls	cond	(Intercept)	chas	nox	rm	dis	rad	tax	black	lstat
8.74	1.000000e+00	0	0.00	0.29	0.00	0.00	0.00	0.00	0.00	0.00
0.21	6.420000e+00	0	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.02	1.907000e+01	0	0.03	0.21	0.01	0.00	0.00	0.00	0.36	0.19
0.02	2.278000e+01	0	0.00	0.02	0.02	0.00	0.00	0.00	0.49	0.28
0.01	3.812000e+01	0	0.00	0.13	0.01	0.00	0.01	0.23	0.11	0.02
0.00	4.968000e+01	0	0.01	0.08	0.74	0.13	0.00	0.00	0.01	0.36
0.00	5.834000e+01	0	0.00	0.27	0.21	0.87	0.00	0.00	0.01	0.14
0.00	1.869900e+02	0	0.00	0.00	0.02	0.00	0.99	0.77	0.01	0.00
0.00	1.823892e+18	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

```
  houseStep3 <- update(houseStepDn, .~. - ptratio - rad)
  kable(round(collin(houseStep3,center=TRUE),2))
```

Eignvls	cond	(Intercept)	chas	nox	rm	dis	tax	black	lstat
7.77	1.000000e+00	0	0.00	0.29	0.00	0.00	0.00	0.00	0.00

Eignvls	cond	(Intercept)	chas	nox	rm	dis	tax	black	lstat
0.19	6.420000e+00	0	0.96	0.00	0.00	0.00	0.00	0.00	0.00
0.02	1.914000e+01	0	0.03	0.20	0.01	0.00	0.01	0.37	0.19
0.01	2.279000e+01	0	0.00	0.02	0.02	0.00	0.00	0.50	0.28
0.00	4.031000e+01	0	0.00	0.14	0.01	0.00	0.98	0.10	0.02
0.00	4.968000e+01	0	0.01	0.08	0.75	0.13	0.01	0.02	0.36
0.00	5.836000e+01	0	0.00	0.27	0.21	0.87	0.00	0.01	0.14
0.00	1.823825e+18	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00

```
houseStep4 <- update(houseStepDn, .~. - ptratio -rad -dis)
kable(round(collin(houseStep4,center=TRUE),2))
```

Eignvls	cond	(Intercept)	chas	nox	rm	tax	black	lstat
6.80	1.000000e+00	0	0.00	0.46	0.00	0.00	0.00	0.00
0.16	6.420000e+00	0	0.96	0.00	0.00	0.00	0.00	0.00
0.02	1.913000e+01	0	0.03	0.31	0.01	0.01	0.37	0.20
0.01	2.279000e+01	0	0.00	0.03	0.02	0.00	0.50	0.29
0.00	4.033000e+01	0	0.01	0.19	0.00	0.99	0.10	0.03
0.00	5.092000e+01	0	0.00	0.01	0.96	0.00	0.02	0.49
0.00	1.823406e+18	1	0.00	0.00	0.00	0.00	0.00	0.00

```
houseStep5 <- update(houseStepDn, .~. - ptratio -rad - dis -rm)
kable(round(collin(houseStep5,center=TRUE),2))
```

Eignvls	cond	(Intercept)	chas	nox	tax	black	lstat
5.83	1.000000e+00	0	0.00	0.46	0.00	0.00	0.00
0.14	6.420000e+00	0	0.96	0.00	0.00	0.00	0.00
0.02	1.942000e+01	0	0.03	0.29	0.01	0.49	0.23
0.01	2.392000e+01	0	0.00	0.07	0.00	0.42	0.70
0.00	4.039000e+01	0	0.01	0.18	0.99	0.09	0.07
0.00	1.823345e+18	1	0.00	0.00	0.00	0.00	0.00

```
houseStep6 <- update(houseStep5, .~. - tax)
kable(round(collin(houseStep6, center=TRUE),2))
```

Eignvls	cond	(Intercept)	chas	nox	black	lstat
4.86	1.000000e+00	0	0.00	0.60	0.00	0.00

Eignvls	cond	(Intercept)	chas	nox	black	lstat
0.12	6.420000e+00	0	0.97	0.00	0.00	0.00
0.01	1.978000e+01	0	0.02	0.31	0.57	0.26
0.01	2.392000e+01	0	0.00	0.09	0.43	0.74
0.00	1.822811e+18	1	0.00	0.00	0.00	0.00

```
 fits <- ls(patt="^house")[c(1,8,3:7)]
 noxCoef <- lapply(fits, function(x) summary(eval(parse(text=x)))$coef["nox",1:2])
 noxcoefTable <- do.call(rbind,noxCoef[c(8,1:7)])
rownames(noxcoefTable) <- c("Full", "age crim zone indust", "ptratio","radius","distance","rm","tax")
kable(noxcoefTable)
```

SAS Code for #1

```
  ## with code chunk option:  comment=""
sasfile <- scan(file="assn5-S15.sas", what = "a", sep = "\n")
cat(paste(sasfile[1:18],"\n"))
```

```
 Proc import  datafile="/folders/myfolders/bostonHouses.csv"
     out=boston   dbms=dlm    replace;
     delimiter=",";
     getnames=YES;
     run;
  /* Proc Print ;
  run;
 */
 title "Boston Housing Prices";
   PROC REG data = boston;
     model  logMed = crim  zn  indus  chas  nox  rm  age
      dis  rad  tax  ptratio  black  lstat/vif collin collinoint;
    model  logMed = crim  zn  indus  chas  nox  rm  age
      dis  rad  tax  ptratio  black  lstat/ selection = backward ;
    Model logMed = chas  nox  rm  dis  rad  tax  ptratio  black
                   lstat/vif collin collinoint;
  run;
```

Full model coefficient estimates:

```
Full model collin output:
```

Full model collinoint output:

Reduced model coefficient estimates:

Reduced model collin output:

Reduced model collinoint output:

R Code for #2

```
baseball <- read.csv("http://www.math.montana.edu/~jimrc/classes/stat506/data/BaseballStats2000.csv")
#names(baseball)
baseball$lsalary <- log10(baseball$salary)
require(ggplot2,quietly = TRUE)
p1=qplot(y=lsalary, x= G, data=baseball) + geom_smooth() +theme_bw()## OK  * > 5.8
p2=qplot(y=lsalary, x= AB, data=baseball) + geom_smooth() +theme_bw()## OK * > 5.8
p3=qplot(y=lsalary, x= R, data=baseball) + geom_smooth() +theme_bw()## OK  * > 5.8
p4=qplot(y=lsalary, x= H, data=baseball) + geom_smooth() +theme_bw()## OK * > 5.8
p5=qplot(y=lsalary, x= X2B, data=baseball) + geom_smooth() +theme_bw()## OK *> 5.8
p6=qplot(y=lsalary, x= X3B, data=baseball) + geom_smooth() +theme_bw()## Poor
p7=qplot(y=lsalary, x= HR, data=baseball) + geom_smooth() +theme_bw()## OK
p8=qplot(y=lsalary, x= RBI, data=baseball) + geom_smooth() +theme_bw()## OK *>5.8
p9=qplot(y=lsalary, x= SB, data=baseball) + geom_smooth() +theme_bw()## weak
p10=qplot(y=lsalary, x= CS, data=baseball) + geom_smooth()+theme_bw()## weak
p11=qplot(y=lsalary, x= BB, data=baseball) + geom_smooth() +theme_bw()## OK  over 5.8
p12=qplot(y=lsalary, x= SO, data=baseball) + geom_smooth() +theme_bw()## weak over 5.8
p13=qplot(y=lsalary, x= IBB, data=baseball) + geom_smooth() +theme_bw()## poor
p14=qplot(y=lsalary, x= HBP, data=baseball) + geom_smooth()+theme_bw() ## weak
p15=qplot(y=lsalary, x= OBP, data=baseball) + geom_smooth() +theme_bw()## quadratic?
p16=qplot(y=lsalary, x= SH, data=baseball) + geom_smooth() +theme_bw()## poor
p17=qplot(y=lsalary, x= SF, data=baseball) + geom_smooth() +theme_bw()## weak
p18=qplot(y=lsalary, x= GIDP, data=baseball) + geom_smooth() +theme_bw()## maybe over 5.8
p19=qplot(y=lsalary, x= BA, data=baseball) + geom_smooth() +theme_bw()## wonky. OK between 0 &  1/3
p20=qplot(y=lsalary, x= SlugPct, data=baseball) + geom_smooth() +theme_bw()## OK between 0 & 1
p21=qplot(y=lsalary, x= BABIP, data=baseball) + geom_smooth()+theme_bw() ## nada
##table(cut(baseball$BA,c(-.1,.01,.33,1)))
require(gridExtra, quietly = TRUE)
grid.arrange(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,p16,p17,p18,p19,p20)
```

```
par(mfrow=c(1,2))
BBfitAll <- lm(salary ~ ., data = baseball[,-c(1:3,26)])
plot(BBfitAll, which=1)
MASS::boxcox(BBfitAll)
```

```
require(knitr,quietly=TRUE)
baseball$lsalary <- log10(baseball$salary)
BBfitAll <- lm(lsalary ~ ., data = baseball[,-c(1:3,25)])
source("http://www.math.montana.edu/~jimrc/classes/stat506/Rcode/collin.r")
kable(matrix(vif(BBfitAll),nrow=1,dimnames=list(NULL,names(coef(BBfitAll)))),digits=2)
```

(Intercept)	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH
0	16.26	109.85	33.32	113.69	10.83	2.22	17.46	29.75	3.34	2.94	9.85	7.44	2.2	1.68	1.4

```r
kable(collin(BBfitAll)[14:22,],digits=2)
```

	Eignvls	cond	(Intercept)	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB
14	0.02	25.97	0.00	0.07	0.01	0.02	0.01	0.53	0.04	0.00	0.00	0.02	0.01	0.02	0.03	0.00
15	0.01	32.28	0.07	0.11	0.00	0.08	0.01	0.07	0.01	0.02	0.01	0.02	0.00	0.01	0.04	0.00
16	0.01	34.53	0.40	0.00	0.00	0.00	0.00	0.03	0.01	0.01	0.03	0.00	0.00	0.01	0.10	0.01
17	0.01	36.11	0.05	0.01	0.00	0.25	0.00	0.01	0.01	0.11	0.60	0.04	0.00	0.04	0.00	0.03
18	0.01	42.36	0.17	0.24	0.00	0.14	0.00	0.05	0.01	0.39	0.22	0.01	0.00	0.03	0.11	0.02
19	0.01	52.90	0.04	0.41	0.06	0.46	0.11	0.00	0.01	0.15	0.09	0.00	0.00	0.10	0.00	0.01
20	0.00	72.73	0.08	0.00	0.01	0.00	0.01	0.06	0.02	0.11	0.00	0.00	0.00	0.21	0.03	0.00
21	0.00	107.17	0.16	0.12	0.76	0.03	0.51	0.00	0.00	0.01	0.03	0.00	0.00	0.05	0.27	0.03
22	0.00	126.60	0.02	0.04	0.16	0.01	0.35	0.04	0.02	0.02	0.00	0.00	0.00	0.10	0.00	0.00

```r
kable(collin(BBfitAll, center=TRUE)[16:21,],digits=2)
```

	Eignvls	cond	(Intercept)	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB
16	0.02	25.75	0	0.11	0.00	0.07	0.05	0.52	0.03	0.06	0.00	0.05	0	0.03	0.14	0.02
17	0.01	37.50	0	0.12	0.01	0.56	0.18	0.01	0.00	0.06	0.05	0.01	0	0.09	0.00	0.01
18	0.01	38.72	0	0.00	0.00	0.23	0.02	0.03	0.03	0.31	0.05	0.01	0	0.07	0.09	0.03
19	0.00	77.45	0	0.00	0.00	0.00	0.02	0.06	0.02	0.11	0.00	0.00	0	0.17	0.03	0.00
20	0.00	119.15	0	0.17	0.88	0.00	0.46	0.00	0.00	0.00	0.01	0.00	0	0.02	0.31	0.02
21	0.00	131.65	0	0.03	0.12	0.00	0.24	0.02	0.01	0.02	0.00	0.00	0	0.15	0.00	0.00

```r
kable(data.frame( Line = 21:17,
                  Condition = c(132,119,78,39,36),
                  Term = c("BA","AB","SlugPct","HR","R"),
                  "Collinear with" = c("OBP","H, BA","BA, OBP","BABIP","H, BABIP")))
```

Line	Condition	Term	Collinear.with
21	132	BA	OBP
20	119	AB	H, BA
19	78	SlugPct	BA, OBP
18	39	HR	BABIP
17	36	R	H, BABIP

```r
BBdown <- step(BBfitAll,trace=FALSE)
kable(matrix(round(vif(BBdown),2),nrow=1,dimnames=list(NULL,names(coef(BBdown)))))
```

```r
kable(collin(BBdown, TRUE), digits=2)
```

```r
kable(data.frame( Line = 16:14,
                  Condition = c(75, 58,35),
                  Term = c("OBP","AB","BABIP"),
                  CollinearWith = c("AB","OBP","HR?")))
```

Line	Condition	Term	CollinearWith
16	75	OBP	AB
15	58	AB	OBP
14	35	BABIP	HR?

```r
BBdown2 <- update(BBdown, .~.-BABIP)
kable(matrix(round(vif(BBdown2),2),nrow=1))
```

0	15.32	33.11	9.35	1.92	13.52	23.41	1.98	6.6	5	2.07	1.6	1.38	3.52	7.35	6.28

```r
kable(round(collin(BBdown2,center=TRUE)[13:15-1,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	RBI	CS	BB	SO	IBB	HBP
12	0.021	19.950	0	0.005	0.000	0.163	0.001	0.241	0.497	0.002	0.002	0.020	0.000	0.007
13	0.014	24.386	0	0.052	0.000	0.122	0.006	0.308	0.160	0.000	0.034	0.399	0.016	0.000
14	0.002	58.209	0	0.379	0.915	0.146	0.039	0.026	0.065	0.068	0.021	0.027	0.011	0.016

```r
BBdown3 <- update(BBdown, .~.-BABIP-BB)
kable(round(collin(BBdown3,center=TRUE)[13:15-2,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	RBI	CS	SO	IBB	HBP	SH
11	0.020	19.693	0	0.005	0.000	0.142	0.001	0.238	0.470	0.003	0.030	0.002	0.007	0.051
12	0.014	23.628	0	0.068	0.000	0.143	0.005	0.354	0.174	0.000	0.343	0.002	0.000	0.049
13	0.003	55.619	0	0.263	0.462	0.045	0.008	0.094	0.060	0.058	0.036	0.004	0.028	0.001

```r
BBdown4 <- update(BBdown, .~.-BABIP-BB-RBI)
kable(round(collin(BBdown4,center=TRUE)[13:15-3,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	CS	SO	IBB	HBP	SH	GIDP
10	0.023	18.173	0	0.131	0.001	0.079	0.001	0.101	0.001	0.581	0.019	0.000	0.014	0.004
11	0.015	22.315	0	0.042	0.001	0.028	0.004	0.189	0.001	0.327	0.003	0.003	0.076	0.016

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	CS	SO	IBB	HBP	SH	GIDP
12	0.003	52.885	0	0.342	0.554	0.129	0.015	0.017	0.054	0.029	0.004	0.023	0.001	0.076

```r
BBdown5 <- update(BBdown, .~.-BABIP-BB-RBI-SH)
kable(round(collin(BBdown5,center=TRUE)[13:15-4,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	CS	SO	IBB	HBP	GIDP	SlugP
9	0.024	18.022	0	0.137	0.001	0.109	0.000	0.067	0.002	0.492	0.018	0.000	0.003	0.05
10	0.017	21.497	0	0.040	0.001	0.029	0.002	0.249	0.000	0.415	0.003	0.002	0.015	0.15
11	0.003	52.857	0	0.352	0.574	0.134	0.017	0.016	0.058	0.029	0.003	0.023	0.078	0.19

```r
BBdown6 <- update(BBdown, .~.-BABIP-BB-RBI-SH-IBB)
kable(round(collin(BBdown6,center=TRUE)[13:15-5,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	HR	CS	SO	HBP	GIDP	SlugPct	OB
8	0.024	17.753	0	0.143	0.001	0.121	0.000	0.064	0.003	0.503	0.000	0.002	0.047	0.01
9	0.017	21.328	0	0.037	0.001	0.031	0.002	0.274	0.000	0.396	0.002	0.015	0.164	0.04
10	0.003	52.412	0	0.332	0.507	0.115	0.012	0.022	0.052	0.024	0.023	0.068	0.247	0.42

```r
BBdown7 <- update(BBdown, .~.-BABIP-BB-RBI-SH-IBB-HR)
kable(round(collin(BBdown7,center=TRUE)[13:15-6,],3))
```

	Eignvls	cond	(Intercept)	G	AB	X2B	X3B	CS	SO	HBP	GIDP	SlugPct	OBP
7	0.028	16.063	0	0.111	0.000	0.043	0.009	0.004	0.842	0.003	0.049	0.000	0.001
8	0.020	18.891	0	0.004	0.001	0.208	0.000	0.008	0.000	0.003	0.009	0.254	0.037
9	0.003	50.315	0	0.072	0.026	0.001	0.000	0.024	0.025	0.014	0.003	0.670	0.937

```r
 fits <- ls(patt="^BB")
 obpCoef <- lapply(fits, function(x) summary(eval(parse(text=x)))$coef["OBP",1:2])
 coefTable <- do.call(rbind, obpCoef[c(8,1:7)])
 coefTable <- data.frame(Removals = c("None", "SF H R SB BA", "BABIP","BB","RBI","SH","IBB","HR"), coef
kable(coefTable)
```

```r
  require(lme4, quietly=TRUE)
  BBlmer1 <- lmer(lsalary ~ X3B + HR + RBI + SB + CS  + SO + IBB +
    HBP + SH + SF + GIDP + OBP+(1|teamID), data = baseball)
  BBlmer2 <- lmer(lsalary ~ X3B + HR + RBI + SB + CS  + SO + IBB +
    HBP + SH + SF + GIDP + OBP+(1|teamID)+(1|playerID), data = baseball)
  kable(anova(BBlmer1,BBlmer2))
  kable(rbind(summary(BBlmer1)$coefficients["OBP",1:2],
```

```
                      summary(BBlmer2)$coefficients["OBP",1:2]))
```

SAS Code for #2

```
  ## with code chunk option:  comment=""
cat(paste(sasfile[19:38],"\n"))
```

```
  PROC IMPORT  datafile="/folders/myfolders/BaseballStats2000.csv"
     out=baseball   dbms=dlm    replace;
     delimiter=",";
     getnames=YES;
     run;
   data baseball;
        set baseball;
        lsalary = log10(salary);
  run;
 /* Proc contents ;
  run;
 */
   title "Baseball Salaries";

   PROC REG data = baseball plots = none;
     model  lsalary = G AB R H X2B X3B HR RBI SB CS BB SO IBB HBP
         SH SF GIDP BA SlugPct OBP BABIP / vif collin  collinoint selection = backwards;
     model lsalary = G X2B X3B HR RBI CS BB SO IBB HBP SH GIDP SlugPct OBP BABIP/ vif collin  collinoin
  run;
 NA
```

Full model coefficient estimates:

`Full model collin output:`

n	eigen	cond	Intercept	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB
1	16.92	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	1.41	3.47	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.03
3	1.20	3.75	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.05	0.04	0.00	0.00	0.00
4	0.45	6.13	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.22
5	0.43	6.25	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.22
6	0.37	6.80	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.02	0.02	0.00	0.00	0.10
7	0.30	7.45	0.00	0.00	0.00	0.00	0.00	0.00	0.67	0.00	0.00	0.05	0.12	0.00	0.00	0.01
8	0.21	8.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.01	0.01	0.04	0.17
9	0.19	9.44	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.02	0.00	0.00	0.01
10	0.15	10.62	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.64	0.74	0.00	0.00	0.00
11	0.10	13.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.01	0.00	0.03	0.20	0.05	0.01

n	eigen	cond	Intercept	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	H
12	0.08	14.83	0.01	0.00	0.00	0.01	0.00	0.09	0.03	0.00	0.00	0.05	0.00	0.03	0.21	0.04	
13	0.07	15.60	0.00	0.01	0.00	0.00	0.00	0.11	0.03	0.02	0.00	0.00	0.00	0.18	0.10	0.11	
14	0.03	23.77	0.02	0.13	0.01	0.01	0.01	0.36	0.02	0.00	0.00	0.01	0.01	0.02	0.06	0.00	
15	0.02	26.15	0.13	0.01	0.00	0.02	0.01	0.15	0.01	0.00	0.01	0.02	0.00	0.00	0.00	0.00	
16	0.02	27.06	0.47	0.00	0.00	0.01	0.00	0.06	0.01	0.01	0.01	0.00	0.01	0.01	0.08	0.01	
17	0.01	34.55	0.05	0.14	0.00	0.35	0.00	0.02	0.03	0.00	0.30	0.07	0.00	0.06	0.06	0.05	
18	0.01	38.19	0.06	0.30	0.00	0.00	0.01	0.08	0.00	0.38	0.47	0.00	0.00	0.00	0.09	0.00	
19	0.01	49.03	0.01	0.25	0.06	0.55	0.08	0.00	0.02	0.31	0.17	0.01	0.00	0.10	0.01	0.02	
20	0.01	57.36	0.09	0.01	0.01	0.01	0.01	0.08	0.02	0.11	0.00	0.00	0.00	0.24	0.03	0.00	
21	0.00	90.82	0.07	0.04	0.30	0.01	0.10	0.01	0.01	0.02	0.01	0.00	0.00	0.12	0.20	0.01	
22	0.00	105.83	0.09	0.11	0.61	0.02	0.78	0.03	0.01	0.01	0.02	0.00	0.00	0.03	0.07	0.01	

Full model collinoint output:

N	eigen	cond	G	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH
1	12.50	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	2.05	2.47	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.04	0.04	0.00	0.00	0.01	0.00	0.06
3	1.54	2.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01
4	0.78	3.99	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.02	0.00	0.00	0.07	0.02	0.27
5	0.67	4.33	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.72	0.01
6	0.65	4.37	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.26	0.01	0.42
7	0.47	5.17	0.00	0.00	0.00	0.00	0.00	0.33	0.01	0.00	0.04	0.11	0.00	0.03	0.09	0.00	0.03
8	0.45	5.28	0.00	0.00	0.00	0.00	0.00	0.23	0.00	0.00	0.01	0.02	0.00	0.07	0.06	0.09	0.00
9	0.40	5.58	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.01	0.01	0.00	0.00	0.09	0.00	0.06
10	0.39	5.65	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.06	0.07
11	0.26	6.87	0.01	0.00	0.00	0.00	0.00	0.00	0.03	0.01	0.04	0.00	0.13	0.02	0.03	0.01	0.01
12	0.22	7.56	0.00	0.00	0.00	0.00	0.01	0.04	0.00	0.00	0.53	0.71	0.01	0.01	0.01	0.00	0.00
13	0.17	8.50	0.02	0.00	0.00	0.00	0.15	0.05	0.01	0.00	0.11	0.02	0.01	0.01	0.01	0.01	0.02
14	0.15	8.98	0.03	0.00	0.02	0.00	0.01	0.01	0.03	0.01	0.07	0.04	0.11	0.27	0.17	0.00	0.00
15	0.10	11.38	0.15	0.00	0.00	0.00	0.40	0.00	0.03	0.01	0.01	0.00	0.02	0.06	0.00	0.01	0.02
16	0.06	14.51	0.24	0.00	0.00	0.00	0.00	0.01	0.01	0.02	0.01	0.00	0.22	0.14	0.02	0.02	0.00
17	0.05	16.46	0.22	0.01	0.10	0.01	0.19	0.04	0.07	0.05	0.06	0.00	0.06	0.08	0.01	0.00	0.02
18	0.03	20.36	0.02	0.00	0.24	0.00	0.00	0.01	0.11	0.45	0.03	0.00	0.25	0.01	0.03	0.04	0.00
19	0.02	23.15	0.05	0.03	0.00	0.03	0.21	0.05	0.32	0.17	0.00	0.00	0.06	0.03	0.01	0.01	0.00
20	0.02	25.94	0.13	0.09	0.58	0.05	0.00	0.02	0.37	0.25	0.01	0.00	0.11	0.07	0.02	0.00	0.00
21	0.00	50.52	0.13	0.86	0.05	0.90	0.02	0.00	0.00	0.04	0.00	0.00	0.00	0.19	0.03	0.00	0.00

Reducing the model:

Reduced model coefficient estimates:

Full model collin output:

Full model collinoint output: