Take-Home Midterm

2022-10-02

Contents

36-617: Applied Linear Models
Fall 2022
Solutions

```
library(arm)  ## includes lme4, MASS, Matrix
library(ggplot2); theme_set(theme_bw())
library(gridExtra) ## to arrange ggplots...
library(GGally) ## for ggpairs...

library(leaps) ## regsubsets(), summary(), coef()
library(car)   ## subsets(), mmps(), vif(), etc.

library(glmnet) ## for glmnet, cv.glmnet, etc.

set.seed(100000)     ## cross-validation for lasso chooses folds randomly, and
                     ## settng the pseudo-random number generator seed manually
                     ## like this ensure that the folds will be the same each
                     ## time I re-run this rmd file...
```

Problem 1.

Return again to the `beauty` data. We will use the variables in the reduced data set `beauty.red`, with the `profevaluaion` variable also removed, and whatever transformations you decided to use in HW04.

Problem 1(a).

Print a summary of the fitted model you obtained after completing problems 3(c) and (d) on HW04, and write a short paragraph interpreting the fitted model for a college dean who is trying to understand what factors, other than teaching quality, might affect course evaluations.

First, I re-create the model that I obtained after completing 3(c) and 3(d) (your model may be different—that's ok!).

```
## Setting up beauty.red as the problem requires...
beauty <- read.csv("ProfEvaltnsBeautyPublic.csv")
prof.loc <- grep("profnumber",names(beauty))
multiclass.loc <- grep("multipleclass",names(beauty))
class.locs <- grep("class",names(beauty))
profeval.loc <- grep("profevaluation",names(beauty))
beauty.red <- beauty[,-c(prof.loc,multiclass.loc,class.locs,profeval.loc)]
```

```
## Removing individual beauty scores and keeping standardized average....
btystd.locs <- grep("btystd",names(beauty.red))[-c(1,8)]
## we are keeping btystdave, which comes first in this list,
## and btystdvariance, which is eighth...
beauty.red <- beauty.red[,-btystd.locs]
```

```
trans.names <- names(beauty.red)[c(3:12,19:20,23)]
## indices slightly different since I already removed "provevaluation"...
```

```
bjpowers <- c(1.0, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 1.0, 1.0, -0.5, 2.0, -0.5, 0.5, 3.0)
names(bjpowers) <- trans.names
## See solutions to hw04 to see where these powers came from...
## They are not the only possible transformations; you may have come up with different ones!
```

```
beauty.trans <- beauty.red
for (i in trans.names) {
  beauty.trans[,i] <- beauty.red[,i]^bjpowers[i]
  names(beauty.trans)[grep(i,names(beauty.trans))] <- paste("t",i,sep=".")
}
names(beauty.trans)[grep("t.btystdave",names(beauty.trans))] <- "btystdave"
## didn't transform btystdave...
names(beauty.trans)[grep("t.courseevaluation",names(beauty.trans))] <- "courseevaluation"
## didn't transform courseevaluation...
names(beauty.trans)[grep("t.age",names(beauty.trans))] <- "age"
## didn't transform age...
```

```
lm.1 <- lm(courseevaluation ~ ., data=beauty.trans)

lm.2 <- update(lm.1, . ~ . - t.beautyf2upper - t.beautyflowerdiv - t.beautyfupperdiv -
                 t.beautym2upper - t.beautymlowerdiv - t.beautymupperdiv -
                 t.profevaluation - t.percentevaluating)
```

And finally, here is a summary of my model after 3(c) and 3(d) from hw04 [your model may be different!]:

```
summary(lm.2)
```

```
##
## Call:
## lm(formula = courseevaluation ~ tenured + minority + age + btystdave +
##     t.didevaluation + female + formal + fulldept + lower + nonenglish +
##     onecredit + t.students + tenuretrack + blkandwhite + t.btystdvariance,
##     data = beauty.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.79811 -0.30955  0.06675  0.37106  1.01035
```

```
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.178124   0.205649  20.317  < 2e-16 ***
## tenured            0.070225   0.073518   0.955 0.339993
## minority          -0.158351   0.077954  -2.031 0.042812 *
## age               -0.007292   0.003187  -2.288 0.022606 *
## btystdave          0.091502   0.033789   2.708 0.007029 **
## t.didevaluation   -1.543028   0.836527  -1.845 0.065762 .
## female            -0.182510   0.052762  -3.459 0.000594 ***
## formal             0.142602   0.069106   2.064 0.039639 *
## fulldept           0.199671   0.084187   2.372 0.018127 *
## lower              0.018096   0.056750   0.319 0.749980
## nonenglish        -0.311596   0.111027  -2.806 0.005227 **
## onecredit          0.533936   0.115457   4.625 4.92e-06 ***
## t.students         2.657462   0.942026   2.821 0.005000 **
## tenuretrack       -0.146123   0.081752  -1.787 0.074551 .
## blkandwhite        0.202276   0.071665   2.823 0.004977 **
## t.btystdvariance -0.043745   0.052827  -0.828 0.408067
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4951 on 447 degrees of freedom
## Multiple R-squared:  0.2297, Adjusted R-squared:  0.2038
## F-statistic: 8.885 on 15 and 447 DF,  p-value: < 2.2e-16
```

I did not write a short paragraph here, in order to show in some detail the reasoning I might use. A short paragraph that summarizes your overall conclusions would be fine here.

The estimated model here is

$$\begin{aligned}(\text{courseevaluation}) \;=\; & (4.18) + (0.53)(\text{onecredit}) + (-0.18)(\text{female}) + (0.20)(\text{blkandwhite}) + \\ & (2.66)\frac{1}{\sqrt{(\text{students})}} + (-0.31)(\text{nonenglish}) + (0.09)(\text{btystdave}) + \\ & (0.20)(\text{fulldept}) + (-0.01)(\text{age}) + (0.14)(\text{formal}) + (-0.16)(\text{minority}) + \\ & (-1.54)\frac{1}{\sqrt{(\text{didevaluation})}} + (-0.15)(\text{tenuretrack}) + (0.07)(\text{tenured}) + \\ & (-0.04)\sqrt{(\text{t.btystdvariance})} + (0.02)(\text{lower})\end{aligned}$$

where I have organized the predictors in decreasing order of statistical importance.

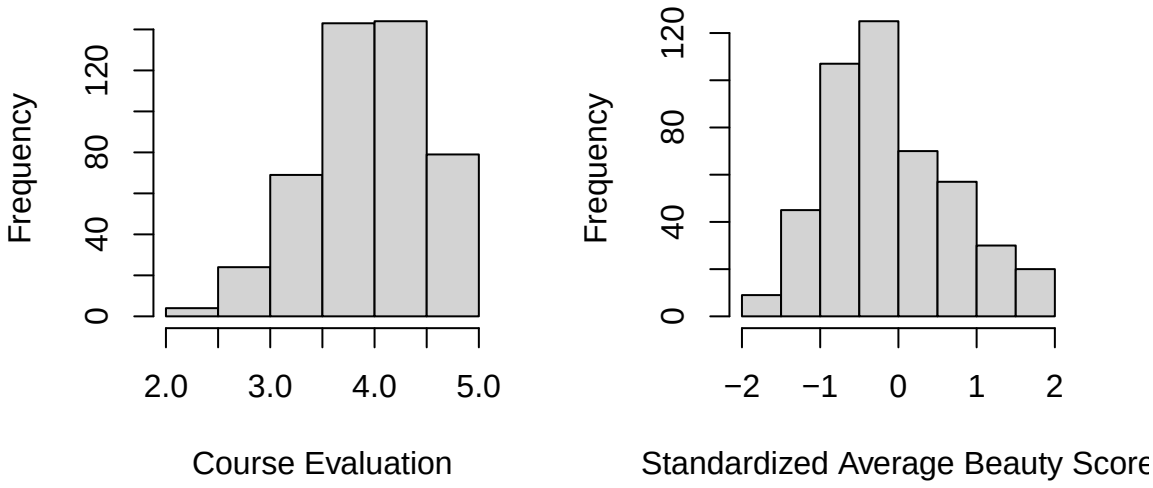
This says, according to the fitted model, that before we know anything, each course "starts with" a baseline evaluation of 4.18 points, on the 5-point course evaluation scale; this is near the median course rating for the college (which is reassuring!), as shown in the left panel below.

```
## Summary Statistics for Course Evaluation:
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.100   3.600   4.000   3.998   4.400   5.000
```

```
## Summary Statistics for Standardized Average Beauty Score:
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.53884 -0.74462 -0.15636 -0.08835  0.45725  1.88167
```



From this baseline score of 4.18, the course evaluation

- *Increases by just over half a point (0.53) on average, for courses that are only one credit (the "fun" or "easy" courses?).*
- *Decreases by roughly a fifth of a point (−0.18) on average, if the instructor is female.*
- *Increases by a fifth of a point if* `blkandwhite=1` *(but we are not sure what* `blkandwhite` *is...).*
- *Increases as the number of students (course size) in the course decreases. The relationship here is a little hard to give a simple interpretation to, but as an example, if the course size goes from 50 to 25, the average increase in course evaluation is about 0.16 points. (I would advise refitting the model with untransformed* `students` *(course size) variable, to get a better handle on this.).*
- *Decreases by about 1/3 of a point on average (−0.31) if the instructor is a non-native English speaker.*
- *Increases by about 1/10 of a point on average (0.09) for each 1-point increase in standardized average beauty rating of the instructor by students. In the right panel above we see that 1 point would be quite a large jump in beauty rating, so even though this variable is statistically important, beauty rating does not seem to have a great practical effect on course rating.*
- *Increases by 1/5 of a point on average if everyone in the instructor's department has their pictures on the web.*
- *Decreases by 1/100 point on average for each additional year older the instructor is. Thus, even though it is statistically important, instructor age doesn't seem to have much effect on course rating.*
- *Increases by about 0.14 if the instructor wears formal attire in their picture on the web.*
- *Decreases by about 0.16 on average if the instructor is a member of a minority.*

The remaining variables, `didevaluation` *(number of students in the class who evaluated the course)* `tenuretrack` *(is the instructor in the tenure track, or just hired to teach),* `tenured` *(is the instructor tenured),* `btystdvariance` *(a measure of how disparate the student ratings of instructor beauty were) and* `lower` *(whether this is a freshman/sophomore course) were not statistically important, and could be ignored.*

One exception to this advice might be `tenured` *and* `tenuretrack`*. If the instructor is tenured, then they are also in the tenure track. i.e.,* `tenured=1` *implies that* `tenuretrack=1`*, as we can see from this table:*

```
with(beauty.red,table(tenured,tenuretrack))
```

```
##         tenuretrack
## tenured   0   1
##       0 102 108
##       1   0 253
```

Therefore it is likely that each of these variables are sapping some effect from the other one, due to this partial collinearity. We could decide to remove one and keep the other, or we could recode the variables as follows:

```
tenured <- tenured  ## i.e., leave "tenured" alone...
ttrack.not.tenured <- tenuretrack - tenured
```

in order to explore whether instructors working toward tenure tend to get higher or lower course ratings than those that already have tenure.

Some overall conclusions that we could make include:

- *One-unit courses get a big boost in course rating.*
- *Beauty rating and age don't seem to matter very much by themselves in the course rating.*
- *Nevertheless, belonging to a department with all instructors' pictures on the web, and dressing well for one's own picture, has a positive effect on course ratings.*
- *Smaller classes tend to get better course ratings.*
- *Being female, a member of a minority, or a non-native speaker of English all have significant negative effects on course ratings, on average.*

Problem 1(b).

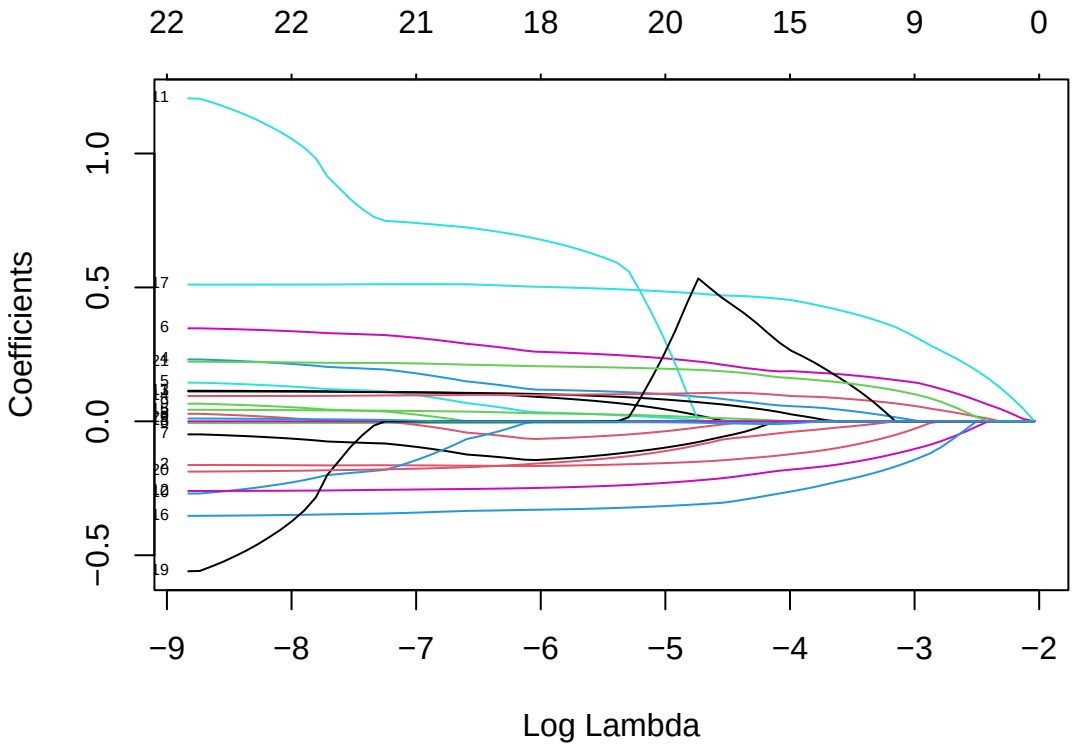
Using the version of `beauty.red` that you used to obtain the model in part (a) [including any transformations that you applied for HW04], apply the lasso to select variables for predicting `courseevaluation` (or a transformation of `courseevaluation` if that's what you used in HW04) for this data set.

- Is it feasible to use shrinkage plots for the lasso, as in lectures 08 and 09? If so, try it. If not, explain why not.
- The function `cv.glmnet` in `library(glmnet)` tries to find an optimal λ by cross-validation using mean-squared prediction error. Read the documentation and try variable selection using `cv.glmnet`. Note that `cv.glmnet` produces both `lambda.min` (the best value found by cross-validation) and `lambda.1se` (the value of λ that is one SE larger than `lambda.min`, which many people use to protect against capitalization on chance).
- You can compare the results of the two values of λ with code like this:

  ```
  result <- cv.glmnet(x,y)
  plot(result)
  c(lambda.1se=result$lambda.1se,lambda.min=result$lambda.min)
  cbind(coef(result),coef(result,s=result$lambda.1se),coef(result,s=result$lambda.min))
  ```

First, we try glmnet's shrinkage plot, using the transformed variables (I have put them all in `beauty.trans`*:*

```
y <- beauty.trans$courseevaluation
X <- as.matrix(beauty.trans[,-grep("courseevaluation",names(beauty.trans))])
lasso.fits <- glmnet(X,y,alpha=1)
plot(lasso.fits,xvar="lambda",label=TRUE)
```




```
data.frame("curve number"=1:dim(X)[2],"variable name"=dimnames(X)[[2]])
```

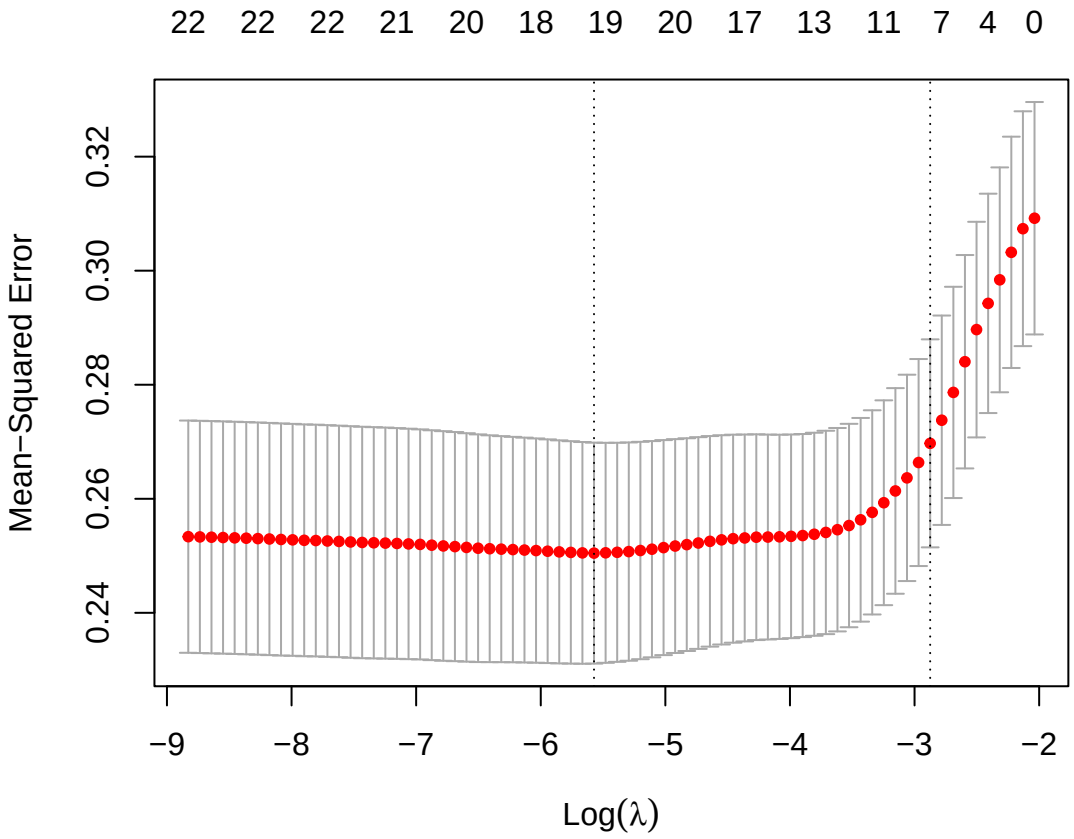
```
##    curve.number       variable.name
## 1             1             tenured
## 2             2            minority
## 3             3                 age
## 4             4    t.beautyf2upper
## 5             5   t.beautyflowerdiv
## 6             6   t.beautyfupperdiv
## 7             7    t.beautym2upper
## 8             8   t.beautymlowerdiv
## 9             9   t.beautymupperdiv
## 10           10           btystdave
## 11           11    t.didevaluation
## 12           12              female
## 13           13              formal
## 14           14            fulldept
## 15           15               lower
## 16           16          nonenglish
## 17           17           onecredit
## 18           18 t.percentevaluating
## 19           19          t.students
## 20           20         tenuretrack
## 21           21         blkandwhite
## 22           22    t.btystdvariance
```

Since there are 22 variables, it is rather hard to figure out which curves go with which variables. Thus the

shrinkage plot is not all that useful.

Now we will try selecting λ by cross-validation:

```
result <- cv.glmnet(X,y)
plot(result)
```




```
## shows estimated cross-valudation MSE at various log-lambda values,
## along with estimated CI for each MSE

round(c(lambda.1se=result$lambda.1se,lambda.min=result$lambda.min),4)
```

```
## lambda.1se lambda.min
##     0.0564     0.0038
```

```
round(log(c(log.lambda.1se=result$lambda.1se,log.lambda.min=result$lambda.min)),4)
```

```
## log.lambda.1se log.lambda.min
##        -2.8746        -5.5725
```

```
cbind(coef(result,s=result$lambda.1se),coef(result,s=result$lambda.min))
```

```
## 23 x 2 sparse Matrix of class "dgCMatrix"
##                                s1           s1
## (Intercept)          3.570581e+00  3.444960e+00
## tenured              .             7.646993e-02
```

```
## minority            -6.130281e-03 -1.633152e-01
## age                  .            -4.495629e-03
## t.beautyf2upper      .             1.133365e-01
## t.beautyflowerdiv    .             2.694991e-02
## t.beautyfupperdiv    1.320847e-01  2.518241e-01
## t.beautym2upper      .            -1.293034e-01
## t.beautymlowerdiv    .            -5.684918e-02
## t.beautymupperdiv    .             .
## btystdave            .             .
## t.didevaluation      .             6.242201e-01
## female              -8.727379e-02 -2.428761e-01
## formal               .             9.543455e-02
## fulldept             4.778214e-02  1.003810e-01
## lower                .             2.699718e-02
## nonenglish          -1.194918e-01 -3.258322e-01
## onecredit            2.840980e-01  4.964934e-01
## t.percentevaluating  1.873768e-05  3.391977e-05
## t.students           .             .
## tenuretrack          .            -1.428527e-01
## blkandwhite          8.629148e-02  2.026144e-01
## t.btystdvariance     .            -5.133563e-04
```

Notice that many more variables are selected for `lambda.min` *than for* `lambda.1se`*. This suggests overfitting (capitalization on chance) and we might prefer the simpler model suggested for* `lambda.1se`*.*

Let's look at the `lambda.1se` *model. The variables kept in the model are:* `minority`*,* `t.beautyfupperdiv`*,* `female`*,* `fulldept`*,* `nonenglish`*,* `onecredit`*,* `t.percentevaluating`*, and* `blkandwhite`*.*

If we refit the model with these variables on the full data set we get

```
lm.3 <- lm(courseevaluation ~ minority + t.beautyfupperdiv + female +
             fulldept + nonenglish + onecredit + t.percentevaluating + blkandwhite,
           data=beauty.trans)
summary(lm.3)
```

```
##
## Call:
## lm(formula = courseevaluation ~ minority + t.beautyfupperdiv +
##     female + fulldept + nonenglish + onecredit + t.percentevaluating +
##     blkandwhite, data = beauty.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.00620 -0.32200  0.06454  0.37007  1.10491
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.171e+00  1.239e-01  25.586  < 2e-16 ***
## minority            -1.874e-01  7.435e-02  -2.520 0.012078 *
## t.beautyfupperdiv    2.592e-01  5.393e-02   4.806 2.09e-06 ***
## female              -2.210e-01  4.968e-02  -4.449 1.09e-05 ***
## fulldept             1.306e-01  8.148e-02   1.602 0.109763
## nonenglish          -3.199e-01  1.017e-01  -3.144 0.001777 **
## onecredit            5.509e-01  1.016e-01   5.422 9.58e-08 ***
## t.percentevaluating  3.647e-05  1.045e-05   3.490 0.000529 ***
## blkandwhite          2.160e-01  6.408e-02   3.371 0.000813 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4899 on 454 degrees of freedom
## Multiple R-squared:  0.234,  Adjusted R-squared:  0.2205
## F-statistic: 17.33 on 8 and 454 DF,  p-value: < 2.2e-16
```

(Note: the results of `cv.glmnet` *can change from run to run, because* `cv.glmnet` *is choosing the folds for cross-validation randomly.)*

Problem 1(c).

Compare the model in part (a) with the model in part (b): Make a table showing which variables remain in the final model for each of two models, and then write a brief paragraph saying which model you would use to help the college dean understand factors other than teaching quality that affect course evaluations, based on this table and any other evidence that seems relevant, and explain your reasoning.

```
part.a.names <- rownames(summary(lm.2)$coef)  ## lm.2 is from part (a)

part.b.names <- rownames(summary(lm.3)$coef)  ## lm.3 is from part (b)

all.vars <- union(part.a.names,part.b.names)

tab <- matrix(NA,ncol=4,nrow=length(all.vars))
dimnames(tab) <- list(all.vars,c("lm.2 Est", "lm.2 pval", "     lm.3 Est", "lm.3 pval"))
tab[part.a.names,1] <- summary(lm.2)$coef[,1]
tab[part.a.names,2] <- summary(lm.2)$coef[,4]
tab[part.b.names,3] <- summary(lm.3)$coef[,1]
tab[part.b.names,4] <- summary(lm.3)$coef[,4]

round(tab,2)
```

```
##                     lm.2 Est lm.2 pval      lm.3 Est lm.3 pval
## (Intercept)             4.18      0.00          3.17      0.00
## tenured                 0.07      0.34            NA        NA
## minority               -0.16      0.04         -0.19      0.01
## age                    -0.01      0.02            NA        NA
## btystdave               0.09      0.01            NA        NA
## t.didevaluation        -1.54      0.07            NA        NA
## female                 -0.18      0.00         -0.22      0.00
## formal                  0.14      0.04            NA        NA
## fulldept                0.20      0.02          0.13      0.11
## lower                   0.02      0.75            NA        NA
## nonenglish             -0.31      0.01         -0.32      0.00
## onecredit               0.53      0.00          0.55      0.00
## t.students              2.66      0.01            NA        NA
## tenuretrack            -0.15      0.07            NA        NA
## blkandwhite             0.20      0.00          0.22      0.00
## t.btystdvariance       -0.04      0.41            NA        NA
## t.beautyfupperdiv         NA        NA          0.26      0.00
## t.percentevaluating       NA        NA          0.00      0.00
```

In the table above, the column of variables on the left lists all the variables in either model. The next two columns show, if the variable is a predictor in lm.2 (from part a), its estimated coefficeint and pvalue for testing whether it is significantly different from zero. An NA is listed if that variable is not in lm.2. The last two columns provide similar information for lm.3 (from part b).

Except for the fact that lm.3 has an individual student's beauty scores, whereas lm.2 has the average beauty score, the variables in lm.3 are a subset of the variables in lm.2 (even if we exclude the variables with nonsignificant p-values). There is a good argument for keeping the model with more variables, since that gives you more to talk about with the dean (and if you chose a model with more variables for that reason, that would be fine). However, in this case I like the smaller model since it focuses on important variables, but I would modify it in two ways: (1) replace the individual beauty rating with the `btystdave` *and (2) put* `tenured` *and* `tenuretrack` *back in the model so that we could have the discussion about tenured, vs tenure-track but not tenured, vs instructor-only.*

(I am not sure what I would do with the formal and fulldept variables, which seem to function differently in the two models.)

Problem 2.

The file `cdi.dat`, in the same Canvas folder as this midterm assignment sheet, is taken from Kutner et al. (2005)[1]: It provides selected county demographic information (CDI) for 440 of the most populous counties in the United States. Each line of the data set has an identification number with a county name and state abbreviation and provides information on 14 variables for a single county. Counties with missing data were deleted from the data set. The information generally pertains to the years 1990 and 1992. The definitions of the variables are given in Table~1.

Table 1: Variable definitions for CDI data from Kutner et al. (2005). *Original source:* Geospatial and Statistical Data Center, University of Virginia.

Variable Number	Variable Name	Description
1	Identification number	1–440
2	County	County name
3	State	Two-letter state abbreviation
4	Land area	Land area (square miles)
5	Total population	Estimated 1990 population
6	Percent of population aged 18–34	Percent of 1990 CDI population aged 18–34
7	Percent of population 65 or older	Percent of 1990 CDI population aged 65 or old
8	Number of active physicians	Number of professionally active nonfederal physicians during 1990
9	Number of hospital beds	Total number of beds, cribs, and bassinets during 1990
10	Total serious crimes	Total number of serious crimes in 1990, including murder, rape, robbery, aggravated assault, burglary, larceny-theft, and motor vehicle theft, as reported by law enforcement agencies
11	Percent high school graduates	Percent of adult population (persons 25 years old or older) who completed 12 or more years of school
12	Percent bachelor's degrees	Percent of adult population (persons 25 years old or older) with bachelor's degree
13	Percent below poverty level	Percent of 1990 CDI population with income below poverty level
14	Percent unemployment	Percent of 1990 CDI population that is unemployed
15	Per capita income	Per-capita income (i.e. average income per person) of 1990 CDI population (in dollars)
16	Total personal income	Total personal income of 1990 CDI population (in millions of dollars)
17	Geographic region	Geographic region classification used by the US Bureau of the Census, NE (northeast region of the US), NC (north-central region of the US), S (southern region of the US), and W (Western region of the US)

[1]Kutner, M.H., Nachsheim, C.J., Neter, J. & Li, W. (2005) *Applied Linear Statistical Models, Fifth Edition*. NY: McGraw-Hill/Irwin.

Problem 2(a).

Data description.

- Make a table or tables showing appropriate summary statistics for each variable in the data set. Note that summary statistics for continuous variables should be different from the summary statistics for categorical variables.
- Indicate where (in which variables) there is missing data (NA's), if any, how much there is (in each variable) and why it might be there.
- Make some appropriate descriptive EDA plots to illustrate any important features of the variables or possible important relationships among them.

```
cdi <- read.table("cdi.dat",header=T)

apply(cdi,2,function(x) any(is.na(x)))
```

```
##             id         county          state      land.area            pop
##          FALSE          FALSE          FALSE          FALSE          FALSE
##      pop.18_34    pop.65_plus        doctors      hosp.beds         crimes
##          FALSE          FALSE          FALSE          FALSE          FALSE
##    pct.hs.grad   pct.bach.deg  pct.below.pov      pct.unemp per.cap.income
##          FALSE          FALSE          FALSE          FALSE          FALSE
##     tot.income         region
##          FALSE          FALSE
```

As we can see above, there is no missing data in the data set, as promised in the problem statement.

I will not summarize the variables `id` *and* `county`*:*

- `id` *is just the row number of each observation in the data set and is not interesting.*
- `county` *is nearly unique for each row (the combination "county, state" is unique for each row; some county names occur in two or three different states), and so no summary would be helpful.*

The variables `state` *and* `region` *are categorical and I will just make a table of counts for each:*

```
table(cdi$state)
```

```
##
## AL AR AZ CA CO CT DC DE FL GA HI ID IL IN KS KY LA MA MD ME MI MN MO MS MT NC
##  7  2  5 34  9  8  1  2 29  9  3  1 17 14  4  3  9 11 10  5 18  7  8  3  1 18
## ND NE NH NJ NM NV NY OH OK OR PA RI SC SD TN TX UT VA VT WA WI WV
##  1  3  4 18  2  2 22 24  4  6 29  3 11  1  8 28  4  9  1 10 11  1
```

```
length(unique(cdi$state))
```

```
## [1] 48
```

```
table(cdi$region)
```

```
##
##  NC  NE   S   W
## 108 103 152  77
```

Whereas the data is fairly evenly spread across regions (except that West is a little light), there is a lot of variation in the number of counties included from each state, and two states appear to have zero counties (they are not included in the data at all). The `state` *variable causes problems in some regression models, and is difficult to incorporate into* `regsubsets` *and* `glmnet` *variable selection and so I will leave it out of regression analyses below.*

The other variables are continuous and I will provide 5-number summaries, means and SD's to sumamrize them:

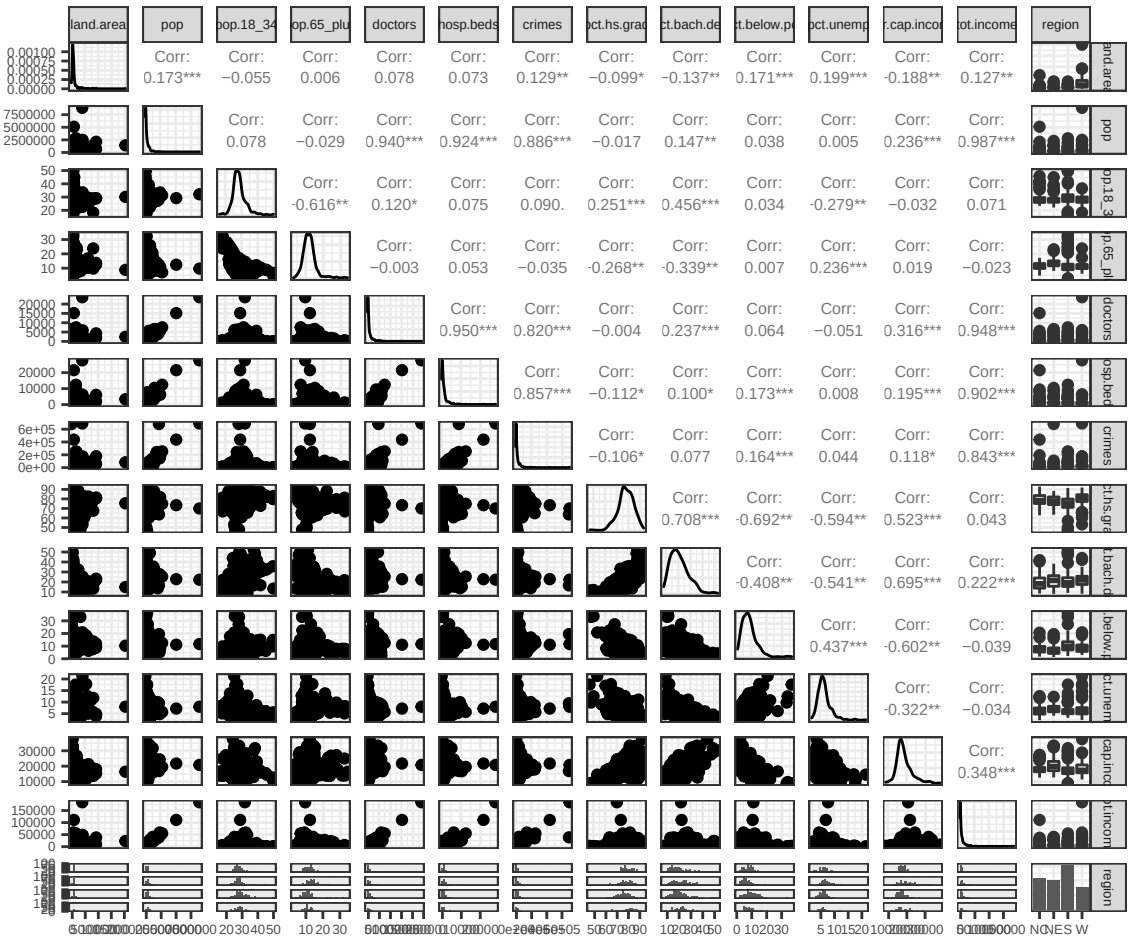
```
summary.with.sd <- function(x) {
  su <- summary(x)
  return(c(su[1:4],SD=sd(x),su[5:6]))
}
old.width <- options()$width
options(width=200)
round(t(apply(cdi[,-c(1:3,17)],2,summary.with.sd)),2)
```

```
##                    Min.   1st Qu.    Median      Mean        SD   3rd Qu.      Max.
## land.area          15.0    451.25    656.50   1041.41   1549.92    946.75   20062.0
## pop            100043.0 139027.25 217280.50 393010.92 601987.02 436064.50 8863164.0
## pop.18_34          16.4     26.20     28.10     28.57      4.19     30.02      49.7
## pop.65_plus         3.0      9.88     11.75     12.17      3.99     13.62      33.8
## doctors            39.0    182.75    401.00    988.00   1789.75   1036.00   23677.0
## hosp.beds          92.0    390.75    755.00   1458.63   2289.13   1575.75   27700.0
## crimes            563.0   6219.50  11820.50  27111.62  58237.51  26279.50  688936.0
## pct.hs.grad        46.6     73.88     77.70     77.56      7.02     82.40      92.9
## pct.bach.deg        8.1     15.28     19.70     21.08      7.65     25.33      52.3
## pct.below.pov       1.4      5.30      7.90      8.72      4.66     10.90      36.3
## pct.unemp           2.2      5.10      6.20      6.60      2.34      7.50      21.3
## per.cap.income   8899.0  16118.25  17759.00  18561.48   4059.19  20270.00   37541.0
## tot.income       1141.0   2311.00   3857.00   7869.27  12884.32   8654.25  184230.0
```

```
options(width=old.width)
```

It seems valuable to make a scatter plot matrix with density plots for all the data except for `id` *and* `county` *(I will also skip* `state` *since it has too many levels for* `ggpairs` *to deal with):*

```
ggpairs(cdi[,-c(1:3)], upper = list(continuous = wrap("cor", size = 2))) +
  theme(text=element_text(size=6))  ## theme affects the labels at the sides...
```

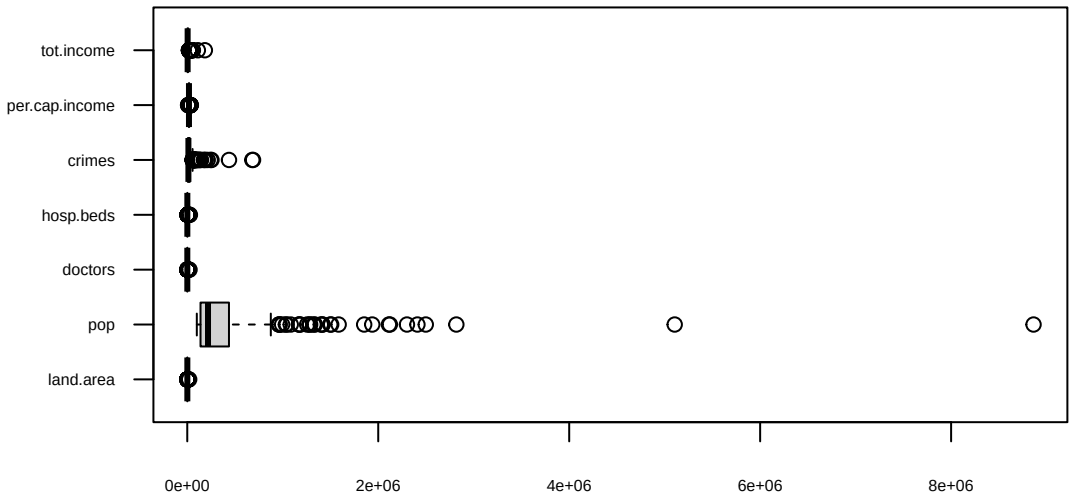


We can see (either from the plot or from the numerical summaries) that several variables have substantial right-skew: `land.area`, `pop`, `doctors`, `hosp.beds`, `crimes`, `per.cap.income`, *and* `tot.income`. *Some other variables have some right-skew as well but it doesn't seem as severe.*

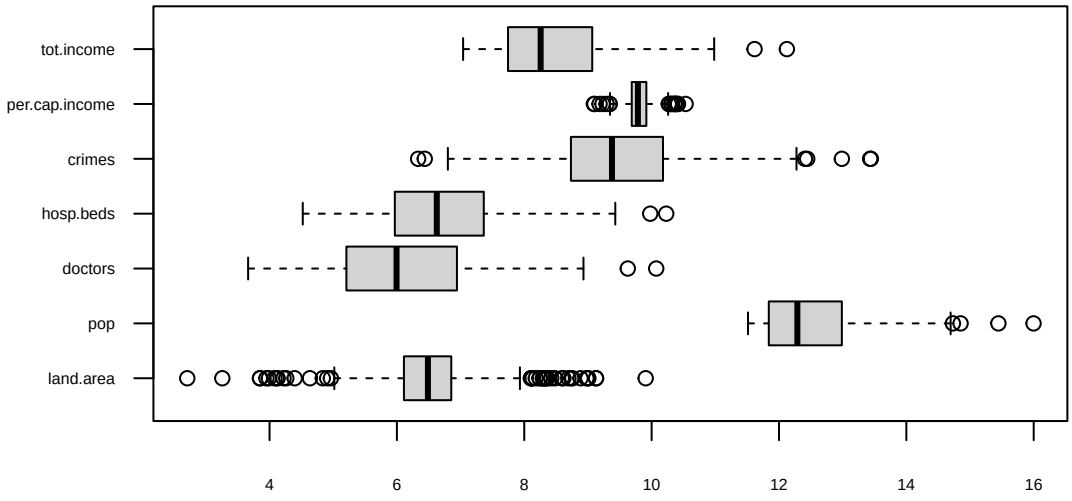
What I have done above is enough to answer the question. Below I explore transformations just a little bit more, but this is not required to answer the question.

From the plots below it appears that applying log(x) to these variables cleans them up about as much as we could expect, though some variables still have some severe outliers.

```
skew.candidates <- c("land.area", "pop", "doctors", "hosp.beds", "crimes",
                     "per.cap.income", "tot.income")
boxplot(cdi[,skew.candidates],horizontal=TRUE,las=1,cex.axis=0.5)
```




```
boxplot(log(cdi[,skew.candidates]),horizontal=TRUE,las=1,cex.axis=0.5)
```



Problem 2(b).

Build a regression model that predicts per-capita income from crimes and region of the country (using ony these three variables, not the full set of variables in the data set). Should there be any interactions in the model? What does your model say about the relationship between per-capita income and crimes? Do your answers change, depending on whether you use number of crimes, or "per-capita crime" = (number of crimes)/(population) as a measure of crimes? If so, which one best describes the relationship between per-capita income and crimes? Why? Show the fitted model results and explain your answer to these questions in terms of those results (as well as any economics knowledge you may have).

Below are the additive and interactive models predicting per.cap.income from crime and region. We can see from the F test (anova table), from AIC, or from BIC, that the additive model, `lm.1` *is preferred. According to* `summary(lm.1)`*, the NE (Northeast) region of the country has higher income than the North Central (NC, the missing/baseline category) region and the other regions don't have significantly different per.cap.income, on average, from the NC region. Raw crime counts seem to have a strong association with per.cap.income.*

However, the residual diagnostic plots show that lm.1 has some extreme outliers, at least one of which is strongly influential according to Cook's Distance, that may be affecting these results.

```
lm.1 <- lm(per.cap.income ~ crimes + region, data=cdi)
lm.2 <- lm(per.cap.income ~ crimes * region, data=cdi)

anova(lm.1,lm.2)
```

```
## Analysis of Variance Table
##
## Model 1: per.cap.income ~ crimes + region
## Model 2: per.cap.income ~ crimes * region
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
## 1    435 6501791845
## 2    432 6438799739  3  62992106 1.4088 0.2396
```

```
cbind(AIC(lm.1,lm.2),BIC(lm.1,lm.2))
```

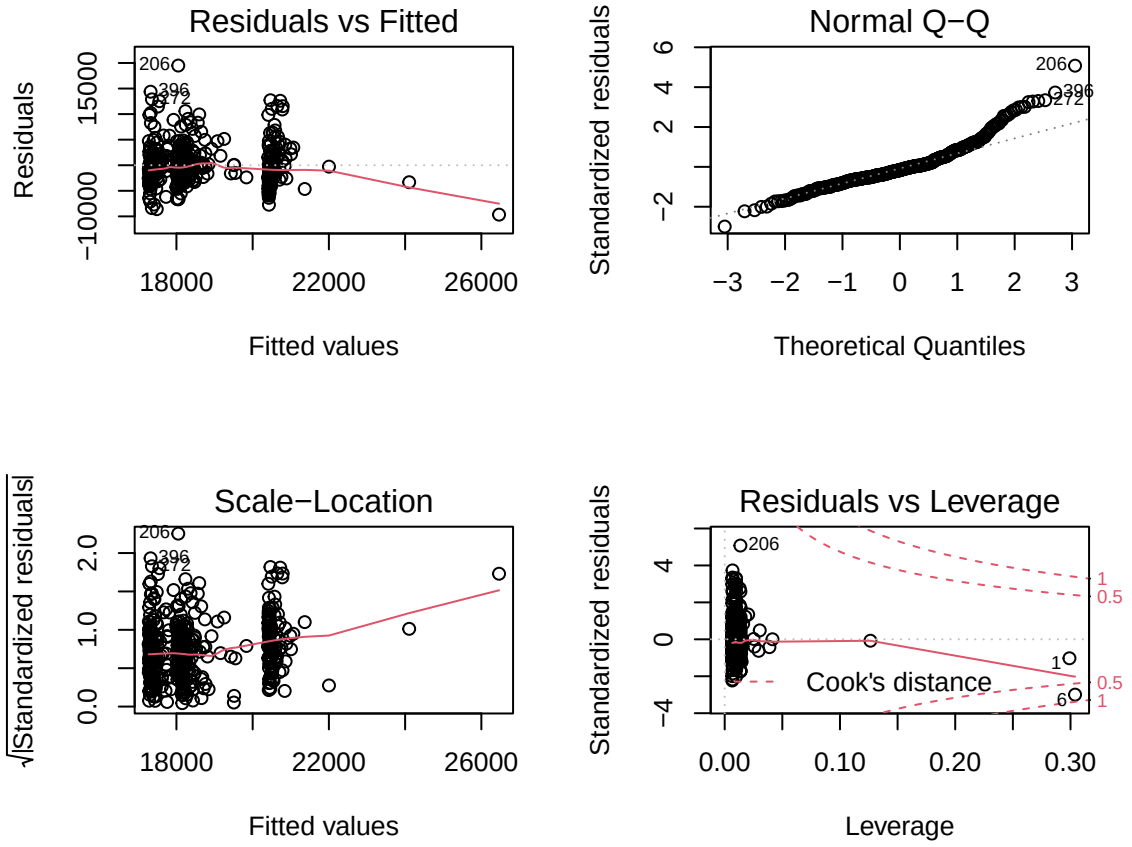
```
##      df      AIC df      BIC
## lm.1  6 8524.436  6 8548.957
## lm.2  9 8526.153  9 8562.934
```

```
summary(lm.1)
```

```
##
## Call:
## lm(formula = per.cap.income ~ crimes + region, data = cdi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9661.0 -2260.7  -618.3  1650.0 19492.6
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.811e+04  3.784e+02  47.846  < 2e-16 ***
## crimes       8.915e-03  3.188e-03   2.797  0.00539 **
## regionNE     2.286e+03  5.325e+02   4.293 2.17e-05 ***
## regionS     -8.606e+02  4.868e+02  -1.768  0.07782 .
## regionW     -1.428e+02  5.796e+02  -0.246  0.80548
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3866 on 435 degrees of freedom
## Multiple R-squared:  0.1011, Adjusted R-squared:  0.09288
## F-statistic: 12.24 on 4 and 435 DF,  p-value: 1.946e-09
```

```
par(mfrow=c(2,2))
plot(lm.1)
```



And here are the additive and interactive models predicting per.cap.income from per.cap.crime and region. Once again, the additive model "wins". From the `summary(lm.3)` *table, we see that the story for regions is the same—NE seems to have substantially higher incomes than the rest—but the story for per.cap.crime is now different: per-capita crime does not seem to be significantly associated with per-capita income.*

The residual diagnostic plots still show some high outliers, but only one observation, #6, that is close to being strongly influential on the fit. The fit is not yet perfect (perhaps transformations would help) but the story is interestingly different from what we got when we used crime count alone.

```
per.cap.crime <- with(cdi,crimes/pop)

lm.3 <- lm(per.cap.income ~ per.cap.crime + region, data=cdi)
lm.4 <- lm(per.cap.income ~ per.cap.crime * region, data=cdi)

anova(lm.3,lm.4)
```

```
## Analysis of Variance Table
##
```

```
## Model 1: per.cap.income ~ per.cap.crime + region
## Model 2: per.cap.income ~ per.cap.crime * region
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
## 1    435 6609753963
## 2    432 6607856753  3   1897210 0.0413 0.9888
```

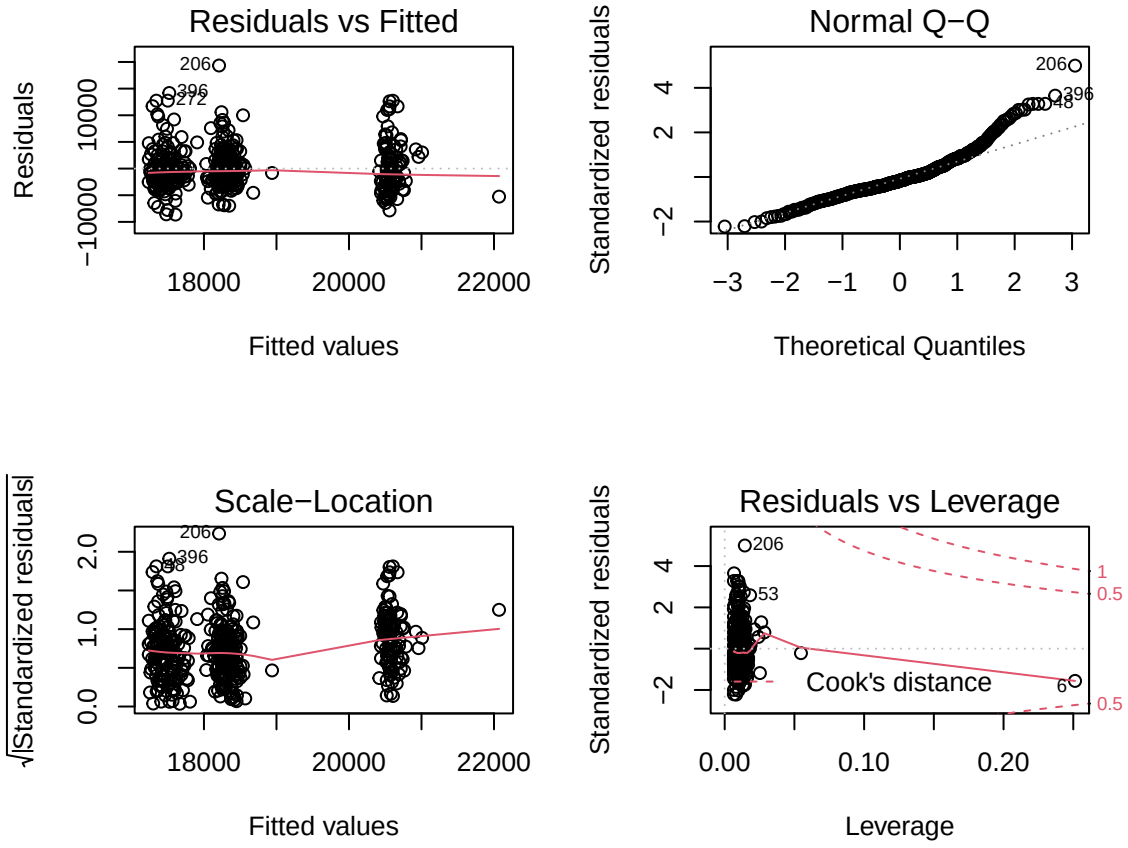
```
cbind(AIC(lm.3,lm.4),BIC(lm.3,lm.4))
```

```
##      df      AIC df      BIC
## lm.3  6 8531.682  6 8556.203
## lm.4  9 8537.556  9 8574.337
```

```
summary(lm.3)
```

```
##
## Call:
## lm(formula = per.cap.income ~ per.cap.crime + region, data = cdi)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8634  -2300   -631   1710  19333
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   18006.04     537.04  33.528  < 2e-16 ***
## per.cap.crime  5773.20    7520.41   0.768   0.4431
## regionNE       2354.70     541.97   4.345 1.74e-05 ***
## regionS        -927.45     512.31  -1.810   0.0709 .
## regionW         -34.92     586.03  -0.060   0.9525
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3898 on 435 degrees of freedom
## Multiple R-squared:  0.08622,    Adjusted R-squared:  0.07782
## F-statistic: 10.26 on 4 and 435 DF,  p-value: 6.007e-08
```

```
par(mfrow=c(2,2))
plot(lm.3)
```



Finally, to answer the question about which model best describes the relationship between crime and per.cap.income, actually both models say there is very little relationship:

- *In* `lm.1`*, the variable* `crimes` *is a significant predictor, but the coefficient is 0.0089, which means for 100 additional crimes, per.cap.income would only be expected to increase by $0.89 (89 cents). So although* `crimes` *is statistically significant, it doesn't appear to be practically significant.*
- *In* `lm.3`*, the variable* `per.cap.crime` *is not even close to being a significant predictor.*

It is also worth noting that the R^2's in both `lm.1` *and* `lm.3` *are pitifully small, so neither crime (however it is measured) nor region is doing a very good job of accounting for the variation in per-capita income.*

Problem 2(c).

Use methods we have discussed in class and/or methods from Sheather Chapters 5, 6 & 7 (including, as needed: transformations, interactions, variable selection, residual diagnostic plots, fit indices, added-variable or marginal model plots, etc.) to find the multiple regression model predicting per-capita income from the other variables, that makes the "best'' tradeoff between the following criteria:

- Reflects the social science and the meaning of the variables
- Satisfies modeling assumptions
- Clearly indicated by the data
- Can be explained to someone who is more interested in social, economic and health factors than in mathematics and statistics.

Organize your work so that it tells an interesting data analysis story that a statistican (like me!) might like, and be sure to explain why and how you arrived at a final model.

(**Note:** No matter what you do, you are likely to be unhappy with some or all of these criteria; the better you make one criterion, the worse another is likely to get. So you will have to find a compromise or tradeoff between these criteria. Explain how you decided to make the tradeoff(s) you made.)

Following an approach we have used before, I will

- *First consider transformations to pull in outliers and symmetrize distributions somewhat, in order to reduce the likelihood of high leverage or influentual points;*
- *Next try some general variable selection, beginning with a "kitchen sink" model, and proceeding to other methods.*
- *Look at diagnostic plots to see if improvements in distribution or functional form are needed;*
- *Think about the practical meanings of the variables and the model(s) to see if any further adjustments are needed.*

Transformations:

In general I am mentally biased against a lot of transformations, and also biased against complicated searches for really good transformations.

Since I found a few log-transformations "by eye" that seemed to help somewhat when I was looking at summaries of the data, I will try those transformations here.

Recall that, above, I defined

```
skew.candidates <- c("land.area", "pop", "doctors", "hosp.beds", "crimes",
                     "per.cap.income", "tot.income")
```

and found that log-transformations seemed to provide some relief from extreme right-skewing in these variables.

So...

```
cdi.trans <- cdi
cdi.trans[,skew.candidates] <-  log(cdi[,skew.candidates])
skew.cand.locs <- names(cdi.trans) %in% skew.candidates
names(cdi.trans)[skew.cand.locs] <- paste0("log.",skew.candidates)
cdi.trans <- cdi.trans[,-grep("id",names(cdi.trans))]
cdi.trans <- cdi.trans[,-grep("county",names(cdi.trans))]
cdi.trans <- cdi.trans[,-grep("state",names(cdi.trans))]
```

Variable Selection: The Initial Kitchen Sink Model

Let's start with the "kitchen sink" model...

```
cdi.0 <- lm(log.per.cap.income ~ ., data=cdi.trans)
summary(cdi.0)
```

```
##
## Call:
## lm(formula = log.per.cap.income ~ ., data = cdi.trans)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.730e-04 -4.586e-05  3.740e-06  5.420e-05  3.343e-04
##
## Coefficients:
```

```
##                   Estimate Std. Error    t value Pr(>|t|)
## (Intercept)     1.382e+01  3.010e-04  45893.977   <2e-16 ***
## log.land.area  -6.637e-06  7.623e-06     -0.871    0.384
## log.pop        -1.000e+00  6.536e-05 -15299.141   <2e-16 ***
## pop.18_34       4.306e-07  2.037e-06      0.211    0.833
## pop.65_plus     2.530e-07  1.921e-06      0.132    0.895
## log.doctors    -2.265e-05  1.931e-05     -1.173    0.242
## log.hosp.beds   9.098e-06  1.724e-05      0.528    0.598
## log.crimes      1.187e-05  1.510e-05      0.786    0.432
## pct.hs.grad    -1.526e-07  1.592e-06     -0.096    0.924
## pct.bach.deg    7.351e-07  1.743e-06      0.422    0.673
## pct.below.pov  -6.753e-07  2.603e-06     -0.259    0.795
## pct.unemp       3.180e-06  3.241e-06      0.981    0.327
## log.tot.income  1.000e+00  6.464e-05  15470.149   <2e-16 ***
## regionNE       -5.810e-06  1.682e-05     -0.345    0.730
## regionS        -2.812e-06  1.649e-05     -0.171    0.865
## regionW         7.912e-06  2.048e-05      0.386    0.699
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0001072 on 424 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.089e+08 on 15 and 424 DF,  p-value: < 2.2e-16
```

This is weird; look at some of these numbers: R^2 *and* R^2_{adj} *both equal 1, residual standard error is* $s = 0.0001072$*, and only two variables matter,* `log.pop` *and* `log.tot.income`*, and their* $\hat{\beta}$*'s are essentially* -1 *and 1 with tiny SE's. All the others are not even close to being significant so we can guess that more formal variable selection would lead to a fitted model like this (with essentially no error):*

$$\textit{(log.per.cap.income)} = 1.382 - \textit{(log.pop)} + \textit{(log.tot.income)}$$

What the heck is going on???

If we exponentiate both sides, we get

$$\textit{(per-capita income in each county)} = (3.98) \cdot \frac{\textit{(total income in the county)}}{\textit{(population of the county)}}$$

The very definition of per-capita income is total income over population size, (and the factor of 4 (≈ 3.98*) is due to the fact that not all the units are the same: per-capita income is in dollars, total income is in millions of dollars, etc.).*

So this is not a very useful model for the social scientist. It just says we rediscovered the formula for per-capita income! To proceed we should take one or both of the variables `log.pop` *and* `log.tot.income` *out of the model.*

I looked at summaries of the regressions taking out only `log.pop`*, only* `log.tot.income`*, or both. There isn't a purely mathematical reason for preferring one or the other of these three models, but I did try to guess what might make sense. For example, when I took out* `log.tot.income` *but left in* `log.pop`*, then* `log.crimes` *had a significant positive coefficient. Having income go up with crimes doesn't really make sense (and doesn't agree with part (b) of this problem! This seems to reflect a collinearity between total number of crimes and total population and not give us a really interpretable model.*

Finally I decided to take them both out, since they are "encoded" in the response variable `log.per.cap.income` *anyway. Since I'm not going to use them anymore, I also took them out of the data frame, to make further variable selection, etc., easier.*

```
cdi.trans <- cdi.trans[,!(names(cdi.trans) %in% c("log.pop","log.tot.income"))]
```

So my initial model will be:

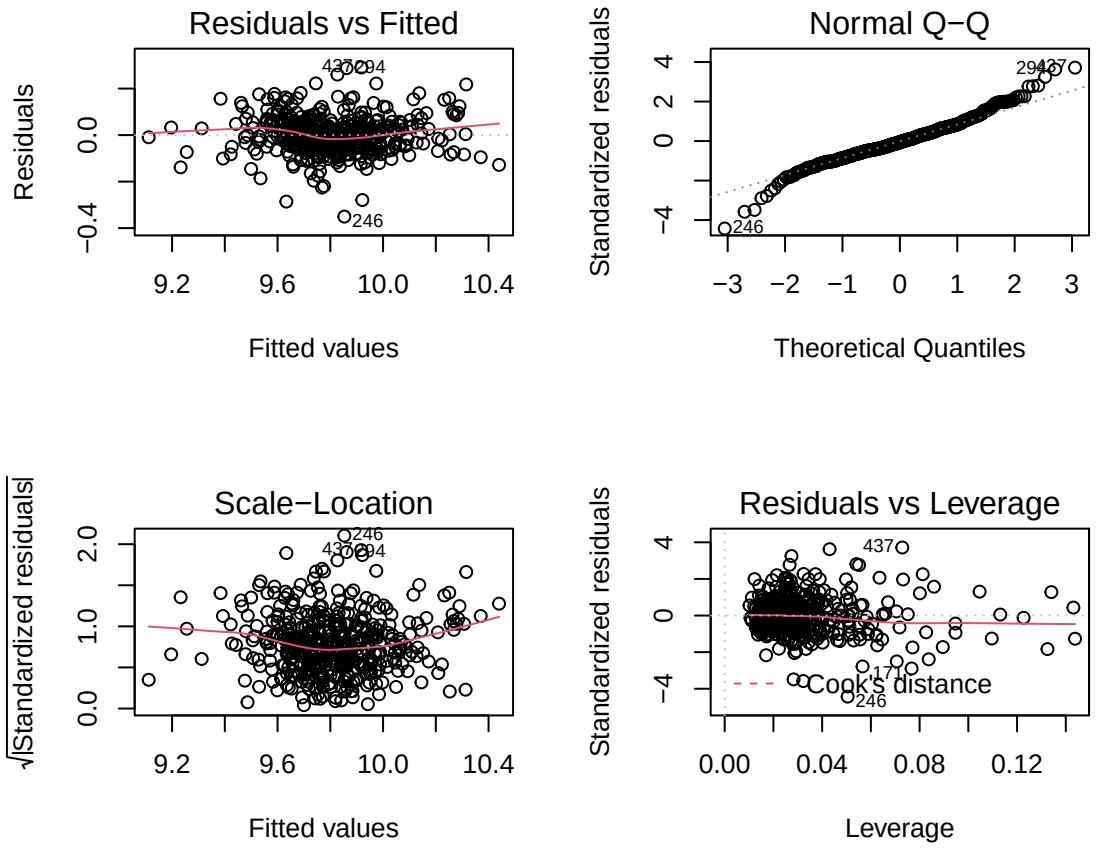
```
cdi.1 <- lm(log.per.cap.income ~ ., data=cdi.trans)
summary(cdi.1)
```

```
##
## Call:
## lm(formula = log.per.cap.income ~ ., data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35060 -0.04671 -0.00502  0.04525  0.29066
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.353232   0.116270  89.045  < 2e-16 ***
## log.land.area -0.035003   0.005420  -6.459 2.89e-10 ***
## pop.18_34     -0.015478   0.001307 -11.840  < 2e-16 ***
## pop.65_plus   -0.002648   0.001392  -1.903 0.057700 .
## log.doctors    0.047877   0.013146   3.642 0.000304 ***
## log.hosp.beds  0.008624   0.013050   0.661 0.509042
## log.crimes     0.005828   0.008949   0.651 0.515255
## pct.hs.grad   -0.005551   0.001172  -4.737 2.96e-06 ***
## pct.bach.deg   0.016348   0.001057  15.468  < 2e-16 ***
## pct.below.pov -0.024063   0.001413 -17.033  < 2e-16 ***
## pct.unemp      0.008846   0.002380   3.717 0.000229 ***
## regionNE      -0.003192   0.012683  -0.252 0.801414
## regionS       -0.031864   0.012271  -2.597 0.009740 **
## regionW       -0.014019   0.015422  -0.909 0.363849
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08124 on 426 degrees of freedom
## Multiple R-squared:  0.8502, Adjusted R-squared:  0.8456
## F-statistic: 185.9 on 13 and 426 DF,  p-value: < 2.2e-16
```

```
vif(cdi.1)
```

```
##                    GVIF Df GVIF^(1/(2*Df))
## log.land.area  1.484666  1        1.218469
## pop.18_34      1.997041  1        1.413167
## pop.65_plus    2.053608  1        1.433041
## log.doctors   15.046783  1        3.879018
## log.hosp.beds 11.403602  1        3.376922
## log.crimes     6.238869  1        2.497773
## pct.hs.grad    4.495620  1        2.120288
## pct.bach.deg   4.353907  1        2.086602
## pct.below.pov  2.878962  1        1.696750
## pct.unemp      2.059659  1        1.435151
## region         3.580885  3        1.236892
```

```
par(mfrow=c(2,2))
plot(cdi.1)
```



The residual diagnostic plots do not look terrible, and, overall, this is not an awful model to start with. Some coefficients make sense, and those that don't (e.g. per-capita income goes down as percent high school grads goes up, or goes up as percent unemployment goes up) could be due to some remaining collinearity. Also, all of the coefficients seem quite small, although the R^2 is still a decent 85%. All these are reasons to try some additional variable selection to see if we can improve the model (i.e., keep a high R^2 but have coefficients that are larger and still significant, in a model that satisfies the assumptions of linear regression and is interpretable for the social scientist).

Variable Selection: Other Methods

Since there are not many predictors I can try all-subsets variable selection, and I will also try a couple others:

- *All subsets, with AIC and BIC*
- *Stepwise, with AIC and BIC*
- *lasso, using crossvalidation to choose λ, as in problem #1.*

All subsets: *I will do all-subsets variable selection without the categorical* `region` *variable, since* `regsubsets()` *doesn't treat categorical variables correctly, and then try to add* `region` *back in to see if it helps.*

```
pmax <- dim(cdi.trans)[2] -  2 ## for log.per.cap.income and region
all.subsets <- regsubsets(log.per.cap.income ~ . - region, data=cdi.trans, nvmax=pmax)
```

```
tmp <- summary(all.subsets)
p <- 1:dim(tmp$which)[1]
n <- dim(cdi.trans)[1]

attach(tmp)

results <- data.frame(which, BIC=bic, AIC=n*log(rss) + 2*(p+2))

detach()

results
```

```
##    X.Intercept. log.land.area pop.18_34 pop.65_plus log.doctors log.hosp.beds
## 1          TRUE         FALSE     FALSE       FALSE       FALSE         FALSE
## 2          TRUE         FALSE     FALSE       FALSE        TRUE         FALSE
## 3          TRUE         FALSE     FALSE       FALSE        TRUE         FALSE
## 4          TRUE         FALSE      TRUE       FALSE        TRUE         FALSE
## 5          TRUE          TRUE      TRUE       FALSE        TRUE         FALSE
## 6          TRUE          TRUE      TRUE       FALSE        TRUE         FALSE
## 7          TRUE          TRUE      TRUE       FALSE        TRUE         FALSE
## 8          TRUE          TRUE      TRUE        TRUE        TRUE         FALSE
## 9          TRUE          TRUE      TRUE        TRUE        TRUE          TRUE
## 10         TRUE          TRUE      TRUE        TRUE        TRUE          TRUE
##    log.crimes pct.hs.grad pct.bach.deg pct.below.pov pct.unemp       BIC
## 1       FALSE       FALSE         TRUE         FALSE     FALSE -257.5260
## 2       FALSE       FALSE        FALSE          TRUE     FALSE -502.4302
## 3       FALSE       FALSE         TRUE          TRUE     FALSE -572.5538
## 4       FALSE       FALSE         TRUE          TRUE     FALSE -682.8532
## 5       FALSE       FALSE         TRUE          TRUE     FALSE -732.1894
## 6       FALSE       FALSE         TRUE          TRUE      TRUE -761.5908
## 7       FALSE        TRUE         TRUE          TRUE      TRUE -772.0715
## 8       FALSE        TRUE         TRUE          TRUE      TRUE -770.5990
## 9       FALSE        TRUE         TRUE          TRUE      TRUE -766.2235
## 10       TRUE        TRUE         TRUE          TRUE      TRUE -760.4131
##          AIC
## 1  1026.3070
## 2   777.3161
## 3   703.1057
## 4   588.7195
## 5   535.2965
## 6   501.8084
## 7   487.2409
## 8   484.6267
## 9   484.9153
## 10  486.6390
```

```
minimize <- function(res,column) {
  obj <- res[,column]
  j <- (1:length(obj))[obj==min(obj)]
  fla <- names(res)[2:(pmax+1)][unlist(res[j,2:(pmax+1)])]
  fla <- paste("log.per.cap.income ~", paste(fla, collapse=" + "))
  return(fla)
}
```

```r
best.bic.fla <- minimize(results,"BIC")


best.aic.fla <- minimize(results,"AIC")
```

```r
summary(best.bic.model <- lm(best.bic.fla,data=cdi.trans))
```

```
##
## Call:
## lm(formula = best.bic.fla, data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34147 -0.04886 -0.00538  0.04818  0.26969
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.2224950  0.0931210 109.776  < 2e-16 ***
## log.land.area -0.0356741  0.0047767  -7.468 4.53e-13 ***
## pop.18_34     -0.0139002  0.0011113 -12.508  < 2e-16 ***
## log.doctors    0.0606769  0.0040183  15.100  < 2e-16 ***
## pct.hs.grad   -0.0044064  0.0010823  -4.071 5.56e-05 ***
## pct.bach.deg   0.0153853  0.0009246  16.641  < 2e-16 ***
## pct.below.pov -0.0242784  0.0012583 -19.294  < 2e-16 ***
## pct.unemp      0.0106037  0.0021771   4.871 1.56e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.082 on 432 degrees of freedom
## Multiple R-squared:  0.8452, Adjusted R-squared:  0.8427
## F-statistic: 336.9 on 7 and 432 DF,  p-value: < 2.2e-16
```

```r
summary(best.aic.model <- lm(best.aic.fla,data=cdi.trans))
```

```
##
## Call:
## lm(formula = best.aic.fla, data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35756 -0.04551 -0.00543  0.04844  0.27399
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.3159666  0.1025858 100.559  < 2e-16 ***
## log.land.area -0.0364935  0.0047728  -7.646 1.36e-13 ***
## pop.18_34     -0.0153488  0.0012988 -11.818  < 2e-16 ***
## pop.65_plus   -0.0027664  0.0012978  -2.132   0.0336 *
## log.doctors    0.0626053  0.0041029  15.259  < 2e-16 ***
## pct.hs.grad   -0.0046579  0.0010843  -4.296 2.15e-05 ***
## pct.bach.deg   0.0152149  0.0009242  16.462  < 2e-16 ***
## pct.below.pov -0.0246144  0.0012631 -19.488  < 2e-16 ***
## pct.unemp      0.0107688  0.0021696   4.963 9.99e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.08167 on 431 degrees of freedom
## Multiple R-squared:  0.8468, Adjusted R-squared:  0.8439
## F-statistic: 297.7 on 8 and 431 DF,  p-value: < 2.2e-16
```

We can see that the best BIC model and the best AIC model differ only in that the best AIC model adds `pop.65_plus` *to the BIC model.*

Let's see what happens when we add region back in:

```
best.bic.plus.region <- update(best.bic.model, . ~ . + region)

best.aic.plus.region <- update(best.aic.model, . ~ . + region)

cbind(AIC(best.bic.model,
          best.bic.plus.region,
          best.aic.model,
          best.aic.plus.region),
      BIC(best.bic.model,
          best.bic.plus.region,
          best.aic.model,
          best.aic.plus.region)
)
```

```
##                      df       AIC df       BIC
## best.bic.model        9 -942.2740  9 -905.4931
## best.bic.plus.region 12 -945.3762 12 -896.3350
## best.aic.model       10 -944.8883 10 -904.0206
## best.aic.plus.region 13 -947.5395 13 -894.4114
```

So, we see that the minimum BIC model does not include region, but the minimum AIC model does.

Stepwise: *Fortunately, the* `stepAIC` *procedure does know how to deal with categorical variables like* `region`*, so no special treatment of region is required:*

```
summary(best.bic.stepwise <- stepAIC(cdi.1,k=log(n),trace=0))
```

```
##
## Call:
## lm(formula = log.per.cap.income ~ log.land.area + pop.18_34 +
##     log.doctors + pct.hs.grad + pct.bach.deg + pct.below.pov +
##     pct.unemp, data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34147 -0.04886 -0.00538  0.04818  0.26969
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.2224950  0.0931210 109.776  < 2e-16 ***
## log.land.area -0.0356741  0.0047767  -7.468 4.53e-13 ***
## pop.18_34     -0.0139002  0.0011113 -12.508  < 2e-16 ***
## log.doctors    0.0606769  0.0040183  15.100  < 2e-16 ***
## pct.hs.grad   -0.0044064  0.0010823  -4.071 5.56e-05 ***
## pct.bach.deg   0.0153853  0.0009246  16.641  < 2e-16 ***
## pct.below.pov -0.0242784  0.0012583 -19.294  < 2e-16 ***
## pct.unemp      0.0106037  0.0021771   4.871 1.56e-06 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.082 on 432 degrees of freedom
## Multiple R-squared:  0.8452, Adjusted R-squared:  0.8427
## F-statistic: 336.9 on 7 and 432 DF,  p-value: < 2.2e-16
```

```
summary(best.aic.stepwise <- stepAIC(cdi.1,k=2,trace=0))
```

```
##
## Call:
## lm(formula = log.per.cap.income ~ log.land.area + pop.18_34 +
##     pop.65_plus + log.doctors + pct.hs.grad + pct.bach.deg +
##     pct.below.pov + pct.unemp + region, data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34849 -0.04695 -0.00502  0.04524  0.28624
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.3851173  0.1105475  93.943  < 2e-16 ***
## log.land.area -0.0346133  0.0053943  -6.417 3.70e-10 ***
## pop.18_34     -0.0153941  0.0013021 -11.822  < 2e-16 ***
## pop.65_plus   -0.0026499  0.0013137  -2.017  0.04430 *
## log.doctors    0.0608452  0.0041649  14.609  < 2e-16 ***
## pct.hs.grad   -0.0055059  0.0011696  -4.707 3.39e-06 ***
## pct.bach.deg   0.0159212  0.0009688  16.434  < 2e-16 ***
## pct.below.pov -0.0238604  0.0013529 -17.637  < 2e-16 ***
## pct.unemp      0.0090479  0.0023017   3.931 9.86e-05 ***
## regionNE      -0.0061091  0.0123398  -0.495  0.62080
## regionS       -0.0311704  0.0114050  -2.733  0.00654 **
## regionW       -0.0162724  0.0140361  -1.159  0.24697
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08115 on 428 degrees of freedom
## Multiple R-squared:  0.8498, Adjusted R-squared:  0.8459
## F-statistic: 220.1 on 11 and 428 DF,  p-value: < 2.2e-16
```

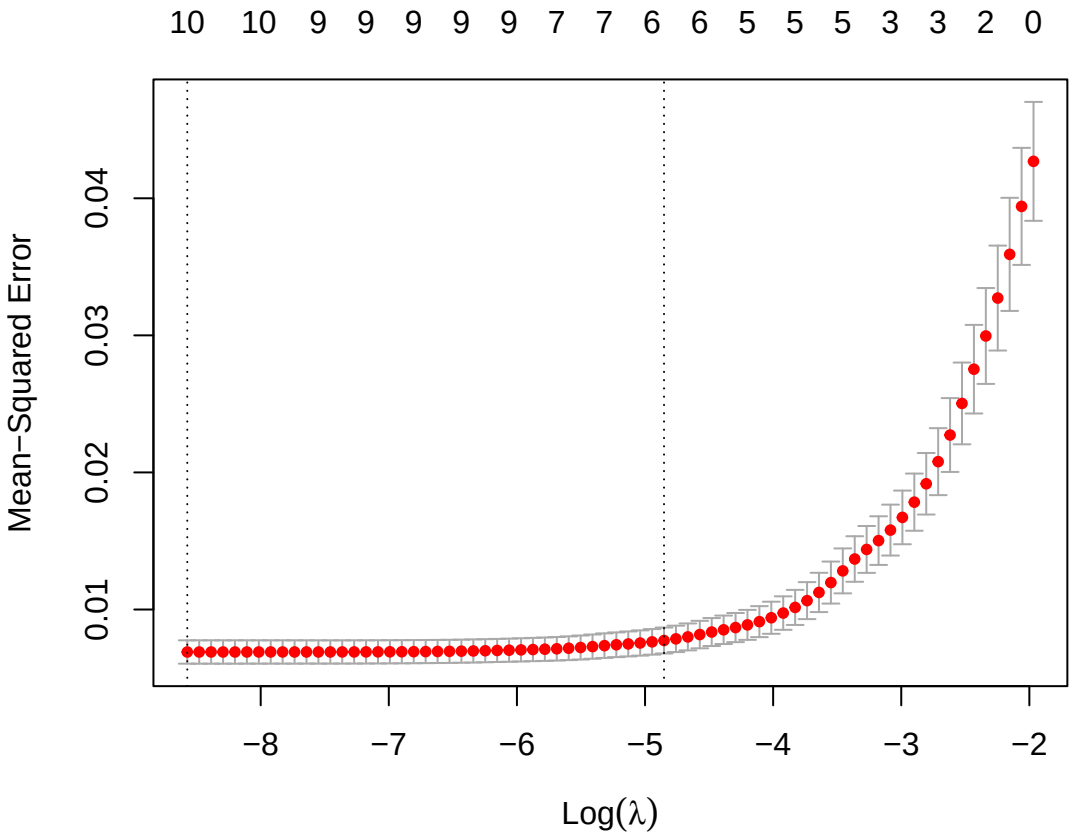
The stepwise AIC model adds the variables `pop.65_plus` *and* `region` *to the stepwise BIC model. Otherwise they are the same.*

Lasso: *Finally, we try the lasso, with λ selected by cross-validation. Because* `glmnet` *cannot deal with categorical variables, we take* `region` *out, and then check to see if it should be added in again, just as we did with the all-subsets procedure.*

```
y <- cdi.trans$log.per.cap.income
X <- as.matrix(cdi.trans[,!(names(cdi.trans) %in% c("log.per.cap.income","region"))])

result <- cv.glmnet(X,y)

plot(result)
```




```
c(lambda.1se=result$lambda.1se,lambda.min=result$lambda.min)
```

```
##   lambda.1se   lambda.min
## 0.0078151963 0.0001891378
```

```
log(c(log.lambda.1se=result$lambda.1se,log.lambda.min=result$lambda.min))
```

```
## log.lambda.1se log.lambda.min
##      -4.851685      -8.573035
```

```
cbind(coef(result,s=result$lambda.1se),coef(result,s=result$lambda.min))
```

```
## 11 x 2 sparse Matrix of class "dgCMatrix"
##                           s1           s1
## (Intercept)    9.873238119 10.288836619
## log.land.area -0.030396717 -0.035715268
## pop.18_34     -0.011344178 -0.015364277
## pop.65_plus    .           -0.003027088
## log.doctors    0.058487164  0.051964486
## log.hosp.beds  .            0.014479819
## log.crimes     .           -0.002419004
## pct.hs.grad    .           -0.004537077
## pct.bach.deg   0.011283588  0.015567203
## pct.below.pov -0.019629650 -0.024753098
## pct.unemp      0.004460639  0.010950352
```

We can see that the predictors for the lasso model that minimizes 10-fold cross-validation prediction error include all the available variables (possibly indicating overfit) while the predictors for the lasso model with prediction error one SE above the minimum includes only the variables `log.land.area`, `pop.18_34`, `log.doctors`, `pct.back.deg`, `pct.below.pov`, *and* `pct.unemp`.

We need to see if adding `region` *back in improves the model or not. For this, we will refit the two lasso models without the lasso penalty, and compare (using AIC and BIC) with the same models but with* `region` *added in.*

```
result.1se <- as.matrix(coef(result,s=result$lambda.1se))
names.1se <- dimnames(result.1se)[[1]][result.1se!=0][-1]

result.min <- as.matrix(coef(result,s=result$lambda.min))
names.min <- dimnames(result.min)[[1]][result.min!=0][-1]

lasso.1se.fla <- paste("log.per.cap.income ~",paste(names.1se,collapse=" + "))
lasso.min.fla <- paste("log.per.cap.income ~",paste(names.min,collapse=" + "))

lasso.1se.model <- lm(lasso.1se.fla,data=cdi.trans)
lasso.1se.plus.region <- update(lasso.1se.model, . ~ . + region)

lasso.min.model <- lm(lasso.min.fla,data=cdi.trans)
lasso.min.plus.region <- update(lasso.min.model, . ~ . + region)

cbind(AIC(lasso.1se.model,
          lasso.1se.plus.region,
          lasso.min.model,
          lasso.min.plus.region),
      BIC(lasso.1se.model,
          lasso.1se.plus.region,
          lasso.min.model,
          lasso.min.plus.region)
)
```

```
##                        df       AIC df       BIC
## lasso.1se.model         8 -927.7066  8 -895.0124
## lasso.1se.plus.region  11 -925.9755 11 -881.0209
## lasso.min.model        12 -942.8760 12 -893.8347
## lasso.min.plus.region  15 -944.7011 15 -883.3995
```

Interstingly, AIC likes the most complex model `lasso.min.plus.region`, *whereas BIC likes the simplest model* `lasso.1se.model`.

Let's summarize the models we have so far in a table:

```
var.names <- c("(Intercept)",names(cdi.trans)[names(cdi.trans)!="log.per.cap.income"])
var.names <- c(var.names[!(var.names=="region")], "regionNE" ,"regionS", "regionW", "regionNC")

tab <- matrix(NA,ncol=6,nrow=length(var.names))
dimnames(tab) <- list(var.names,c("AIC.all",
                                  "AIC.step",
                                  "lasso.AIC",
                                  "BIC.all",
                                  "BIC.step",
                                  "lasso.BIC"))

models <- list(best.aic.plus.region,
               best.aic.stepwise,
```

```
                lasso.min.plus.region,
                best.bic.model,
                best.bic.stepwise,
                lasso.1se.model
                )

for (m in 1:6) {
  coefs <- summary(models[[m]])$coef
  vars <- dimnames(coefs)[[1]]
  vals <- coefs[,1]
  vals <- ifelse(vals==0,NA,vals)
  tab[vars,m] <- vals
}
tab <- rbind(tab,AIC=sapply(models,AIC),BIC=sapply(models,BIC))

round(tab,4)
```

```
##                 AIC.all  AIC.step lasso.AIC   BIC.all  BIC.step lasso.BIC
## (Intercept)     10.3851   10.3851   10.3532   10.2225   10.2225    9.9034
## log.land.area   -0.0346   -0.0346   -0.0350   -0.0357   -0.0357   -0.0402
## pop.18_34       -0.0154   -0.0154   -0.0155   -0.0139   -0.0139   -0.0141
## pop.65_plus     -0.0026   -0.0026   -0.0026        NA        NA        NA
## log.doctors      0.0608    0.0608    0.0479    0.0607    0.0607    0.0629
## log.hosp.beds        NA        NA    0.0086        NA        NA        NA
## log.crimes           NA        NA    0.0058        NA        NA        NA
## pct.hs.grad     -0.0055   -0.0055   -0.0056   -0.0044   -0.0044        NA
## pct.bach.deg     0.0159    0.0159    0.0163    0.0154    0.0154    0.0134
## pct.below.pov   -0.0239   -0.0239   -0.0241   -0.0243   -0.0243   -0.0214
## pct.unemp        0.0090    0.0090    0.0088    0.0106    0.0106    0.0129
## regionNE        -0.0061   -0.0061   -0.0032        NA        NA        NA
## regionS         -0.0312   -0.0312   -0.0319        NA        NA        NA
## regionW         -0.0163   -0.0163   -0.0140        NA        NA        NA
## regionNC             NA        NA        NA        NA        NA        NA
## AIC           -947.5395 -947.5395 -944.7011 -942.2740 -942.2740 -927.7066
## BIC           -894.4114 -894.4114 -883.3995 -905.4931 -905.4931 -895.0124
```

Note that the stepwise and all-subsets approaches led to the same model, using AIC as the measure; but the lasso gave the largest model (identical to cdi.1, and probably an overfitting model). Similarly the all-subsets and stepwise approaches used with BIC led to just a single model, and the lasso gave a slightly smaller one (the results here may vary a bit, depending on the random selection of folds in the `cv.glmnet` *algorithm).*

Since AIC is minimized for the AIC.step model (best.aic.stepwise) and BIC is minimized for the BIC.step model (best.bic.stepwise), I will just consider these two models going forward.

I next considered two-way interactions, with code like this

```
summary(best.aic.stepwise.2 <- update(best.aic.stepwise, . ~ .^2))
summary(best.bic.stepwise.2 <- update(best.bic.stepwise, . ~ .^2))
```

(output not shown, to save space. . .)

While both updated models produced some significant interactions, they were largely uninterpretable (e.g. `pop.65_plus:pct.bach.deg`*), had very small coefficients, and sapped the main effects of statistical significance (this often happens, because interactions are collinear with their main effects). Because of my lack of success with two-way interactions, I did not try three-way or higher interactions.*

Diagnostics

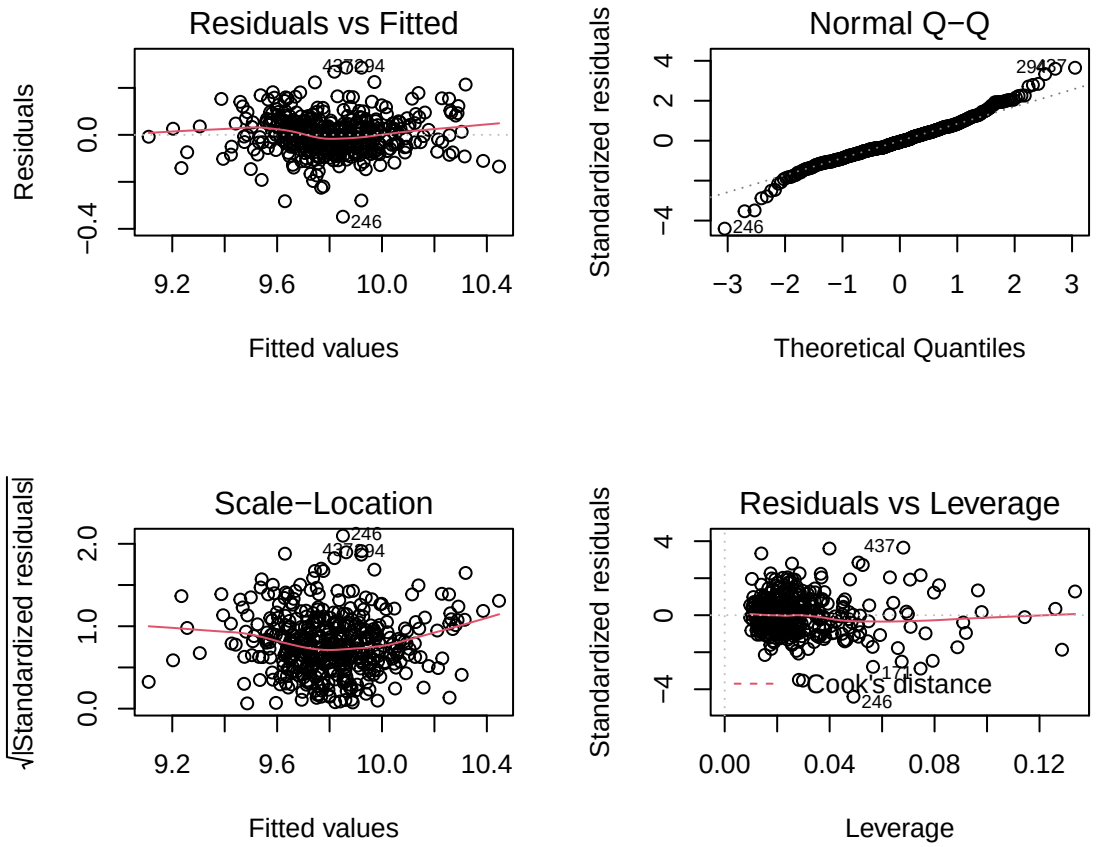
Let's look at diagnostic plots for the two models.

First, best.aic.stepwise:

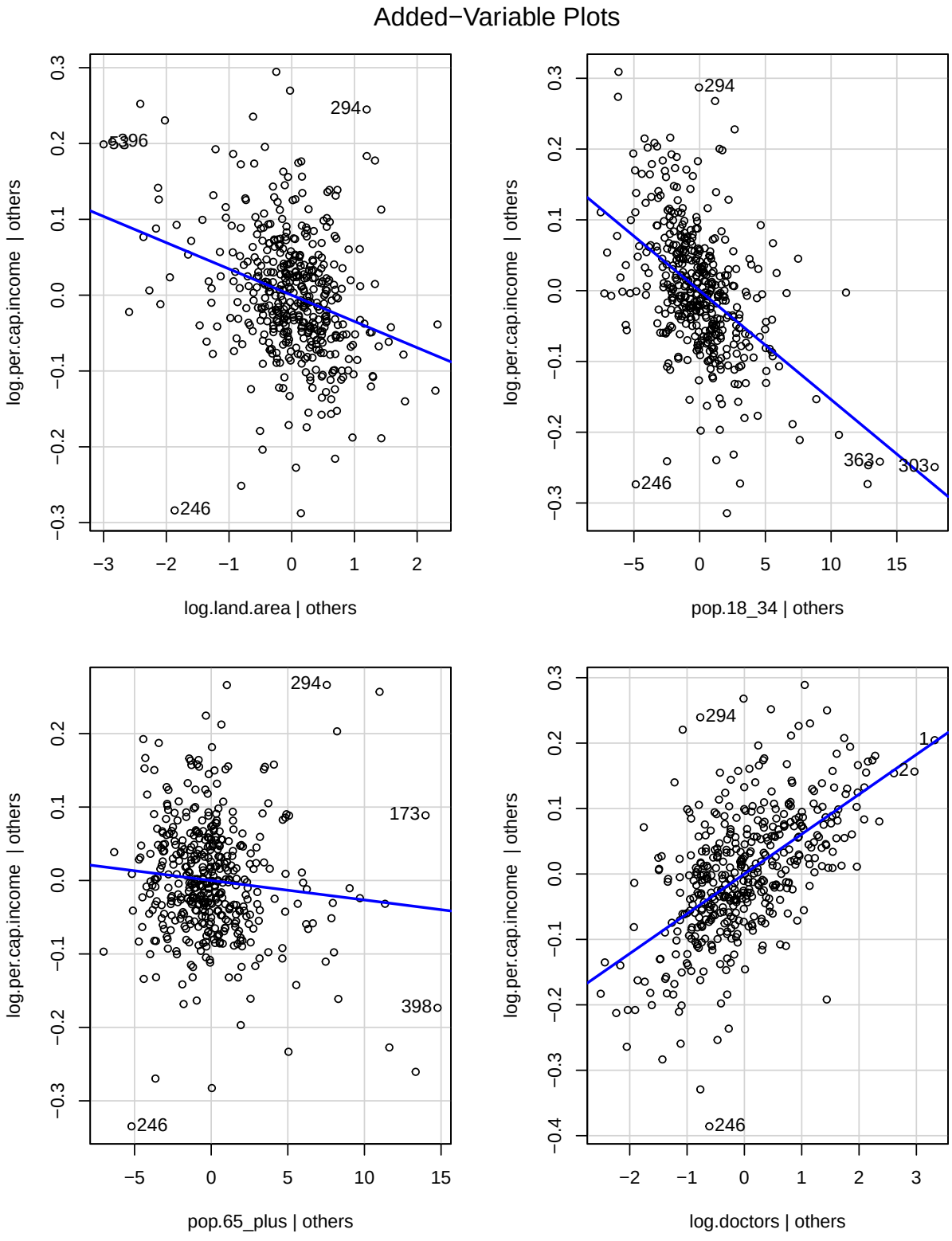
```
vif(best.aic.stepwise)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## log.land.area 1.473876  1        1.214033
## pop.18_34     1.985228  1        1.408981
## pop.65_plus   1.833837  1        1.354192
## log.doctors   1.513383  1        1.230196
## pct.hs.grad   4.487526  1        2.118378
## pct.bach.deg  3.665534  1        1.914558
## pct.below.pov 2.645670  1        1.626552
## pct.unemp     1.930186  1        1.389311
## region        2.364456  3        1.154220
```

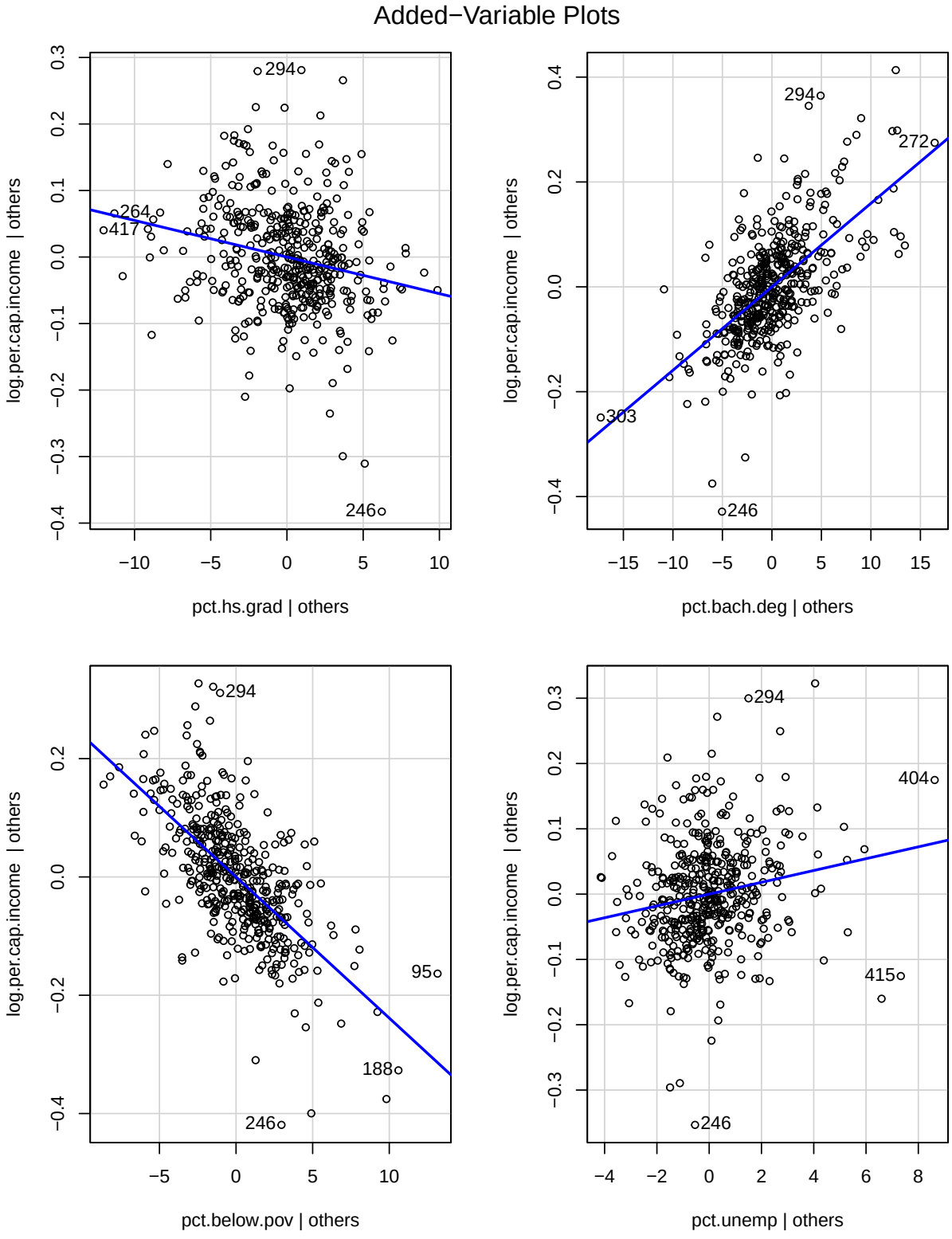
```
par(mfrow=c(2,2))
plot(best.aic.stepwise)
```



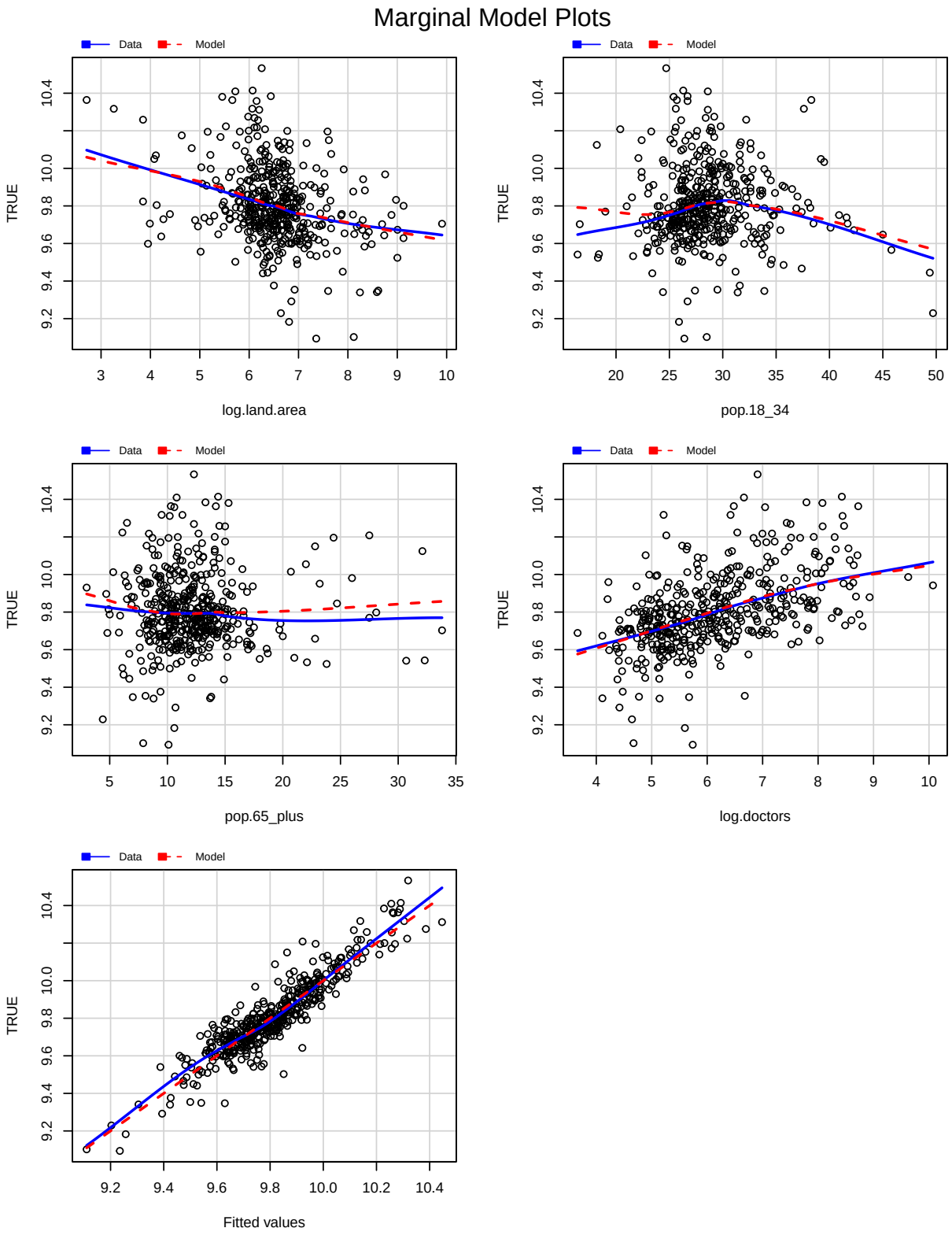

```
## I divided the added-variable and marginal model plots up into several pages each
## to make them more legible...
avPlots(best.aic.stepwise, ~ log.land.area + pop.18_34 + pop.65_plus + log.doctors)
```



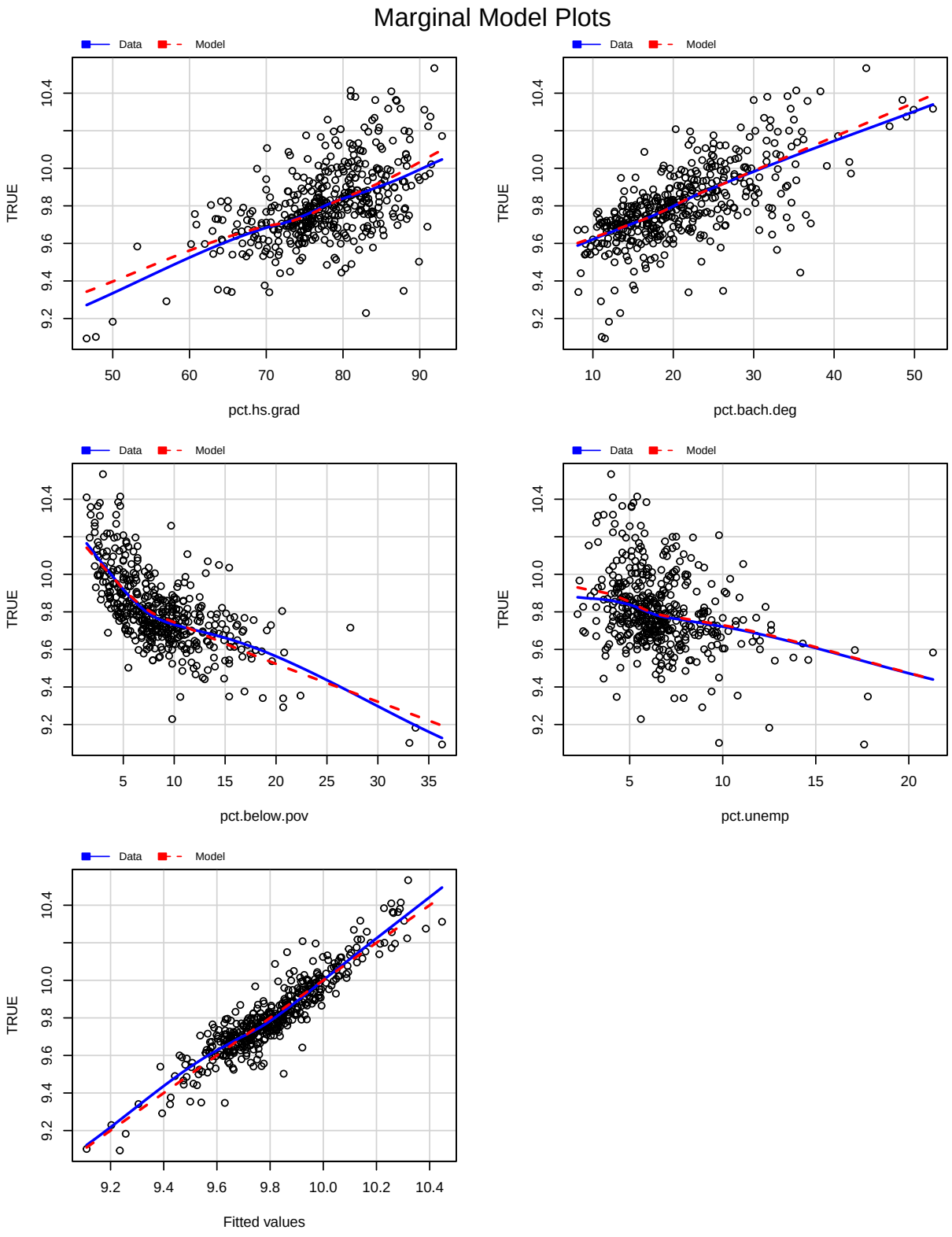

```
avPlots(best.aic.stepwise, ~ pct.hs.grad + pct.bach.deg + pct.below.pov + pct.unemp)
```




```
mmps(best.aic.stepwise, ~ log.land.area + pop.18_34 + pop.65_plus + log.doctors)
```




```
mmps(best.aic.stepwise, ~ pct.hs.grad + pct.bach.deg + pct.below.pov + pct.unemp)
```

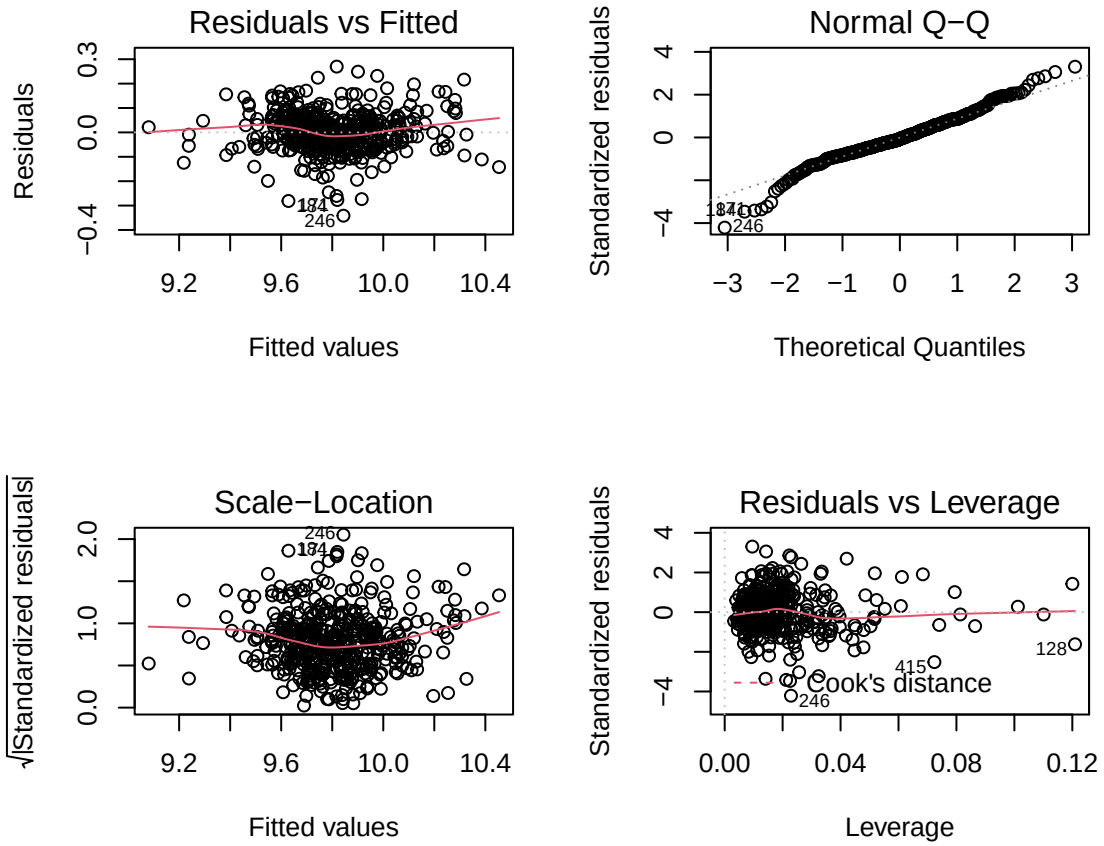


Next, best.bic.stepwise:

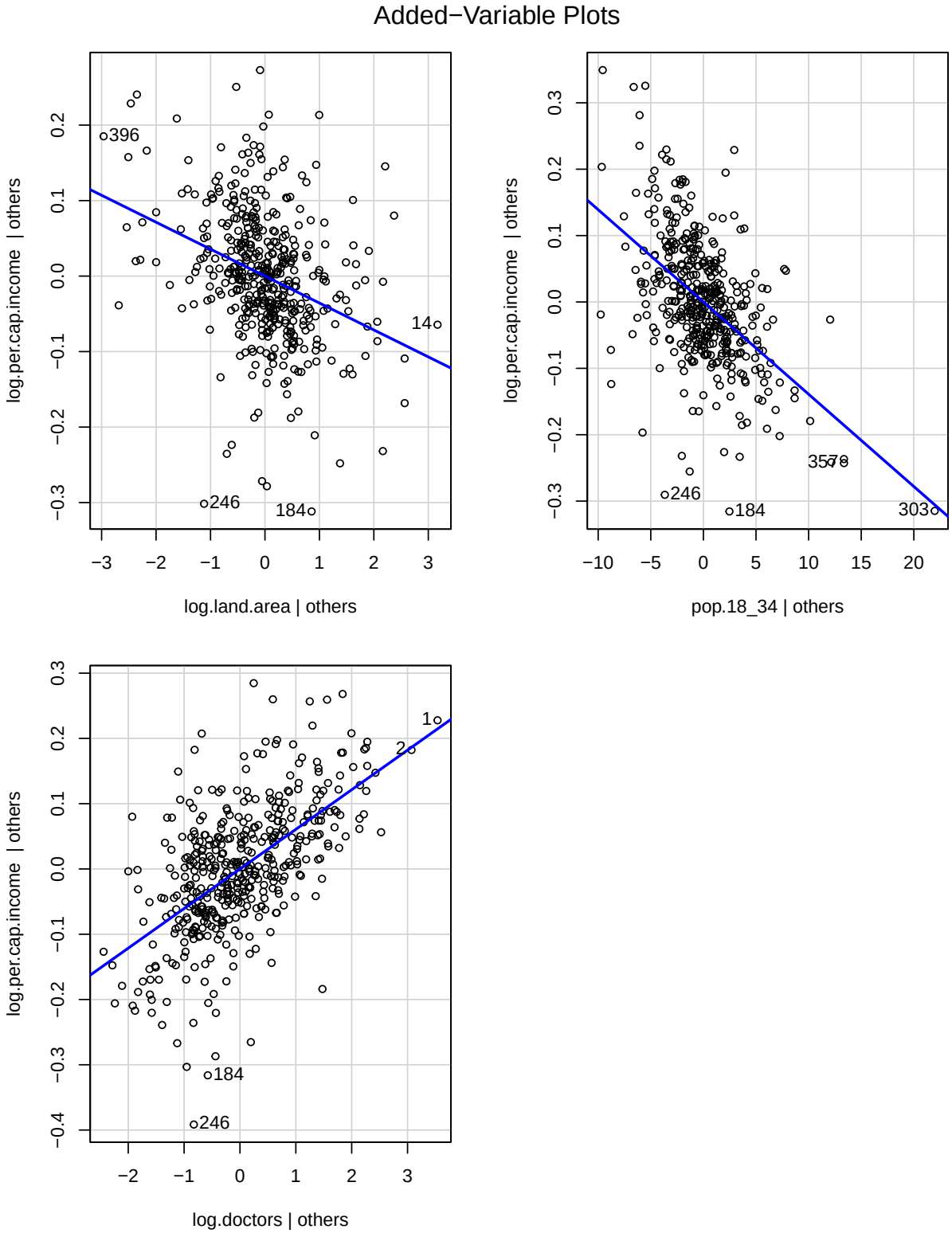
```
vif(best.bic.stepwise)
```

```
## log.land.area     pop.18_34   log.doctors   pct.hs.grad  pct.bach.deg
##      1.131867      1.416145      1.379671      3.763103      3.269565
## pct.below.pov     pct.unemp
##      2.241555      1.691280
```

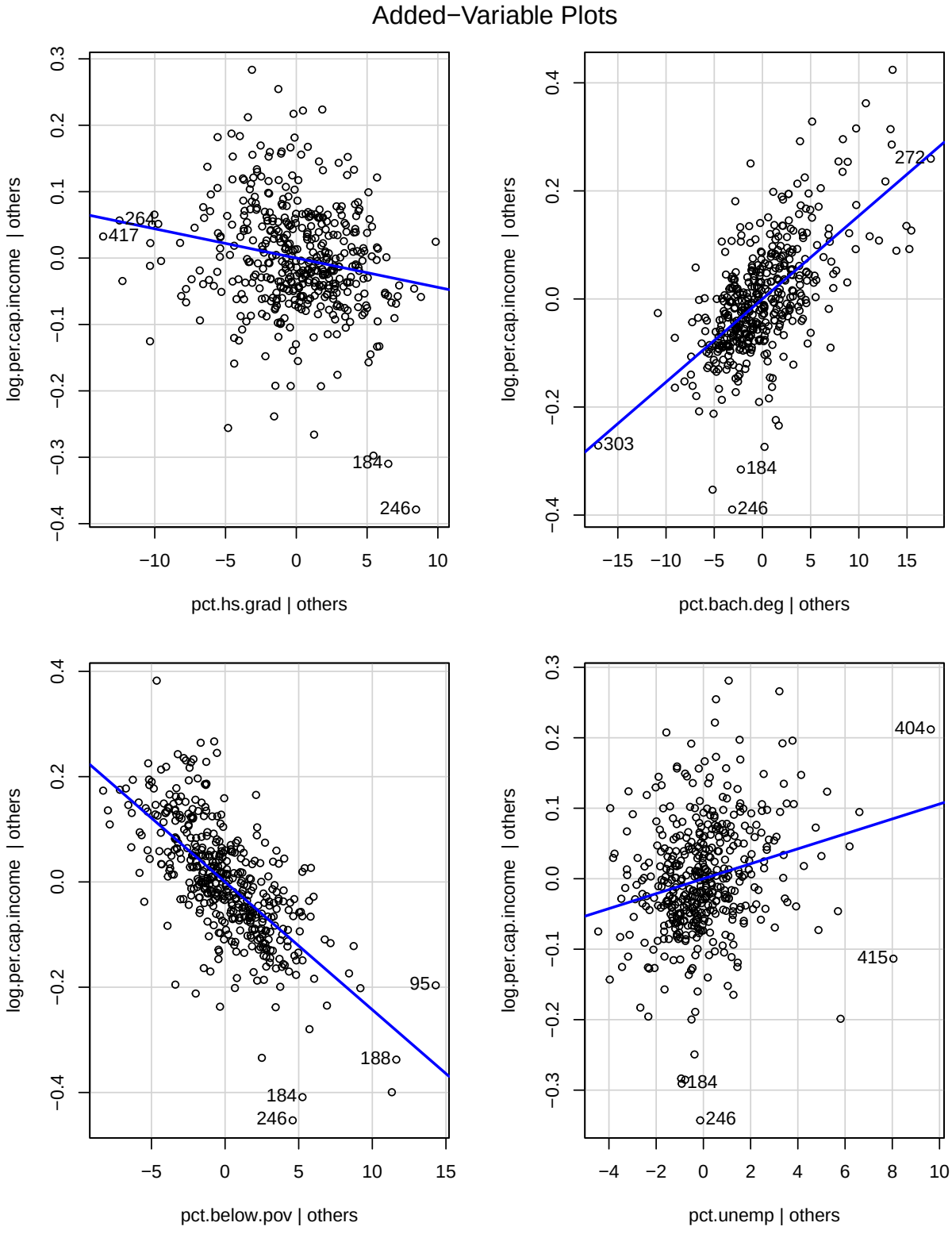
```
par(mfrow=c(2,2))
plot(best.bic.stepwise)
```



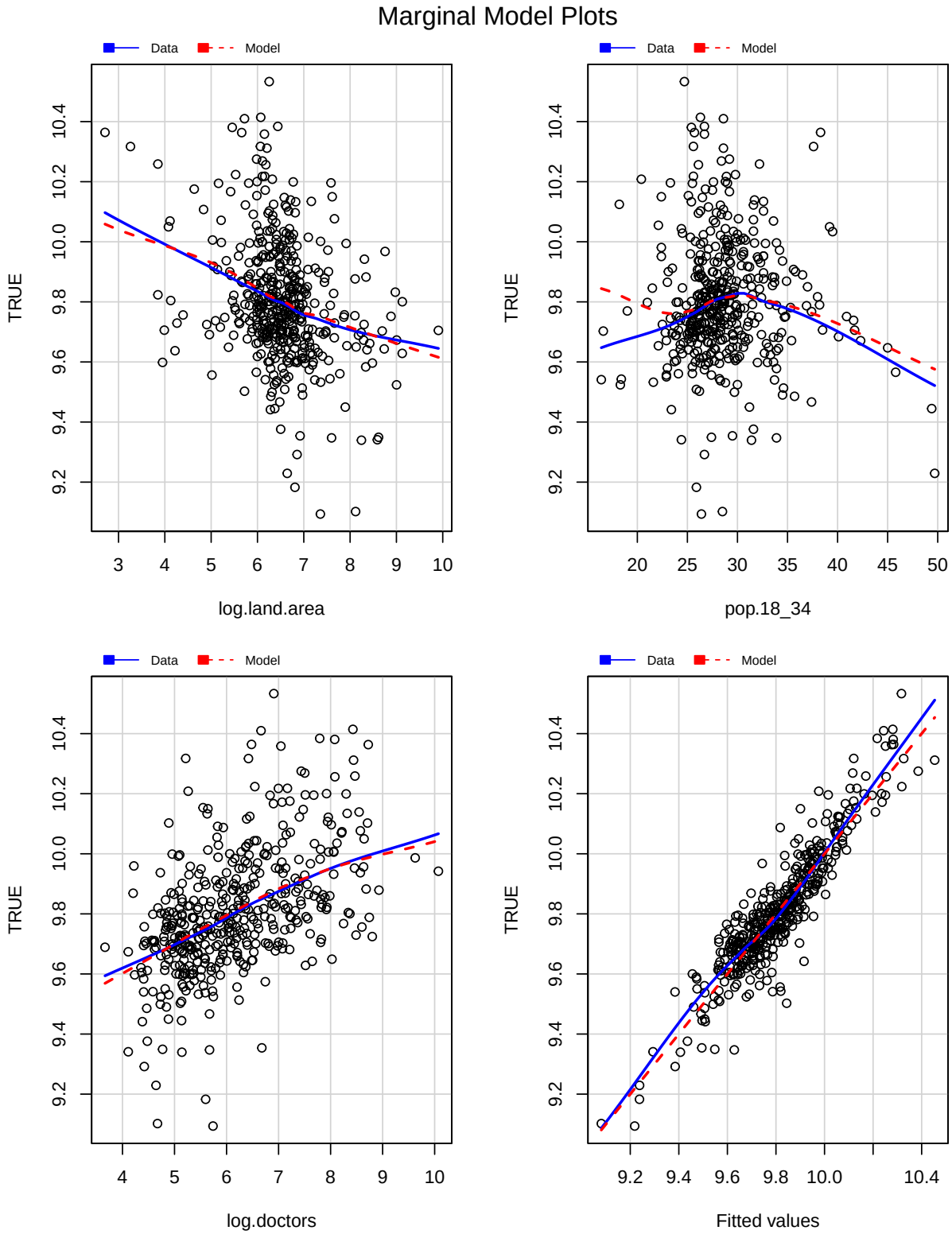

```
## I divided the added-variable and marginal model plots up into several pages each
## to make them more legible...
avPlots(best.bic.stepwise,~ log.land.area + pop.18_34 + log.doctors)
```



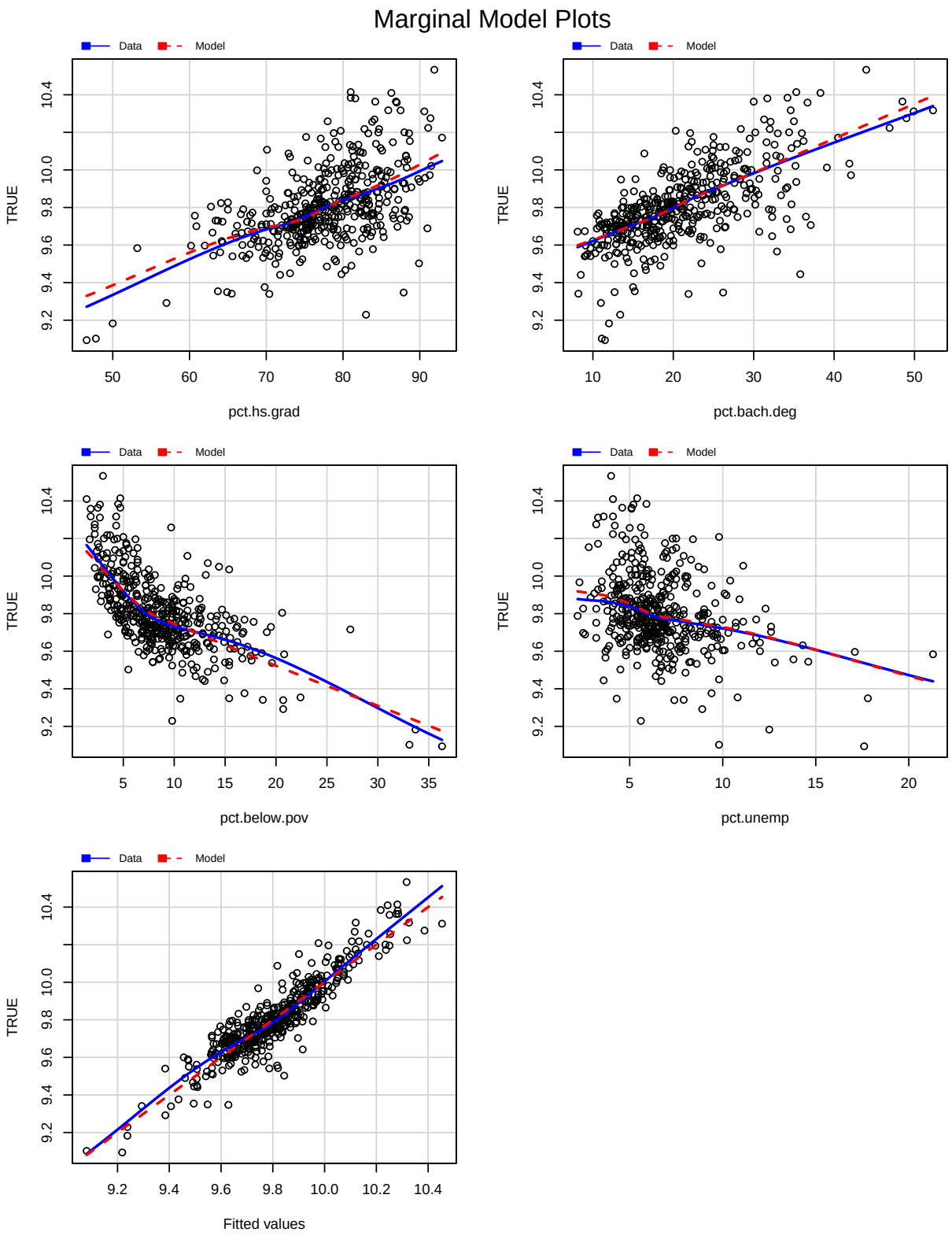

```
avPlots(best.bic.stepwise,~ pct.hs.grad + pct.bach.deg + pct.below.pov + pct.unemp)
```




```
mmps(best.bic.stepwise,~ log.land.area + pop.18_34 + log.doctors)
```




```
mmps(best.bic.stepwise,~ pct.hs.grad + pct.bach.deg + pct.below.pov + pct.unemp)
```



I don't see much in these plots to differentiate the two models. The casewise diagnostic plots asre not perfect, but they only show fairly mild violations of the modeing assumptions; the VIF'a are a little higher for the best.AIC.stepwise model, but that is not surprising since it has more predictors. The avPlots and marginal model plots all look pretty good; there are no suggestions that further transformations are needed for either

model.

For discussion with the social scientist I will take the bigger model, `best.AIC.stepwise`*, because it has more variables to have a conversation about.*

(You may have chosen a smaller model here. That is fine as long as you gave a thoughtful reason for your choice.)

Problem 2(d).

Provide a careful and easy-to-follow interpretation of your final model for a client or collaborator who is more interested in social, economic and health factors than in mathematics and statistics. This should be long enough that you hit all the important points, but not so long that your collaborator gets bored and stops reading.

Here is a summary of the model.

```
vif(best.aic.stepwise)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## log.land.area 1.473876  1        1.214033
## pop.18_34     1.985228  1        1.408981
## pop.65_plus   1.833837  1        1.354192
## log.doctors   1.513383  1        1.230196
## pct.hs.grad   4.487526  1        2.118378
## pct.bach.deg  3.665534  1        1.914558
## pct.below.pov 2.645670  1        1.626552
## pct.unemp     1.930186  1        1.389311
## region        2.364456  3        1.154220
```

```
summary(best.aic.stepwise)
```

```
##
## Call:
## lm(formula = log.per.cap.income ~ log.land.area + pop.18_34 +
##     pop.65_plus + log.doctors + pct.hs.grad + pct.bach.deg +
##     pct.below.pov + pct.unemp + region, data = cdi.trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34849 -0.04695 -0.00502  0.04524  0.28624
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.3851173  0.1105475  93.943  < 2e-16 ***
## log.land.area -0.0346133  0.0053943  -6.417 3.70e-10 ***
## pop.18_34     -0.0153941  0.0013021 -11.822  < 2e-16 ***
## pop.65_plus   -0.0026499  0.0013137  -2.017  0.04430 *
## log.doctors    0.0608452  0.0041649  14.609  < 2e-16 ***
## pct.hs.grad   -0.0055059  0.0011696  -4.707 3.39e-06 ***
## pct.bach.deg   0.0159212  0.0009688  16.434  < 2e-16 ***
## pct.below.pov -0.0238604  0.0013529 -17.637  < 2e-16 ***
## pct.unemp      0.0090479  0.0023017   3.931 9.86e-05 ***
## regionNE      -0.0061091  0.0123398  -0.495  0.62080
## regionS       -0.0311704  0.0114050  -2.733  0.00654 **
## regionW       -0.0162724  0.0140361  -1.159  0.24697
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08115 on 428 degrees of freedom
## Multiple R-squared:  0.8498, Adjusted R-squared:  0.8459
## F-statistic: 220.1 on 11 and 428 DF,  p-value: < 2.2e-16
```

Since the model has a logarithm for y and for some of the x's, the interpretations in the handout "log xform and percent interpretation.pdf" from week03 will be important. In particular

- *When* $\log(x)$ *and* $\log(y)$ *are both in the model, a 1% change in x is associated with an approximately* $\beta\%$ *change in* $E[y]$.
- *When x and* $log(y)$ *are in the model, a one-unit change in x is associated with an approximately* $\beta \cdot 100\%$ *change in* $E[y]$.

The model summarized above provides a pretty good account of variation in per-capita income across counties in the data set. Indeed $R^2 \approx 85\%$ *of that variation is accounted for by the model, a remarkably high* R^2 *for social science data.*

From the estimated coefficients we can infer that

- *A one percent increase in land area is assocated with a 0.03% decrease in avarage per-capita income. This seems counterintuitive until one realizes that major centers of industry and commerce are mostly located in states with smaller populations.*
- *A 1% increase in the number of doctors is associated with a 0.06% increase in average per-capita income. This makes sense, since doctors are usually on the high end of the earning scale.*
- *A one-point increase in the percentage of bachelors' degree holders is associated with a roughly 0.02×100%=2% increase on average in per capita increase. This makes sense, since degree-holdrs tend to earl more mony than on-degree hopders.*
- *A one-point increase in the percentage of persons graduating from high school is associated with a 0.5%* drop *in average per-capita income. This seems counterintuitive, but what might be going is this: since virtually all bachelors' degree holders are also hs graduates, quite often both predictors will change at the same time, and so it makes sense to look at the combined effect of* $0.0159212 - 0.0055059 = 0.0104153$*: a one-unit increase in both variables is associated with about a 1% increase in expected per capita income.*
- *A one point increase in the percent unemployed is associated with a 0.9% increase in per-capita income. This seems conterintuitive, but again this variable may be mildly collinear with pecent below poverty, which has a negative effect on log-income.*
- *The number of people aged 18–35 (and so not yet peak income earners) and the number of people 65 or odler (and hence past peak income years) are both negatively associated with per-capita income, and both of these make sense.*
- *Region matters, to the extent that per capita income in the South seems noticably lower than in the North-Central region (and there is no noticable difference in per-capita income between the other regions and the NE region).*

We can see from this list that changes in the predictors affect per capita income in the ways that one might expect. The exceptions seem to be due to mathematical co-incidences in the relationships between the variables, rather than substantial counterintuitive effects.