Modern Data Mining - HW 2

Bingying Feng
He Zhang
Dingzhu Leng

Overview / Instructions

This is homework #2 of STAT 471/571/701. It will be **due on Feb, 24, 2019 by 11:59 PM** on Canvas. You can directly edit this file to add your answers. Submit the Rmd file *and* a knitted (PDF, Word, or HTML) version, with only 1 submission allowed per HW team.

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':
##
##     expand

## Loading required package: foreach

##
## Attaching package: 'foreach'

## The following objects are masked from 'package:purrr':
##
##     accumulate, when

## Loaded glmnet 2.0-16
```

Problem 0

Review the code and concepts covered during lecture: multiple regression, model selection and penalized regression through elastic net.

Problem 1

Do ISLR, page 262, problem 8, parts (a) through (e), and write up the answer here. This question is designed to help us understanding model selection through simulations. (f) Describe as accurate as possible what C_p and BIC are estimating?

(a) Use the rnorm() function to generate a predictor X of length n = 100, as well as a noise vector of length n = 100.

```r
set.seed(1)
x <- rnorm(100)
noise <- rnorm(100)
```

(b) Generate a response vector Y of length n = 100 according to the model

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + e$$

where β_0, β_1, β_2, and β_3 are constants of your choice.

```r
b0 <- 5
b1 <- 4
b2 <- 3
b3 <- 2
y <- b0 + b1*x + b2*x^2 + b3*x^3 + noise
```

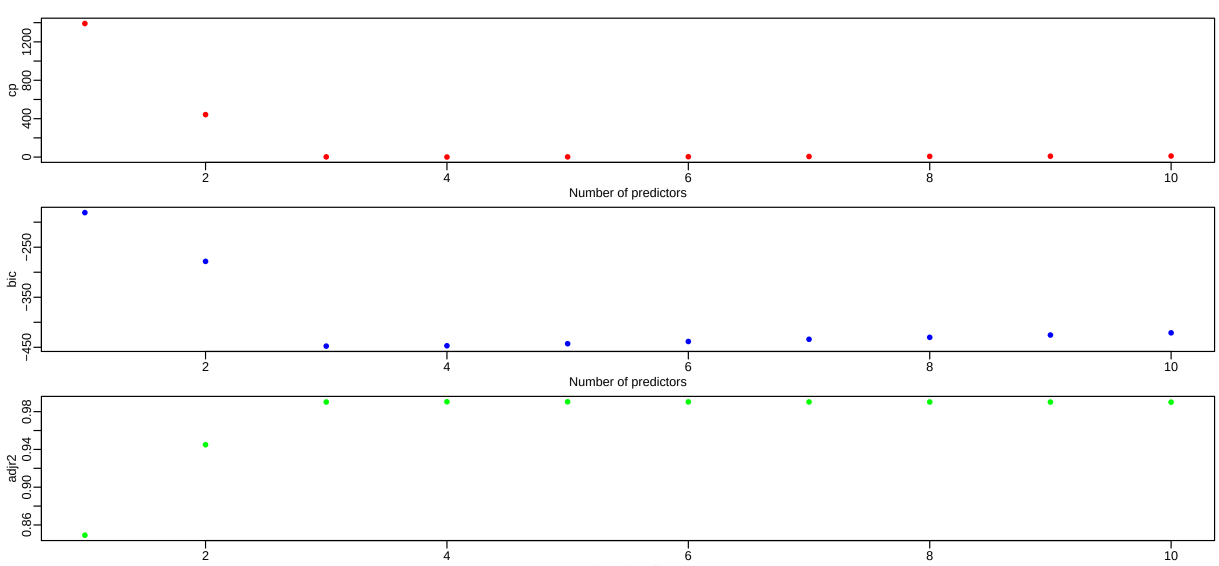
(c) Use the regsubsets() function to perform best subset selection in order to choose the best model containing the predictors X, X2,...,X10. What is the best model obtained according to Cp, BIC, and adjusted R2? Show some plots to provide evidence for your answer, and report the coefficients of the best model obtained. Note you will need to use the data.frame() function to create a single data set containing both X and Y.

```r
data <- data.frame(y = y,x = x)
```

Use library leaps to do regsubsets

```r
fit.exh <- regsubsets(y ~ x + I(x^2) + I(x^3) + I(x^4) + I(x^5) + I(x^6) + I(x^7) + I(x^8) + I(x^9) + I
f.e <- summary(fit.exh)
```

```r
par(mfrow=c(3,1), mar=c(2.5,4,0.5,1), mgp=c(1.5,0.5,0))
plot(f.e$cp, xlab="Number of predictors",
     ylab="cp", col="red", type="p", pch=16)
plot(f.e$bic, xlab="Number of predictors",
     ylab="bic", col="blue", type="p", pch=16)
plot(f.e$adjr2, xlab="Number of predictors",
     ylab="adjr2", col="green", type="p", pch=16)
```



We find the Optimal Model by Cp locate the optimal model size by Cp's is 4

```
opt.size <- which.min(f.e$cp)
opt.size
```

```
## [1] 4
```

Now we look for the optimal variables selected

```
fit.exh.var <- f.e$which # logic indicators which variables are in
fit.exh.var[opt.size,]
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)
##        TRUE        TRUE        TRUE        TRUE       FALSE        TRUE
##      I(x^6)      I(x^7)      I(x^8)      I(x^9)     I(x^10)
##       FALSE       FALSE       FALSE       FALSE       FALSE
```

Find the coefficients

```
coef(fit.exh,4)
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^5)
##  5.07200775  4.38745596  2.84575641  1.55797426  0.08072292
```

Fit the final model

```
fit.final.exh <- lm(y ~ x + I(x^2) + I(x^3) + I(x^5), data)
summary(fit.final.exh)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^3) + I(x^5), data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9204 -0.5775 -0.1686  0.5679  2.1322
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.07201    0.11795  43.001  < 2e-16 ***
## x            4.38746    0.28172  15.574  < 2e-16 ***
## I(x^2)       2.84576    0.08721  32.631  < 2e-16 ***
## I(x^3)       1.55797    0.24560   6.343 7.58e-09 ***
## I(x^5)       0.08072    0.04167   1.937   0.0557 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9491 on 95 degrees of freedom
## Multiple R-squared:  0.9909, Adjusted R-squared:  0.9905
## F-statistic:  2586 on 4 and 95 DF,  p-value: < 2.2e-16
```

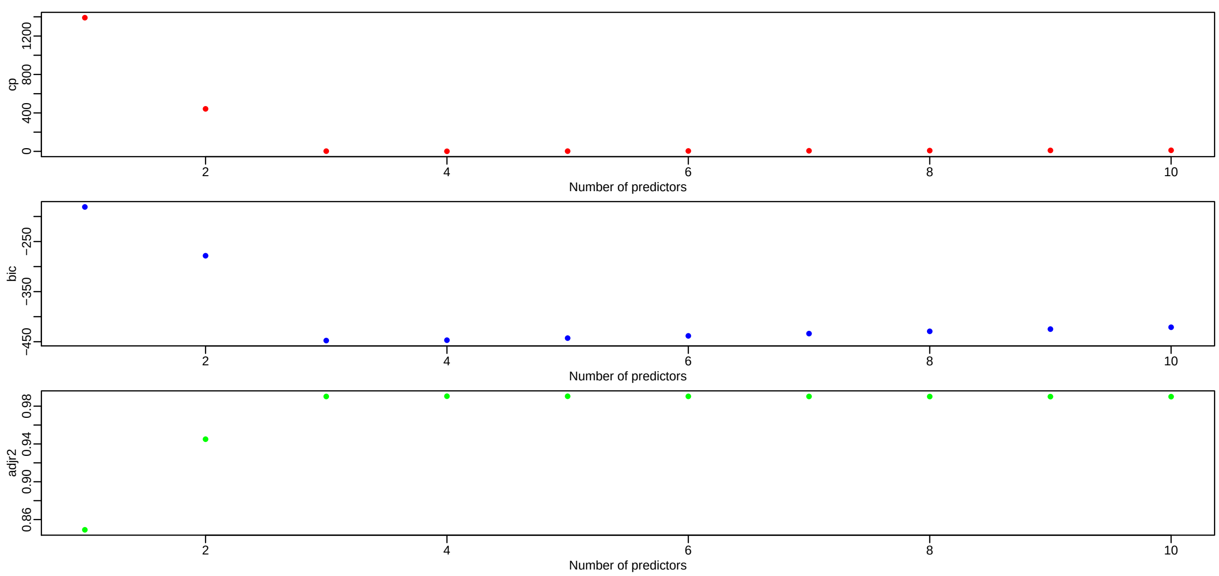
(d) Repeat (c), using forward stepwise selection and also using backwards stepwise selection. How does your answer compare to the results in (c)?

Using Forward Selection

```
fit.forward <- regsubsets(y ~ x + I(x^2) + I(x^3) + I(x^4) + I(x^5) + I(x^6) + I(x^7) + I(x^8) + I(x^9)
f.f <- summary(fit.forward)
```

Plot criteria for comparison

```
par(mfrow=c(3,1), mar=c(2.5,4,0.5,1), mgp=c(1.5,0.5,0))
plot(f.f$cp, xlab="Number of predictors",
     ylab="cp", col="red", type="p", pch=16)
plot(f.f$bic, xlab="Number of predictors",
     ylab="bic", col="blue", type="p", pch=16)
plot(f.f$adjr2, xlab="Number of predictors",
     ylab="adjr2", col="green", type="p", pch=16)
```



We find the Optimal Model by Cp locate the optimal model size by Cp's is 4

```
opt.size <- which.min(f.f$cp)
opt.size
```

```
## [1] 4
```

Now we look for the optimal variables selected

```
fit.f.var <- f.f$which # logic indicators which variables are in
fit.f.var[opt.size,]
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)
##        TRUE        TRUE        TRUE        TRUE       FALSE        TRUE
##      I(x^6)      I(x^7)      I(x^8)      I(x^9)     I(x^10)
##       FALSE       FALSE       FALSE       FALSE       FALSE
```

Find the coefficients

```r
coef(fit.forward,4)
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^5) 
##  5.07200775  4.38745596  2.84575641  1.55797426  0.08072292
```

Fit the final model

```r
fit.final.forward <- lm(y ~ x + I(x^2) + I(x^3) + I(x^5), data)
summary(fit.final.forward)
```

```
## 
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^3) + I(x^5), data = data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.9204 -0.5775 -0.1686  0.5679  2.1322 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  5.07201    0.11795  43.001  < 2e-16 ***
## x            4.38746    0.28172  15.574  < 2e-16 ***
## I(x^2)       2.84576    0.08721  32.631  < 2e-16 ***
## I(x^3)       1.55797    0.24560   6.343 7.58e-09 ***
## I(x^5)       0.08072    0.04167   1.937   0.0557 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.9491 on 95 degrees of freedom
## Multiple R-squared:  0.9909, Adjusted R-squared:  0.9905 
## F-statistic:  2586 on 4 and 95 DF,  p-value: < 2.2e-16
```

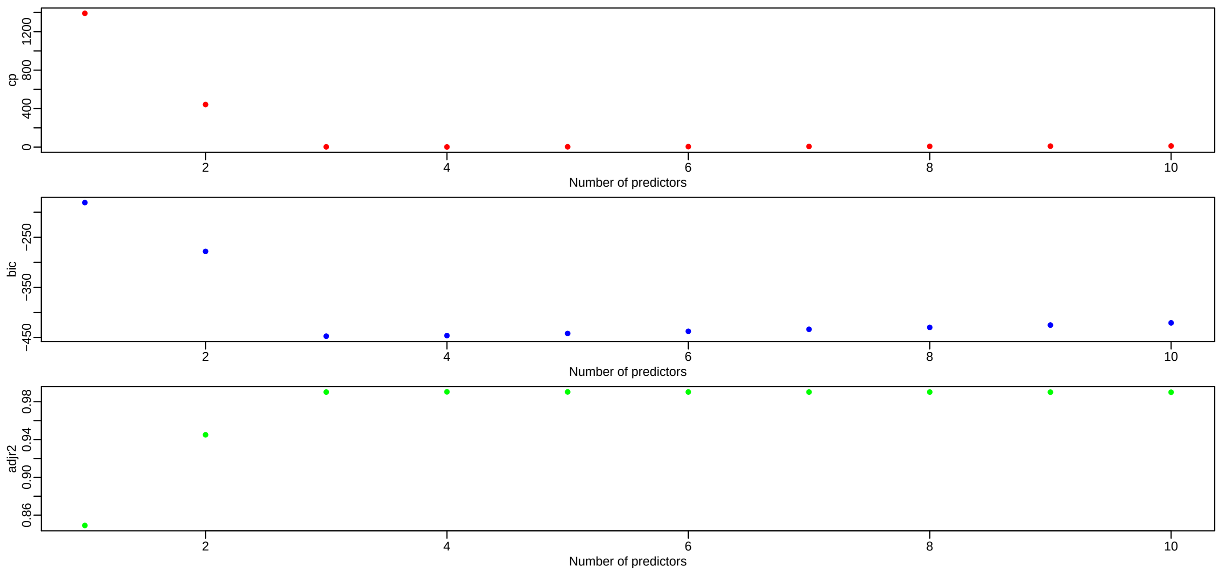
Backward Selection

Using Backward Selection

```r
fit.backward <- regsubsets(y ~ x + I(x^2) + I(x^3) + I(x^4) + I(x^5) + I(x^6) + I(x^7) + I(x^8) + I(x^9
f.b <- summary(fit.backward)
```

Plot criteria for comparison

```r
par(mfrow=c(3,1), mar=c(2.5,4,0.5,1), mgp=c(1.5,0.5,0))
plot(f.b$cp, xlab="Number of predictors",
     ylab="cp", col="red", type="p", pch=16)
plot(f.b$bic, xlab="Number of predictors",
     ylab="bic", col="blue", type="p", pch=16)
plot(f.b$adjr2, xlab="Number of predictors",
     ylab="adjr2", col="green", type="p", pch=16)
```



We find the Optimal Model by Cp locate the optimal model size by Cp's is 4

```
opt.size <- which.min(f.b$cp)
opt.size
```

```
## [1] 4
```

Now we look for the optimal variables selected

```
fit.b.var <- f.b$which # logic indicators which variables are in
fit.b.var[opt.size,]
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
##      I(x^6)      I(x^7)      I(x^8)      I(x^9)     I(x^10)
##       FALSE       FALSE       FALSE        TRUE       FALSE
```

Find the coefficients

```
coef(fit.backward,4)
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^9)
## 5.079236362 4.231905828 2.833494180 1.819555807 0.001290827
```

Fit the final model

```
fit.final.backward <- lm(y ~ x + I(x^2) + I(x^3) + I(x^9), data)
summary(fit.final.backward)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^3) + I(x^9), data = data)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9517 -0.5902 -0.1635  0.5878  2.1755
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.0792364  0.1184665  42.875   <2e-16 ***
## x           4.2319058  0.2321979  18.225   <2e-16 ***
## I(x^2)      2.8334942  0.0890731  31.811   <2e-16 ***
## I(x^3)      1.8195558  0.1255342  14.495   <2e-16 ***
## I(x^9)      0.0012908  0.0007056   1.829   0.0705 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9511 on 95 degrees of freedom
## Multiple R-squared:  0.9909, Adjusted R-squared:  0.9905
## F-statistic:  2575 on 4 and 95 DF,  p-value: < 2.2e-16
```

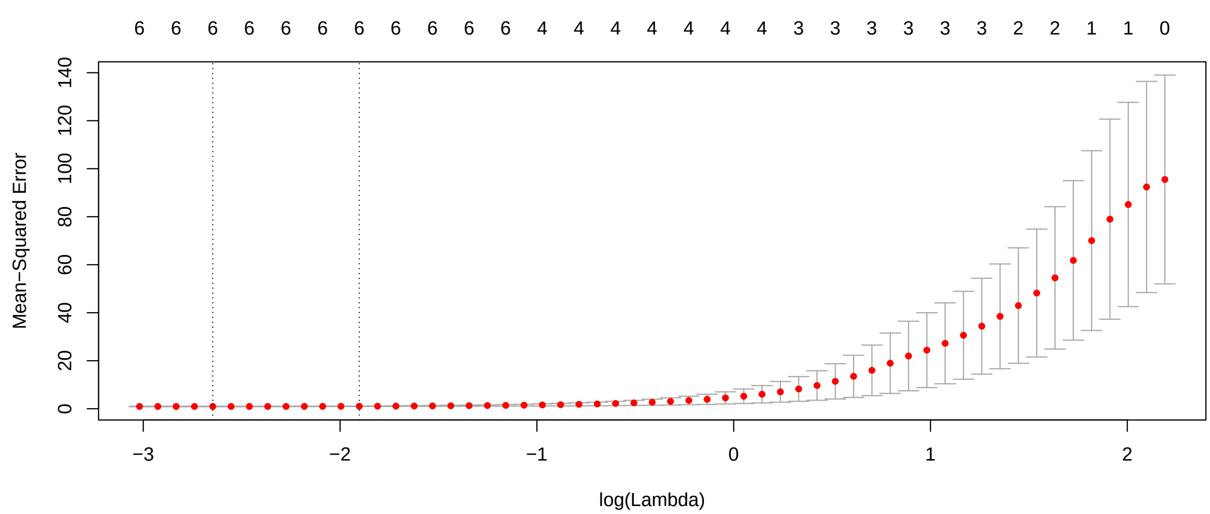
(e) Now fit a lasso model to the simulated data, again using X, X2, . . . ,X10 as predictors. Use cross-validation to select the optimal value of ??. Create plots of the cross-validation error as a function of ??. Report the resulting coefficient estimates, and discuss the results obtained.

prepare the x matrix

```
X <- model.matrix(y ~ x + I(x^2) + I(x^3) + I(x^4) + I(x^5) + I(x^6) + I(x^7) + I(x^8) + I(x^9) + I(x^1
```

To accomplish the Cross Validation, we use the function `cv.glmnet()`

```
fit.cv <- cv.glmnet(X, y, alpha = 1, nfolds = 10)
plot(fit.cv)
```



find λ, we choose to use 1se lambda

```
fit.cv$lambda.1se
```

```
## [1] 0.1491898
```

Find non-zero variables and coefficients, non-zero coefficients are "X"" "I(x^2)" "I(x^3)" "I(x^4)" "I(x^5)" "I(x^7)"

```
coef.1se <- coef(fit.cv, s="lambda.1se")
coef.1se <- coef.1se[which(coef.1se !=0),]
coef.1se
```

```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)
## 5.236711745 4.032740126 2.562445291 1.845121084 0.045313160 0.014629328
##      I(x^7)
## 0.002064488
```

```
var.1se <-rownames(as.matrix(coef.1se))
var.1se
```

```
## [1] "(Intercept)" "x"           "I(x^2)"      "I(x^3)"      "I(x^4)"
## [6] "I(x^5)"      "I(x^7)"
```

prepare for lm fomulae

```
lm.input <- as.formula(paste("y", "~", paste(var.1se[-1], collapse = "+")))
lm.input
```

```
## y ~ x + I(x^2) + I(x^3) + I(x^4) + I(x^5) + I(x^7)
```

Fit the linear model with LASSO output variables

```
fit.1se.lm <-lm(lm.input,data=data)
summary(fit.1se.lm)
```

```
##
## Call:
## lm(formula = lm.input, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9625 -0.5889 -0.1397  0.5544  2.1054
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.123800   0.141295  36.263   <2e-16 ***
## x           4.422672   0.379639  11.650   <2e-16 ***
## I(x^2)      2.685828   0.248398  10.813   <2e-16 ***
## I(x^3)      1.475704   0.589534   2.503   0.0141 *
## I(x^4)      0.041749   0.059924   0.697   0.4877
## I(x^5)      0.129315   0.247049   0.523   0.6019
```

```
## I(x^7)      -0.007111   0.029060  -0.245   0.8072
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9568 on 93 degrees of freedom
## Multiple R-squared:  0.9909, Adjusted R-squared:  0.9904
## F-statistic:  1697 on 6 and 93 DF,  p-value: < 2.2e-16
```

Using LASSO estimation, we obtained 6 non-zero variables which are "X"" "I(x^2)" "I(x^3)" "I(x^4)" "I(x^5)" "I(x^7)". After fitting a linear model to these variables we get a model with adjusted R-square of 0.9904. Also, 3 of 6 variables are significant at 0.05 level.

(f) Describe as accurate as possible what C_p and BIC are estimating?

BIC calculates the probability of the model after seeing the data, when assigning equal probability to each model. In other words, BIC is an estimate of a function of the posterior probability of a model being true, under a certain Bayesian setup, so that a lower BIC means that a model is considered to be more likely to be the true model.

The Cp statistic adds a penalty of 2d??^2 to the training RSS in order to adjust for the fact that the training error tends to underestimate the test error. So the penalty increases as the number of predictors in the model increases; this is intended to adjust for the corresponding decrease in training RSS. If the full model is true, Cp is an unbiased estimator of average prediction errors.

Problem 2:

This will be the last part of the Auto data from ISLR. The original data contains 408 observations about cars. It has some similarity as the Cars data that we use in our lectures. To get the data, first install the package `ISLR`. The data set `Auto` should be loaded automatically. We use this case to go through methods learned so far.

You can access the necessary data with the following code:

```r
# check if you have ISLR package, if not, install it
if(!requireNamespace('ISLR')) install.packages('ISLR')
auto_data <- ISLR::Auto
```

Final modelling question: We want to explore the effects of each feature as best as possible.

You may explore the possibility of variable transformations. We normally do not suggest to transform x for the purpose of interpreation. You may consider to transform y to either correct the violation of the linear model assumptions or if you feel a transformation of y makes more sense from an interpretaation pespective. You may also explore adding interactions and higher order terms. The model(s) should be as *parsimonious* (simple) as possible, unless the gain in accuracy is significant from your point of view. Use Mallow's C_p or BIC to select the model.

Exploratory Data Analysis

How many observations and variables are contained in the data-set?

```r
dim(auto_data)
```

```
## [1] 392   9
```

We have 392 observations and 9 variables.

Variable names

```r
names(auto_data)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"
## [5] "weight"       "acceleration" "year"         "origin"
## [9] "name"
```

Summary of the data

```r
summary(auto_data)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                 name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

Let's see if we have any missing values

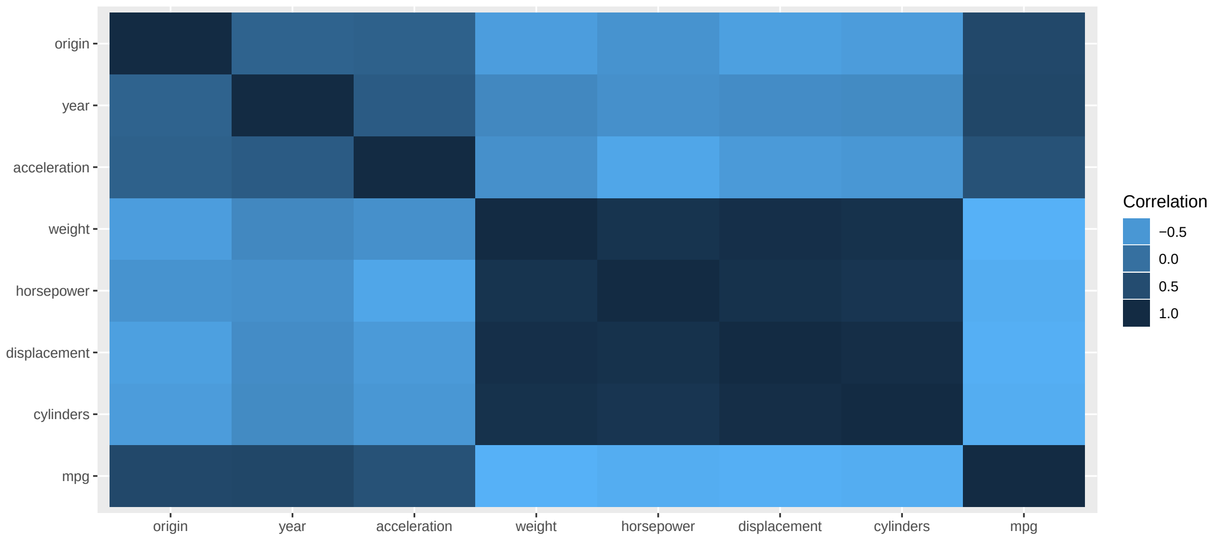
```r
sum(is.na(auto_data)) # this may not work if the missing is not coded as "NA"
```

```
## [1] 0
```

There is no missing value in this data set.

Correlation heatmap

```r
plotData <-melt(cor(auto_data[sapply(auto_data, is.numeric)]))

ggplot(plotData ,
    aes(x = Var1, y = Var2, fill =value)) +
    geom_tile() +
    ylab("") +
    xlab("") +
scale_x_discrete(limits = rev(levels(plotData $Var2))) + #Flip the x- or y-axis
    scale_fill_gradient( low = "#56B1F7", high = "#132B43") +     #lightblue to darkblue
    #scale_fill_gradient( low = "white", high = "black") + #white to black
       guides(fill = guide_legend(title = "Correlation"))
```



model building

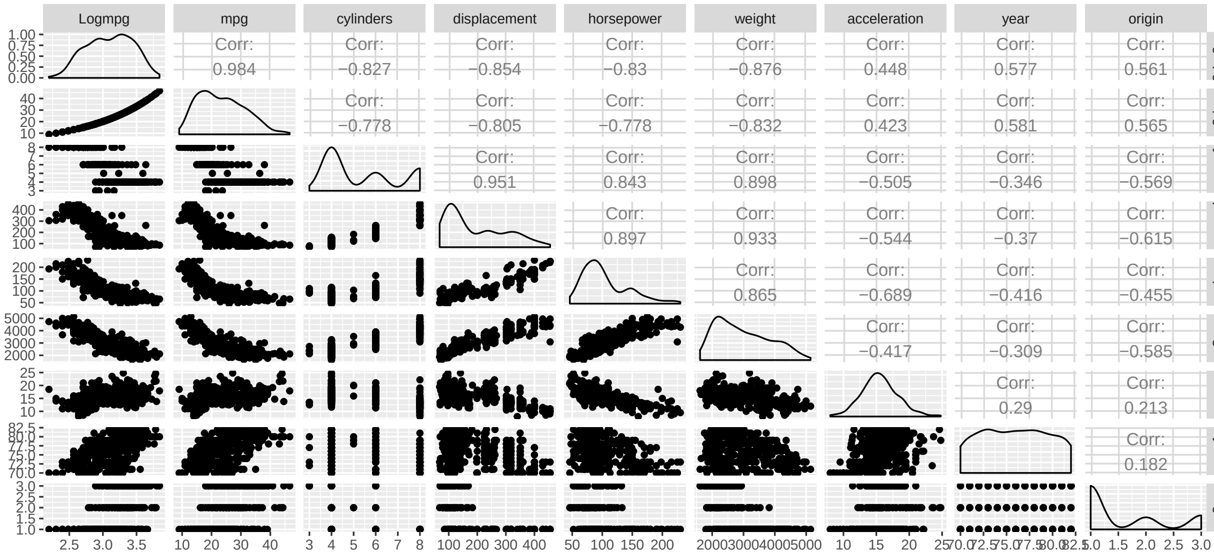
In the linear model assumption, ϵ is normal distribution,which means there is possibility that y is negative. However, it's impossible for "mpg" to be negative or zero. So, we need to use logarithm format of "mpg" as y in our model.

Transform `mpg` to `log(mpg)` and rename the column

```r
data1 <- cbind(log(auto_data$mpg), auto_data)
# data1 <- data.frame(log(data.comp$Salary), data.comp) # Another way of doing the same
# data1 <- data.comp %>% mutate(log_salary = log(Salary)) # dplyr solution
names(data1)[1] <- "Logmpg" # Rename it
```

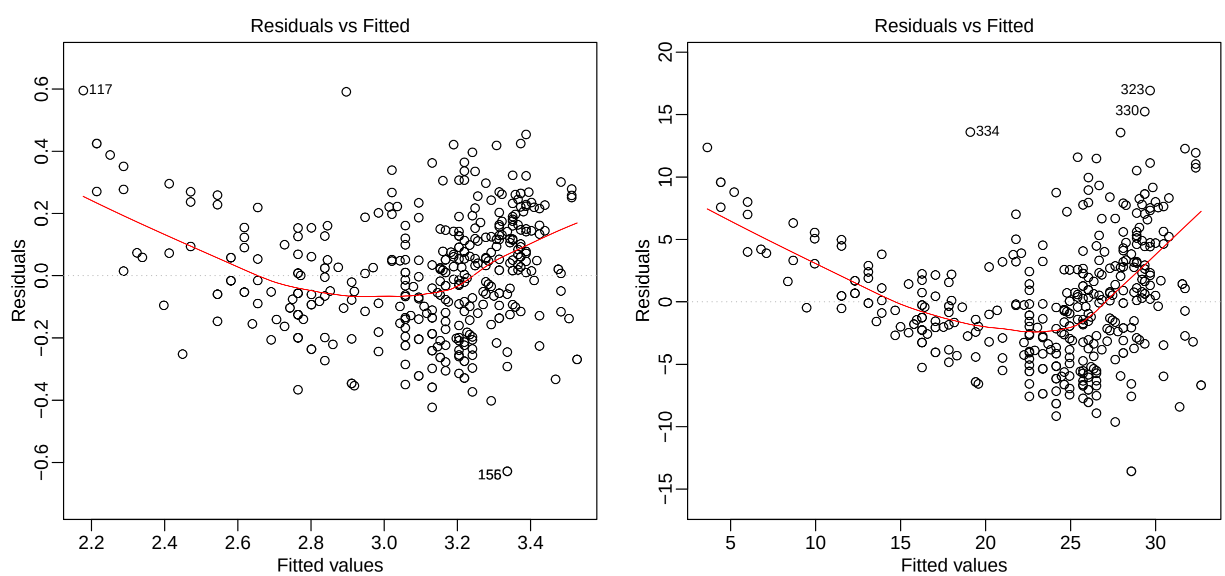
Let's look at some variable transformation candidates.

```r
data1 %>%
  select_if(is.numeric) %>%
  ggpairs()
```



We examine some residual points for a linear regression to see if there are some outliers, heteroscedasticity, normality etc. Compare the residual plot in which the `mpg` variable isn't transformed.

```r
par(mfrow=c(1,2), mar=c(2.5,3,1.5,1), mgp=c(1.5,0.5,0))      # Compare different criteria
plot(lm(Logmpg ~ horsepower, data=data1), 1)
plot(lm(mpg ~ horsepower, data=data1), 1)
```



From the plots, we can see that there is obvious heteroscedasticity if we use "mpg".So, we are going to remove "mpg" from our data set. We also remove "name", which will not appear in our future model.

```r
data2 <- data1[,c(-2,-10)]
summary(data2)
```

```
##      Logmpg         cylinders      displacement     horsepower
##  Min.   :2.197   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:2.833   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :3.125   Median :4.000   Median :151.0   Median : 93.5
```

```
##  Mean   :3.098   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:3.367   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :3.842   Max.   :8.000   Max.   :455.0   Max.   :230.0
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
```

We explore adding interactions but no higher order terms because higher order terms are hard to interprete in this case.

```
fit.exh <- regsubsets(Logmpg ~.*., data2, nvmax=25, method="exhaustive")
names(fit.exh)
```

```
##  [1] "np"        "nrbar"     "d"         "rbar"      "thetab"
##  [6] "first"     "last"      "vorder"    "tol"       "rss"
## [11] "bound"     "nvmax"     "ress"      "ir"        "nbest"
## [16] "lopt"      "il"        "ier"       "xnames"    "method"
## [21] "force.in"  "force.out" "sserr"     "intercept" "lindep"
## [26] "nullrss"   "nn"        "call"
```

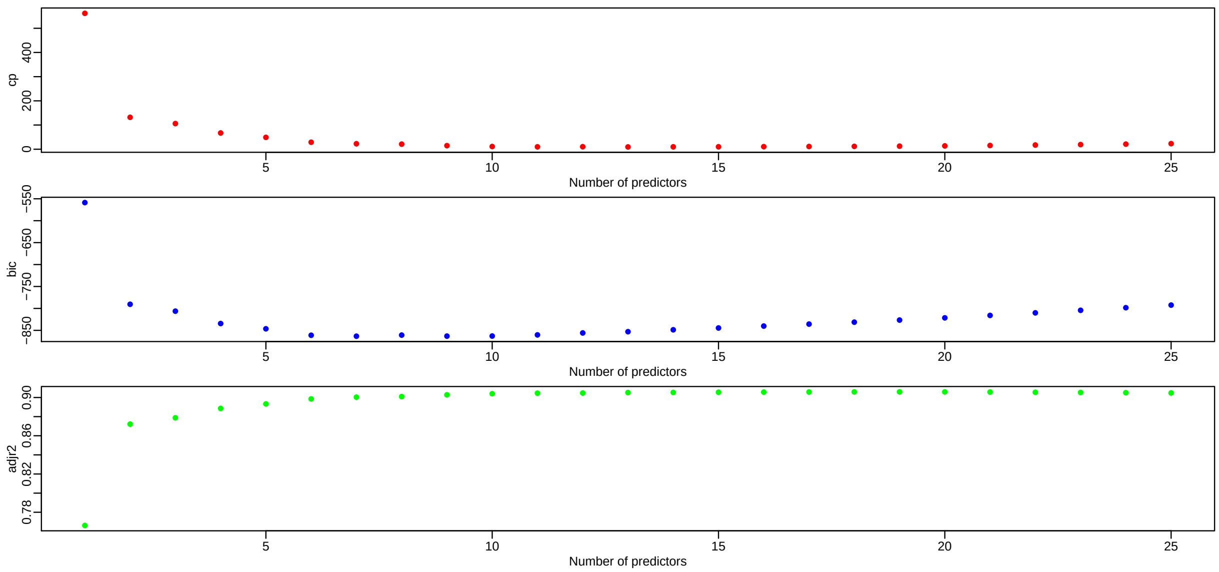
```
f.e <- summary(fit.exh)
names(f.e)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
result<-data.frame(variables = (1:length(f.e$rsq)),
           r_squared = f.e$rsq,
           rss = f.e$rss,
           bic = f.e$bic,
           cp = f.e$cp)
```

Compare different criterion

```
par(mfrow=c(3,1), mar=c(2.5,4,0.5,1), mgp=c(1.5,0.5,0))     # Compare different criteria
plot(f.e$cp, xlab="Number of predictors",
     ylab="cp", col="red", type="p", pch=16)
plot(f.e$bic, xlab="Number of predictors",
     ylab="bic", col="blue", type="p", pch=16)
plot(f.e$adjr2, xlab="Number of predictors",
     ylab="adjr2", col="green", type="p", pch=16)
```



According to elbow rule, we think two variables in the model are the best.

Now we look for the optimal variables selected

```
fit.exh.var <- f.e$which # logic indicators which variables are in
colnames(fit.exh.var)[fit.exh.var[2,]]
```

```
## [1] "(Intercept)" "year"        "weight:year"
```

```
fit.final <- lm(Logmpg ~ year + weight:year, data2)
summary(fit.final)
```

```
##
## Call:
## lm(formula = Logmpg ~ year + weight:year, data = data2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44004 -0.07177  0.00733  0.06756  0.37208
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.736e-01  1.339e-01    5.78 1.54e-08 ***
## year         4.287e-02  1.688e-03   25.39  < 2e-16 ***
## year:weight -4.139e-06  1.019e-07  -40.63  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1216 on 389 degrees of freedom
## Multiple R-squared:  0.8728, Adjusted R-squared:  0.8722
## F-statistic:  1335 on 2 and 389 DF,  p-value: < 2.2e-16
```

```
MSE <- mean(fit.final$residuals^2)
MSE
```

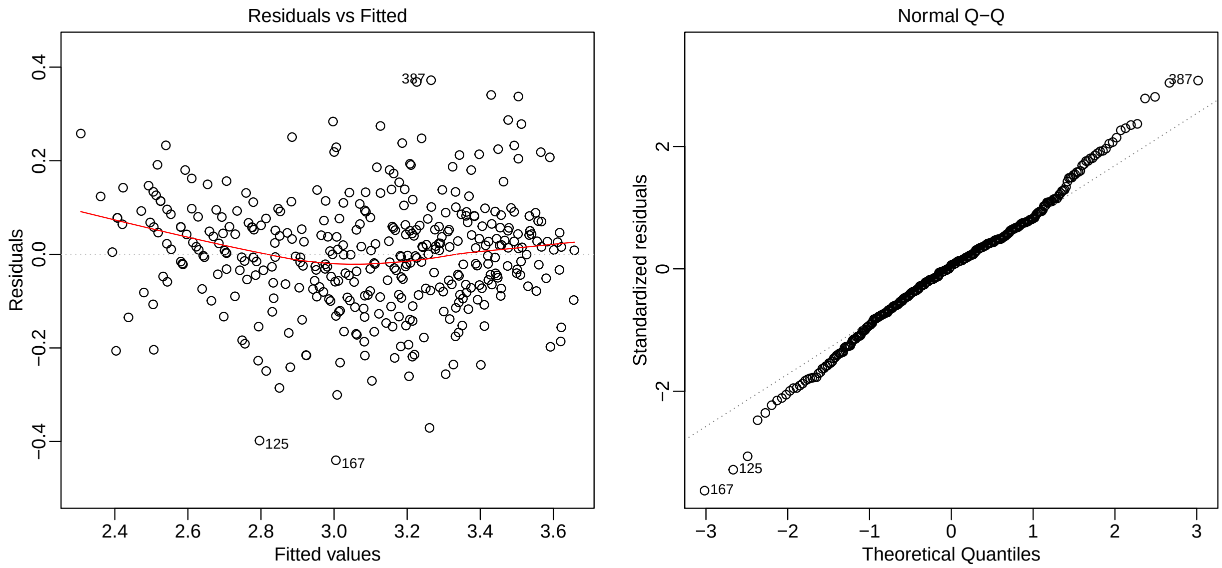
```
## [1] 0.01466486
```

- Describe the final model and its accuracy. Include diagnostic plots with particular focus on the model residuals.

In our final model,we use 'Logmpg' as our dependent variable and 'year','weight:year' as independent variables. All the variables are very significant in final model. The adjusted R-squared is 0.8722, which looks good. Maybe adding some other varibles can also improve accuracy a little bit but we don't think it's worthy to do so.

Model Diagnostics

```r
par(mfrow=c(1,2), mar=c(2.5,3,1.5,1), mgp=c(1.5,0.5,0))
plot(fit.final,1)
plot(fit.final,2)
```



Although there are some outliers and the upper and lower tails of qqplot are off the line,everything is acceptable and looks reasonably fine.

- Summarize the effects found. Year is the most influential variable to mpg. With the development of technology and science,cars bulit in earlier year is very different from those built later. Furthermore, the year effect is significantly different to cars which have different weight.
- Predict the `mpg` of a car that is: built in 1983, in the US, red, 180 inches long, 8 cylinders, 350 displacement, 260 as horsepower, and weighs 4,000 pounds. Give a 95% CI.

```r
newcar <-data2[1,]
newcar[1,] <- NA
newcar["year"] <- 83
newcar["cylinders"] <-8
newcar["displacement"] <- 350
newcar["weight"] <- 4000
newcar["horsepower"] <- 260
newcar["origin"] <- 1
```

```
predict_log<-predict(fit.final,newcar, interval="confidence",se.fit=TRUE)
predict<-exp(predict_log$fit)
predict
```

```
##       fit      lwr      upr
## 1 19.2505 18.56953 19.95645
```

```
# predicted mpg is 19.2505 with a 95% confidence interval of [18.56953, 19.95645]
```

- Any suggestions as to how to improve the quality of the study? They should add more variables which are related to year to the dataset. In the other word, they should know what on earth affect mpg over the time,for example,what's the specific techniques they use to build the car or the matetial each year and so on.

Problem 3: LASSO

Part I: EDA

Crime data continuation: We continue to use the crime data analyzed in the lectures. We first would like to visulize how crime rate (`violentcrimes.perpop`) distributes by states. The follwoing `r`-chunk will read in the entire crime data into the `r`-path and it also creates a subset.

```
setwd("E:/Semester 2/Modern data mining/Homework 2")
crime.all <- read.csv("CrimeData.csv", stringsAsFactors = F, na.strings = c("?"))
crime <- dplyr::filter(crime.all, state %in% c("FL", "CA"))
```

Show a heatmap displaying the mean violent crime by state. You may also show a couple of your favorite summary statistics by state through the heatmaps. Write a brief summary based on your findings.

```
#create a new dataframe
data.heat<- crime.all%>%
  group_by(state)%>%
  summarise(
    mean.income= mean(med.income),
    crime.rate= mean(violentcrimes.perpop,na.rm=TRUE),
    poverty= mean(pct.pop.underpov),
    density=mean(pop.density))
summary(data.heat)
```

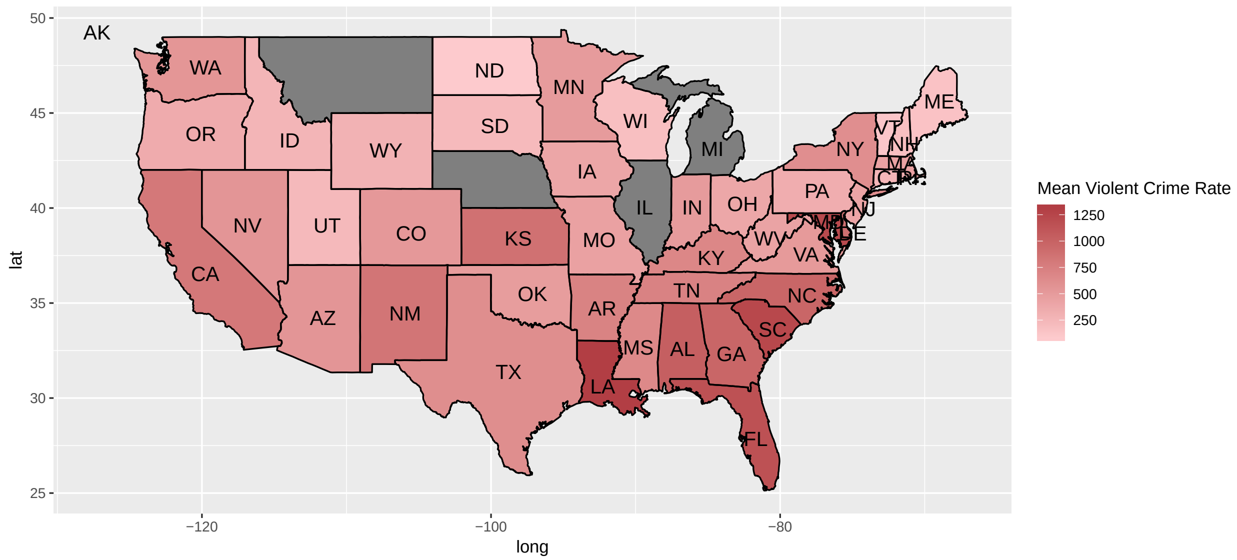
```
##     state             mean.income      crime.rate         poverty
##  Length:48          Min.   :19845   Min.   :  85.06   Min.   : 4.812
##  Class :character   1st Qu.:25903   1st Qu.: 344.41   1st Qu.:10.385
##  Mode  :character   Median :29330   Median : 520.45   Median :12.513
##                     Mean   :30442   Mean   : 623.97   Mean   :13.173
##                     3rd Qu.:32957   3rd Qu.: 788.79   3rd Qu.:16.249
##                     Max.   :47769   Max.   :3048.38   Max.   :27.205
##                                     NA's   :2
##     density
##  Min.   : 363
##  1st Qu.:1589
```

```
##   Median :2148
##   Mean   :2338
##   3rd Qu.:2534
##   Max.   :9539
##
```

```
#match the data to map
data.heat$region <- tolower(state.name[match(data.heat$state, state.abb)])
data.heat$center_lat <- state.center$x[match(data.heat$state, state.abb)]
data.heat$center_long <- state.center$y[match(data.heat$state, state.abb)]
states <- map_data("state")
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map
```

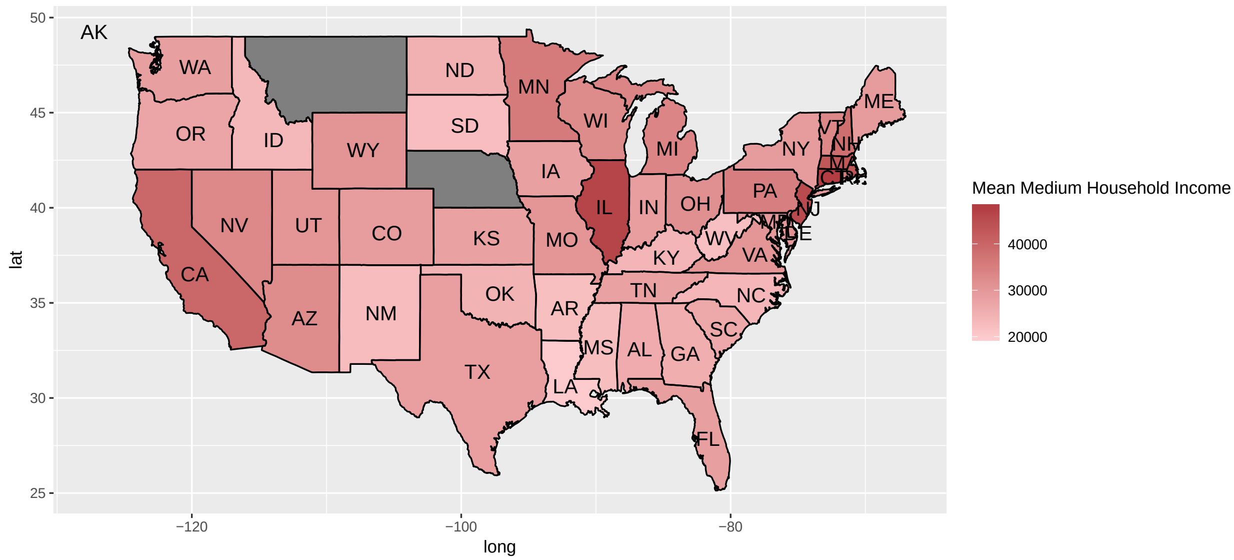
```
map <- merge(states,data.heat, sort=FALSE, by="region", all.x=TRUE)
map <- map[order(map$order),]

#map the mean violent crime rate
ggplot(map,aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=crime.rate))+
  geom_path()+
  geom_text(data=data.heat,aes(x=center_lat,y=center_long,group=NA,label=state,size=2),show.legend= FALS
  scale_fill_continuous(name="Mean Violent Crime Rate",low = "#FDCACD", high = "#B23E44")
```



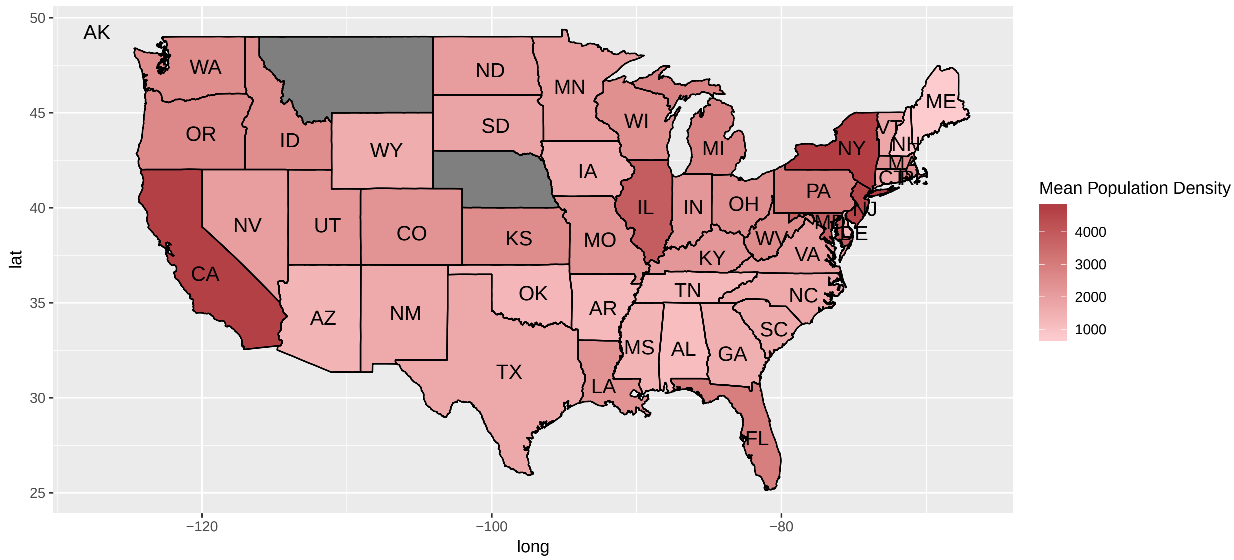

```
# Because the third quartile of crime.rate is much smaller than the max, I decided not to include the l
```

```
#map mean income
ggplot(map,aes(x=long,y=lat,group=group))+
```

```
  geom_polygon(aes(fill=mean.income))+
  geom_path()+
  geom_text(data=data.heat,aes(x=center_lat,y=center_long,group=NA,label=state,size=2),show.legend= FALS
  scale_fill_continuous(name="Mean Medium Household Income",low = "#FDCACD", high = "#B23E44")
```

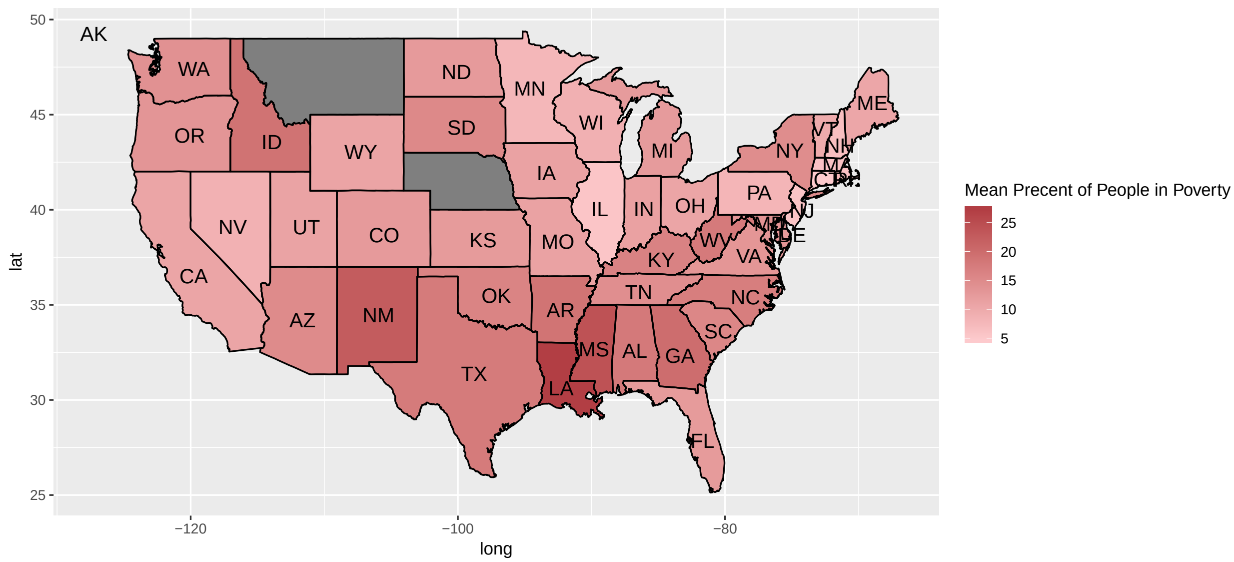



```
#map population density
ggplot(map,aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=density))+
  geom_path()+
  geom_text(data=data.heat,aes(x=center_lat,y=center_long,group=NA,label=state,size=2),show.legend= FALS
  scale_fill_continuous(name="Mean Population Density",low = "#FDCACD", high = "#B23E44")
```




```
ggplot(map,aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=poverty))+
  geom_path()+
```

```r
  geom_text(data=data.heat,aes(x=center_lat,y=center_long,group=NA,label=state,size=2),show.legend= FALS
  scale_fill_continuous(name="Mean Precent of People in Poverty",low = "#FDCACD", high = "#B23E44")
```



The crime map demonstrates the mean of violent crimes per 100K people of each state in 1995. Crime rate data of Illinois and Michigan is missing. States in the southeastern part of the country have higher violent crime rates, while states in the northern part have lower rates. More specifically, South Carolina, Louisiana, Florida, and Mariland are among the states of the highest crime rates, whereas North Dakota, Vermont, Wisconsin and Utah are among those of the lowest.

The second map shows the mean income across the country. The northeastern and southwestern parts generally have higher medium household income. The population density map shows a general low density in the United States except that California, New York, and New Jersey stand out with higher density. Finally, the poverty map shows that southeastern states have higher percentages of poverty, while northern states have lower ones. In brief, the spatial patterns of violent crime rates could be similar with the patterns of poverty to an extent. Little similarity is shown among crime rate, income and density.

Part II: LASSO selection

Our goal for the rest of the study is to find the factors that are related to violent crime. We will only use communities from two states `FL` and `CA` to assure the maximum possible number of variables.

1. Prepare a set of sensible factors/variables that you may use to build a model. You may show the R-chunk to show this step. Explain what varibles you may have excluded in the study and why? Or what other variables you have created to be included in the study.

Then use LASSO to choose a reasonable, small model. Fit an OLS model with the variables obtained. The final model should only include variables with p-values < 0.05. Note: you may choose to use "lambda 1st" or "lambda min" to answer the following questions where applicable.

```r
#data cleaning. Goal: no NA in dataset.
dim(na.omit(crime)) #Can't omit all NA- no observation left
```

```
## [1]   0 147
```

```r
crime%>%
  summarise_all(funs(sum(is.na(.))))
```

```
##   community state county community.code fold population household.size
## 1         0     0    366            367    0          0              0
##   race.pctblack race.pctwhite race.pctasian race.pcthisp age.pct12to21
## 1             0             0             0            0             0
##   age.pct12to29 age.pct16to24 age.pct65up num.urban pct.urban med.income
## 1             0             0           0         0         0          0
##   pct.wage.inc pct.farmself.inc pct.inv.inc pct.socsec.inc pct.pubasst.inc
## 1            0                0           0              0               0
##   pct.retire med.family.inc percap.inc white.percap black.percap
## 1          0              0          0            0            0
##   indian.percap asian.percap other.percap hisp.percap num.underpov
## 1             0            0            0           0            0
##   pct.pop.underpov pct.less9thgrade pct.not.hsgrad pct.bs.ormore
## 1                0                0              0             0
##   pct.unemployed pct.employed pct.employed.manuf pct.employed.profserv
## 1              0            0                  0                     0
##   pct.occup.manuf pct.occup.mgmtprof male.pct.divorce male.pct.nvrmarried
## 1               0                  0                0                   0
##   female.pct.divorce total.pct.divorce ave.people.per.fam pct.fam2parents
## 1                  0                 0                  0               0
##   pct.kids2parents pct.youngkids2parents pct.teens2parents
## 1                0                     0                 0
##   pct.workmom.youngkids pct.workmom num.kids.nvrmarried
## 1                     0           0                   0
##   pct.kids.nvrmarried num.immig pct.immig.recent pct.immig.recent5
## 1                   0         0                0                 0
##   pct.immig.recent8 pct.immig.recent10 pct.pop.immig pct.pop.immig5
## 1                 0                  0             0              0
##   pct.pop.immig8 pct.pop.immig10 pct.english.only pct.no.english.well
## 1              0               0                0                   0
##   pct.fam.hh.large pct.occup.hh.large ave.people.per.hh
## 1                0                  0                 0
##   ave.people.per.ownoccup.hh ave.people.per.rented.hh
## 1                          0                        0
##   pct.people.ownoccup.hh pct.people.dense.hh pct.hh.less3br med.num.br
## 1                      0                   0              0          0
##   num.vacant.house pct.house.occup pct.house.ownoccup pct.house.vacant
## 1                0               0                  0                0
##   pct.house.vacant.6moplus med.yr.house.built pct.house.nophone
## 1                        0                  0                 0
##   pct.house.no.plumb value.ownoccup.house.lowquart value.ownoccup.med
## 1                  0                             0                  0
##   value.ownoccup.highquart ownoccup.qrange rent.lowquart rent.med
## 1                        0               0             0        0
##   rent.highquart rent.qrange med.rent med.rent.aspct.hhinc
## 1              0           0        0                    0
##   med.owncost.aspct.hhinc.wmort med.owncost.as.pct.hhinc.womort
## 1                             0                               0
##   num.in.shelters num.homeless pct.foreignborn pct.born.samestate
## 1               0            0               0                  0
```

```
##   pct.samehouse1985 pct.samecity1985 pct.samestate1985 num.police
## 1                 0                0                 0        294
##   num.police.perpop num.police.fieldops num.police.fieldops.perpop
## 1               294                 294                        294
##   tot.police.requests tot.police.requests.perpop tot.requests.per.police
## 1                 294                        294                     294
##   police.perpop racialmatch.police.to.comm pct.police.white
## 1           294                        294              294
##   pct.police.black pct.police.hisp pct.police.asian pct.police.minority
## 1              294             294              294                 294
##   num.police.drugunits number.drugtypes.seized ave.police.ot.worked
## 1                  294                     294                  294
##   land.area pop.density pct.use.publictransit num.policecars
## 1         0           0                     0            294
##   police.op.budget pct.police.onpatrol gang.unit.deployed
## 1              294                 294                294
##   pct.police.drugunits police.op.budget.perpop num.murders murder.perpop
## 1                    0                     294           0             0
##   num.rapes rapes.perpop num.robberies robberies.perpop num.assaults
## 1         0            0             0                0            1
##   assaults.perpop num.burglaries burglaries.perpop num.larcenies
## 1               1              0                 0             0
##   larcenies.perpop num.autothefts autothefts.perpop num.arsons
## 1                0              0                 0          0
##   arsons.perpop violentcrimes.perpop nonviolentcrimes.perpop
## 1             0                    1                       0
```

```r
# Community code and police data has many NAs. We decide to exclude them.
# We don't care about other kinds of crime in this analysis. We decide to exclude them.
# Moreover, we exlude the variables that are calculate from others.
var_out <- c("county","community.code", "fold","num.police","num.police.perpop","num.police.fieldops",
             "num.police.fieldops.perpop","tot.police.requests","tot.police.requests.perpop","tot.reques
             "police.perpop","racialmatch.police.to.comm","pct.police.white","pct.police.black","pct.pol
             "pct.police.asian","pct.police.minority","num.police.drugunits","number.drugtypes.seized",
              "num.policecars","police.op.budget","pct.police.onpatrol","gang.unit.deployed","num.urban
             "num.larcenies", "num.autothefts", "num.arsons","murder.perpop", "rapes.perpop","robberies
             "assaults.perpop","burglaries.perpop", "larcenies.perpop","autothefts.perpop", "arsons.perp
             "nonviolentcrimes.perpop","pct.police.drugunits")

crime.clean<- crime[!(names(crime)%in% var_out)]
crime.clean%>%
  summarise_all(funs(sum(is.na(.))))
```

```
##   community state population household.size race.pctblack race.pctwhite
## 1         0     0          0              0             0             0
##   race.pctasian race.pcthisp age.pct12to21 age.pct12to29 age.pct16to24
## 1             0            0             0             0             0
##   age.pct65up pct.urban med.income pct.wage.inc pct.farmself.inc
## 1           0         0          0            0                0
##   pct.inv.inc pct.socsec.inc pct.pubasst.inc pct.retire med.family.inc
## 1           0              0               0          0              0
##   percap.inc white.percap black.percap indian.percap asian.percap
## 1          0            0            0             0            0
##   hisp.percap pct.pop.underpov pct.less9thgrade pct.not.hsgrad
```

```
## 1           0                 0                 0               0
##   pct.bs.ormore pct.unemployed pct.employed pct.employed.manuf
## 1             0              0            0                  0
##   pct.employed.profserv pct.occup.manuf pct.occup.mgmtprof
## 1                     0               0                  0
##   male.pct.divorce male.pct.nvrmarried female.pct.divorce
## 1                0                   0                  0
##   total.pct.divorce ave.people.per.fam pct.fam2parents pct.kids2parents
## 1                 0                  0               0                0
##   pct.youngkids2parents pct.teens2parents pct.workmom.youngkids
## 1                     0                 0                     0
##   pct.workmom num.kids.nvrmarried pct.kids.nvrmarried num.immig
## 1           0                   0                   0         0
##   pct.immig.recent pct.immig.recent5 pct.immig.recent8 pct.immig.recent10
## 1                0                 0                 0                  0
##   pct.pop.immig pct.pop.immig5 pct.pop.immig8 pct.pop.immig10
## 1             0              0              0               0
##   pct.english.only pct.no.english.well pct.fam.hh.large pct.occup.hh.large
## 1                0                   0                0                  0
##   ave.people.per.hh ave.people.per.ownoccup.hh ave.people.per.rented.hh
## 1                 0                          0                        0
##   pct.people.ownoccup.hh pct.people.dense.hh pct.hh.less3br med.num.br
## 1                      0                   0              0          0
##   pct.house.occup pct.house.ownoccup pct.house.vacant
## 1               0                  0                0
##   pct.house.vacant.6moplus med.yr.house.built pct.house.nophone
## 1                        0                  0                 0
##   pct.house.no.plumb value.ownoccup.house.lowquart value.ownoccup.med
## 1                  0                             0                  0
##   value.ownoccup.highquart ownoccup.qrange rent.lowquart rent.med
## 1                        0               0             0        0
##   rent.highquart rent.qrange med.rent med.rent.aspct.hhinc
## 1              0           0        0                    0
##   med.owncost.aspct.hhinc.wmort med.owncost.as.pct.hhinc.womort
## 1                             0                               0
##   num.in.shelters num.homeless pct.foreignborn pct.born.samestate
## 1               0            0               0                  0
##   pct.samehouse1985 pct.samecity1985 pct.samestate1985 land.area
## 1                 0                0                 0         0
##   pop.density pct.use.publictransit violentcrimes.perpop
## 1           0                     0                    1
```

```r
crime.clean<- na.omit(crime.clean)
```

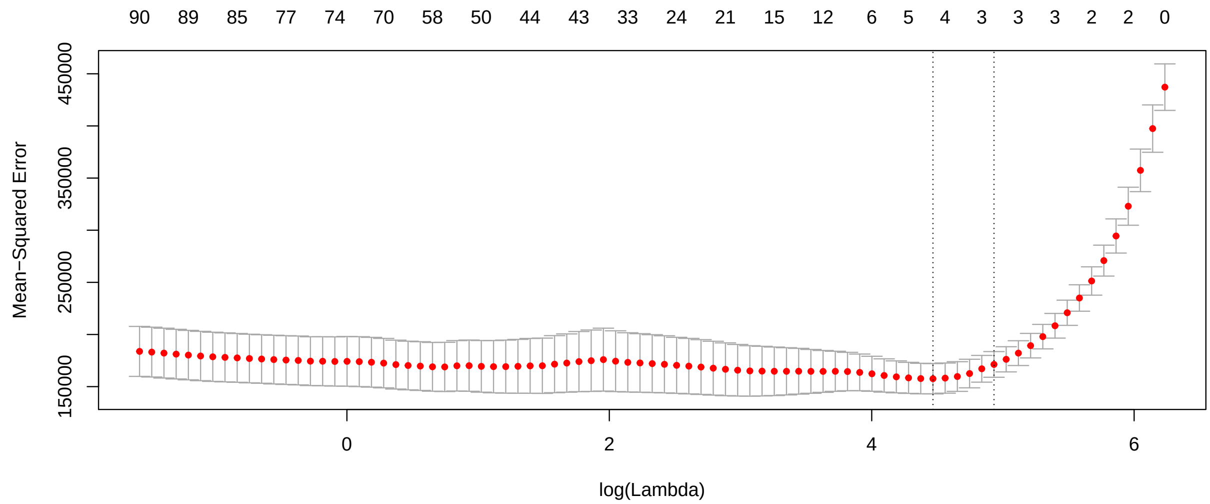
After data cleaning, we have a dataset of 100 variables and 368 observations without NA. The first two indicate county and state, which will not enter the models.

2. What is the model reported by LASSO?

```r
#LASSO
#prepare data
Y<- crime.clean[,100]
X<- model.matrix(violentcrimes.perpop~.,data=crime.clean[,3:100])[,-1]
colnames(X)
```

```
##  [1] "population"                        "household.size"
##  [3] "race.pctblack"                     "race.pctwhite"
##  [5] "race.pctasian"                     "race.pcthisp"
##  [7] "age.pct12to21"                     "age.pct12to29"
##  [9] "age.pct16to24"                     "age.pct65up"
## [11] "pct.urban"                         "med.income"
## [13] "pct.wage.inc"                      "pct.farmself.inc"
## [15] "pct.inv.inc"                       "pct.socsec.inc"
## [17] "pct.pubasst.inc"                   "pct.retire"
## [19] "med.family.inc"                    "percap.inc"
## [21] "white.percap"                      "black.percap"
## [23] "indian.percap"                     "asian.percap"
## [25] "hisp.percap"                       "pct.pop.underpov"
## [27] "pct.less9thgrade"                  "pct.not.hsgrad"
## [29] "pct.bs.ormore"                     "pct.unemployed"
## [31] "pct.employed"                      "pct.employed.manuf"
## [33] "pct.employed.profserv"             "pct.occup.manuf"
## [35] "pct.occup.mgmtprof"                "male.pct.divorce"
## [37] "male.pct.nvrmarried"               "female.pct.divorce"
## [39] "total.pct.divorce"                 "ave.people.per.fam"
## [41] "pct.fam2parents"                   "pct.kids2parents"
## [43] "pct.youngkids2parents"             "pct.teens2parents"
## [45] "pct.workmom.youngkids"             "pct.workmom"
## [47] "num.kids.nvrmarried"               "pct.kids.nvrmarried"
## [49] "num.immig"                         "pct.immig.recent"
## [51] "pct.immig.recent5"                 "pct.immig.recent8"
## [53] "pct.immig.recent10"                "pct.pop.immig"
## [55] "pct.pop.immig5"                    "pct.pop.immig8"
## [57] "pct.pop.immig10"                   "pct.english.only"
## [59] "pct.no.english.well"               "pct.fam.hh.large"
## [61] "pct.occup.hh.large"                "ave.people.per.hh"
## [63] "ave.people.per.ownoccup.hh"        "ave.people.per.rented.hh"
## [65] "pct.people.ownoccup.hh"            "pct.people.dense.hh"
## [67] "pct.hh.less3br"                    "med.num.br"
## [69] "pct.house.occup"                   "pct.house.ownoccup"
## [71] "pct.house.vacant"                  "pct.house.vacant.6moplus"
## [73] "med.yr.house.built"                "pct.house.nophone"
## [75] "pct.house.no.plumb"                "value.ownoccup.house.lowquart"
## [77] "value.ownoccup.med"                "value.ownoccup.highquart"
## [79] "ownoccup.qrange"                   "rent.lowquart"
## [81] "rent.med"                          "rent.highquart"
## [83] "rent.qrange"                       "med.rent"
## [85] "med.rent.aspct.hhinc"              "med.owncost.aspct.hhinc.wmort"
## [87] "med.owncost.as.pct.hhinc.womort" "num.in.shelters"
## [89] "num.homeless"                      "pct.foreignborn"
## [91] "pct.born.samestate"                "pct.samehouse1985"
## [93] "pct.samecity1985"                  "pct.samestate1985"
## [95] "land.area"                         "pop.density"
## [97] "pct.use.publictransit"
```

```
#use cv to select lambda
set.seed(1001)
fit.cv<- cv.glmnet(X,Y,alpha=1,nfolds=8) #not a big fold number, because the dataset is not large.
plot(fit.cv)
```




```r
#lambda min
fit.cv$lambda.min
```

```
## [1] 87.07116
```

```r
coef.min <- coef(fit.cv, s="lambda.min")
coef.min <- coef.min[which(coef.min !=0),]
coef.min
```

```
##         (Intercept)       race.pctblack    pct.kids2parents
##         1856.449967            9.341510          -19.285605
## pct.kids.nvrmarried    pct.house.vacant
##           81.518618            2.940911
```

```r
#lambda first
fit.cv$lambda.1se
```

```
## [1] 138.6419
```

```r
coef.1se <- coef(fit.cv, s="lambda.1se")
coef.1se <- coef.1se[which(coef.1se !=0),]
coef.1se
```

```
##         (Intercept)       race.pctblack    pct.kids2parents
##         1768.766553            6.733932          -17.309554
## pct.kids.nvrmarried
##           75.092585
```

Using lambda min (87.07), the model reported by LASSo is: $violentcrimes.perpop = 1856.449967 + 9.341510 * race.pctblack - 19.285605 * pct.kids2parents + 81.518618 * pct.kids.nvrmarried + 2.940911 * pct.house.vacant$

Using lambda first (138.6419), the model reported by LASSo is: $violentcrimes.perpop = 1768.766553 + 6.733932 * race.pctblack - -17.309554 * pct.kids2parents + 75.092585 * pct.kids.nvrmarried$

We decide to adopt the lambda.min model because it already parsimonious and has smaller prediction error.

3. What is the model after running OLS? Comment on the difference between the equation from questions (1) and (2)

```
fit.lm<- lm(violentcrimes.perpop~race.pctblack+ pct.kids2parents+ pct.kids.nvrmarried+ pct.house.vacant
summary(fit.lm)
```

```
##
## Call:
## lm(formula = violentcrimes.perpop ~ race.pctblack + pct.kids2parents +
##     pct.kids.nvrmarried + pct.house.vacant, data = crime.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1046.86  -224.16   -48.04   157.19  1898.65
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1999.426    264.411   7.562 3.29e-13 ***
## race.pctblack         13.112      2.743   4.780 2.56e-06 ***
## pct.kids2parents     -22.686      3.346  -6.779 4.90e-11 ***
## pct.kids.nvrmarried   85.510     12.746   6.709 7.54e-11 ***
## pct.house.vacant      27.754     11.036   2.515   0.0123 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 375.6 on 363 degrees of freedom
## Multiple R-squared:  0.6846, Adjusted R-squared:  0.6811
## F-statistic:   197 on 4 and 363 DF,  p-value: < 2.2e-16
```

```
mse1= mean(fit.lm$residuals^2)
mse1
```

```
## [1] 139158
```

The OLS model using lambda.first variables is: $violentcrimes.perpop = 1999.426 + 13.112 * race.pctblack - 22.686 * pct.kids2parents + 85.510 * pct.kids.nvrmarried + 27.754 * pct.house.vacant$

The coefficients of the OLS model are different from the model reported by LASSO. The direction of the effect of each variable stays the same (either positive or negative). However, the absolute value of each coefficient is greater in the OLS model. This is because that LASSO estimation is biased given its minimization expression. We should use the OLS model.

4. What is your final model, after excluding high p-value variables?

a) What is your process of getting this final model?
b) Write a brief report based on your final model.

```
summary(fit.lm)
```

```
##
## Call:
## lm(formula = violentcrimes.perpop ~ race.pctblack + pct.kids2parents +
```

```
##     pct.kids.nvrmarried + pct.house.vacant, data = crime.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1046.86  -224.16   -48.04   157.19  1898.65
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1999.426    264.411   7.562 3.29e-13 ***
## race.pctblack         13.112      2.743   4.780 2.56e-06 ***
## pct.kids2parents     -22.686      3.346  -6.779 4.90e-11 ***
## pct.kids.nvrmarried   85.510     12.746   6.709 7.54e-11 ***
## pct.house.vacant      27.754     11.036   2.515   0.0123 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 375.6 on 363 degrees of freedom
## Multiple R-squared:  0.6846, Adjusted R-squared:  0.6811
## F-statistic:   197 on 4 and 363 DF,  p-value: < 2.2e-16
```

Each variable already has a p-value smaller than 0.05. Thereore, the OLS model is the final model. This model is obtained by model selection by LASSO and OLS regression.

$violentcrimes.perpop = 1999.426 + 13.112 * race.pctblack - 22.686 * pct.kids2parents + 85.510 * pct.kids.nvrmarried + 27.754 * pct.house.vacant$

Brief report: Using LASSO model selection and OLS modeling, we found four community-level factors related to violent crime in Florida and California in 1995. Percent of Black population, percent of kids whose parents never married, and percent of vacant houses are positively associated with violent crime rate, while percent of kids with two parents is negatively associated.

To be more specific, holding other variables constant, on average the increase of 1 percent of black population of the community results in 13.11 more crimes per 100K people; the increase of 1 percent of kids whose parents never married raises crime number by 85.5 100K people; the increase of 1 percent vacant houses lead to an increase of 27.7 crimes per 100k people; the increase of 1 percent of kids with two parents, however, reduces crimes by 22.7 per 100k people.

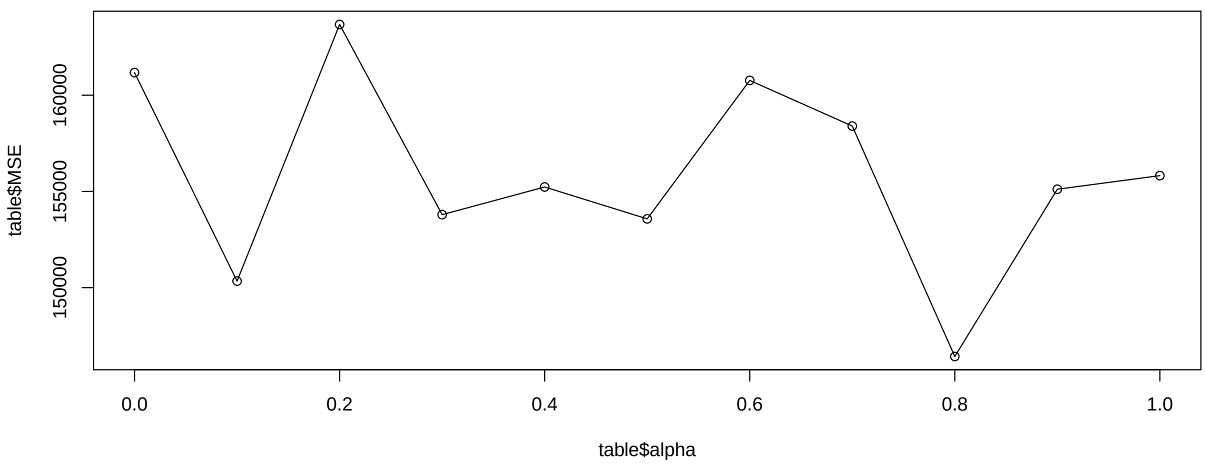
Part III: Elastic Net

Now, instead of LASSO, we want to consider how changing the value of α (i.e. mixing between LASSO and Ridge) will affect the model. Cross-validate between α and λ, instead of just λ. Note that the final model may have variables with p-values higher than 0.05; this is because we are optimizing for accuracy rather than parsimony.

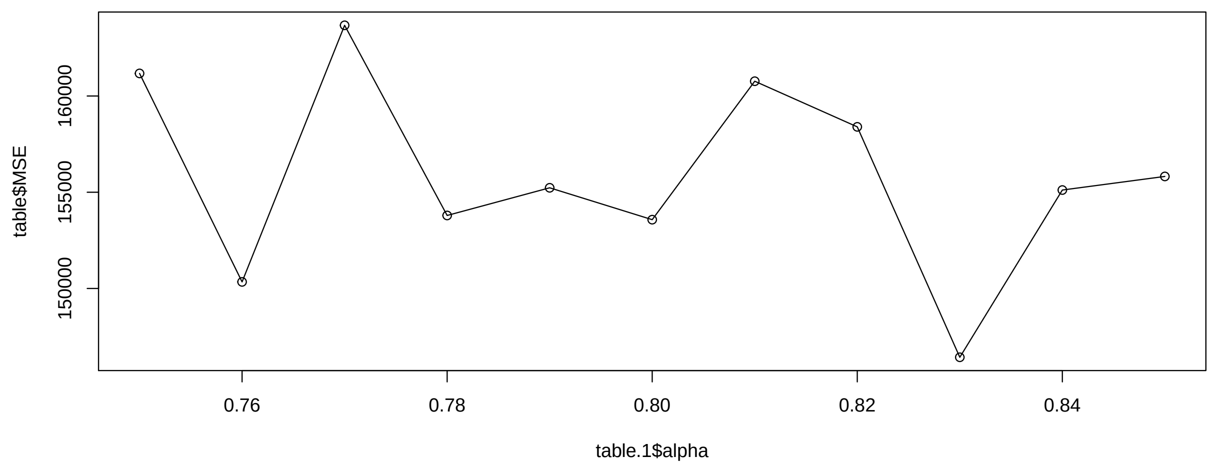
1. What is your final elastic net model? What were the α and λ values? What is the prediction error?

```r
# Goal: cross-validation using different alphas and lambdas to achieve the lowest prediction error (mea
# try with ten alpha values: seq(0,1,0.1)
# have a dataframe to restore alpha, MSE, lambda
# use lambda.min because we are optimizing for accuracy
set.seed(105)
alpha<- seq(0,1,0.1)
MSE<- seq(0,1,0.1)
lambda<- seq(0,1,0.1)
```

```
table<- data.frame(alpha,MSE,lambda)
for(i in 0:10){
  a= i/10
  fit.cv.1<- cv.glmnet(X,Y,alpha=a,nfolds=8)
  table[i+1,2]=min(fit.cv.1$cvm)
  table[i+1,3]=fit.cv.1$lambda.min
}
# plot how mse changes with alpha
plot(table$alpha,table$MSE)
lines(table$alpha,table$MSE)
```




```
#Therefore, we may want a model with alpha close to 0.8
set.seed(105)
#run a new loop
alpha.1<- seq(0.75,0.85,0.01)
table.1<- data.frame(alpha.1,MSE,lambda)
for(i in 0:10){
  a= 0.75+i/100
  fit.cv.1<- cv.glmnet(X,Y,alpha=a,nfolds=8)
  table.1[i+1,2]=min(fit.cv.1$cvm)
  table.1[i+1,3]=fit.cv.1$lambda.min
}
plot(table.1$alpha,table$MSE)
lines(table.1$alpha,table$MSE)
```




```
table.1
```

```
##    alpha.1      MSE    lambda
## 1     0.75 155155.1 41.722339
## 2     0.76 146659.1 25.858071
## 3     0.77 160178.2 53.721893
## 4     0.78 152731.7 33.306364
## 5     0.79 154511.1  2.017779
## 6     0.80 153136.1 39.114693
## 7     0.81 158268.5 97.945682
## 8     0.82 158031.7 31.681664
## 9     0.83 146378.0 34.351664
## 10    0.84 155344.8 78.412075
## 11    0.85 156717.1 77.489580
```

```
fit.elastic<- glmnet(X, Y, alpha=0.83,lambda = 34.351664)
coef<- coef(fit.elastic)
options(scipen = 999)
coef<- coef[which(coef!=0),]
```

Therefore, we end up with alpha=0.83, lambda=lambda.min=31.29996. The prediction error MSE is 148236.0. The final elastic net model is:

```
coef
```

```
##           (Intercept)         race.pctblack      pct.farmself.inc
##       4910.8868405961         16.2969059919        -12.6662204768
##           pct.inv.inc          asian.percap      male.pct.divorce
##         -0.7882430595          0.0020432782         22.0984086198
##     pct.kids2parents pct.youngkids2parents            pct.workmom
##        -16.2713196435         -0.6948855310         -4.3340034659
##   num.kids.nvrmarried   pct.kids.nvrmarried      pct.english.only
##          0.0009058619         60.2519397665         -2.8017333372
##        pct.house.occup      pct.house.vacant    med.yr.house.built
```

```
##        -1.4625338652        11.7368418593        -1.4288053871
##     pct.house.nophone       num.in.shelters
##         7.0146777833         0.0795940456
```

2. Use the elastic net variables in an OLS model. What is the equation, and what is the prediction error?

```r
fit.lm.1<- lm(violentcrimes.perpop~  race.pctblack+ pct.farmself.inc+ pct.inv.inc+ asian.percap+male.pc
summary(fit.lm.1)
```

```
##
## Call:
## lm(formula = violentcrimes.perpop ~ race.pctblack + pct.farmself.inc +
##     pct.inv.inc + asian.percap + male.pct.divorce + pct.kids2parents +
##     pct.youngkids2parents + pct.workmom + num.kids.nvrmarried +
##     pct.kids.nvrmarried + pct.english.only + pct.house.occup +
##     pct.house.vacant + med.yr.house.built + pct.house.nophone +
##     num.in.shelters, data = crime.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1050.71  -183.93   -26.61   141.68  1949.42
##
## Coefficients:
##                          Estimate   Std. Error t value       Pr(>|t|)
## (Intercept)           12098.034673 6409.936439   1.887        0.05993 .
## race.pctblack            22.285544    3.432545   6.492 0.000000000289 ***
## pct.farmself.inc        -61.027318   43.977279  -1.388        0.16611
## pct.inv.inc              -4.773046    3.164100  -1.509        0.13233
## asian.percap              0.007925    0.002533   3.129        0.00190 **
## male.pct.divorce         38.910398   11.950567   3.256        0.00124 **
## pct.kids2parents        -11.490859    6.435386  -1.786        0.07503 .
## pct.youngkids2parents    -1.955190    5.473577  -0.357        0.72115
## pct.workmom              -8.141497    3.769351  -2.160        0.03146 *
## num.kids.nvrmarried       0.001594    0.002587   0.616        0.53806
## pct.kids.nvrmarried      33.552279   18.013617   1.863        0.06335 .
## pct.english.only         -5.803315    2.180617  -2.661        0.00814 **
## pct.house.occup          -7.951433    3.991163  -1.992        0.04712 *
## pct.house.vacant         12.296420   10.959275   1.122        0.26263
## med.yr.house.built       -4.661129    3.206951  -1.453        0.14699
## pct.house.nophone        10.161645   11.445660   0.888        0.37525
## num.in.shelters           0.133583    0.080983   1.650        0.09993 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 351.1 on 351 degrees of freedom
## Multiple R-squared:  0.7334, Adjusted R-squared:  0.7213
## F-statistic: 60.36 on 16 and 351 DF,  p-value: < 0.0000000000000022
```

```r
mse= mean(fit.lm.1$residuals^2)
mse
```

```
## [1] 117609
```

The equation is shown above. The prediction error/ MSE is 117609.

3. Summarize your findings, with particular focus on the difference between the two equations. There are 16 variables in the model with the least prediction error that we found. Among these variables, race.pctblack, asian.percap, male.pct.divorce, pct.workmom,pct.english.only,pct.house.occup are significant at the lavel of 0.05. Compared to the elastic net model, the OLS model has the coefficients in the same directions (positive/negative) but of greater absolute values. This is because that the elastic net coefficients are biased. We should use the OLS model.