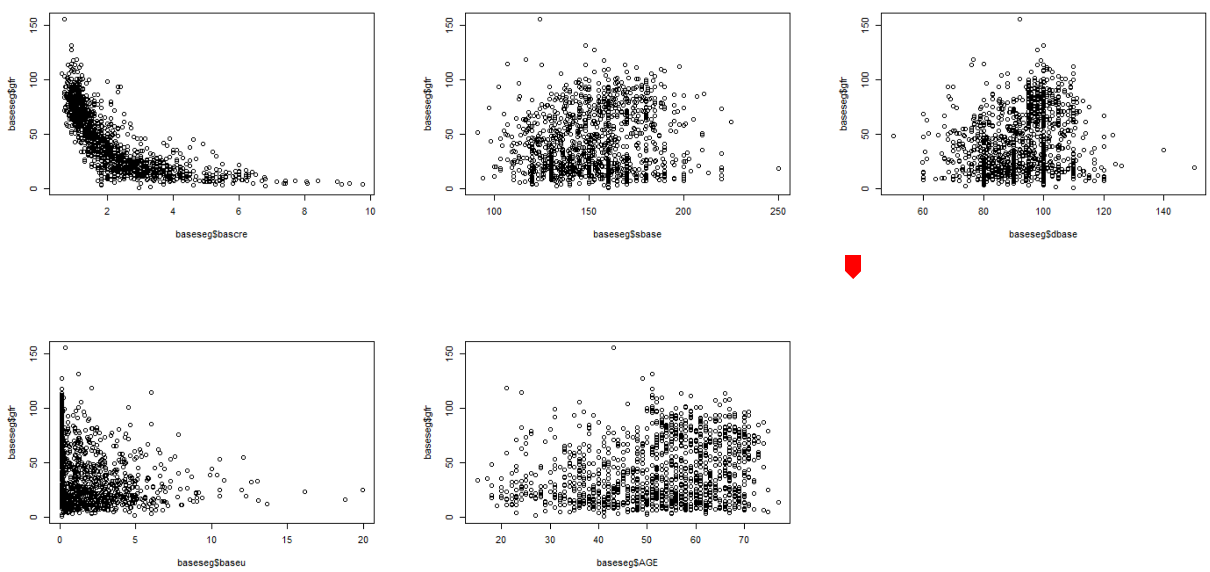
# Homework2

## Yue Peng

Part A:

First, we eliminate the NAs in response variable gfr. Then we start to do linear regressions on those variables: "bascre", "sbase", "dbase", "baseu", "AGE", "SEX", "black". By obtaining the p-value of each slope, we extract the varibles whose p-value is less than 0.05. Finally we get five significant preditors: "bascre", "sbase", "dbase", "baseu", "AGE",

We plot the scatterplot for each potential predictor.




We found the “bascre” predictor had obvious nonlinearity. Thus, we add up the polynomial term step by step. The AIC and adjusted R squared tell us that it is better for us to keep the

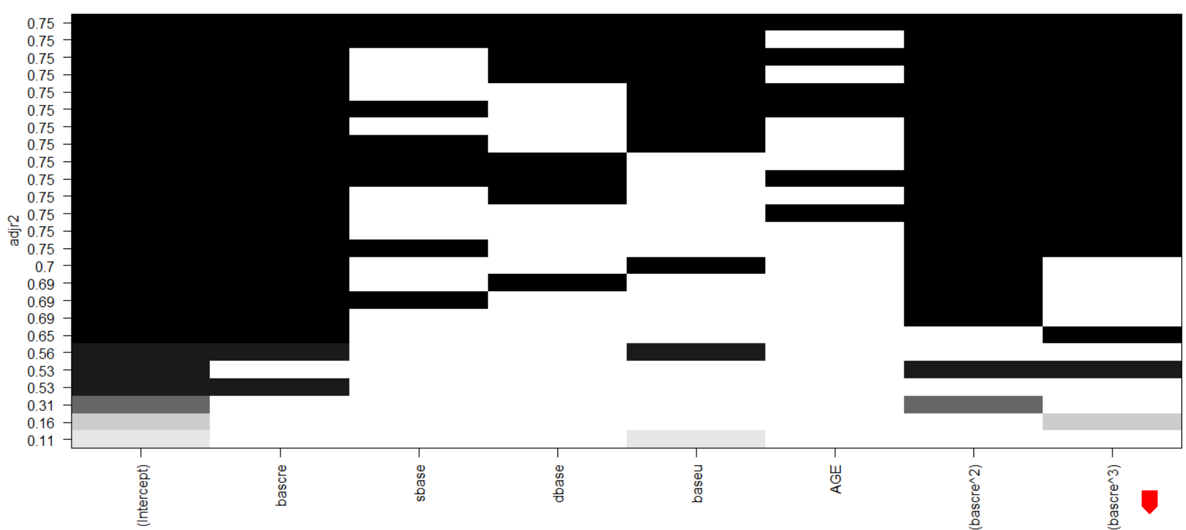
```
Call:
lm(formula = gfr ~ bascre + I(bascre^(2)), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max 
-65.286  -9.696  -0.448   8.980  73.153 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   107.3799     1.4631   73.39   <2e-16 ***
bascre        -39.3495     1.0029  -39.24   <2e-16 ***
I(bascre^(2))   3.6351     0.1401   25.95   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.76 on 1246 degrees of freedom
Multiple R-squared:  0.692,	Adjusted R-squared:  0.6915 
F-statistic:  1400 on 2 and 1246 DF,  p-value: < 2.2e-16
```

```
Call:
lm(formula = gfr ~ bascre + I(bascre^2) + I(bascre^3), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max 
-41.144  -8.170  -1.028   7.241  65.247 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 138.11990    2.24922   61.41   <2e-16 ***
bascre      -75.17292    2.30696  -32.59   <2e-16 ***
I(bascre^2)  14.29358    0.64389   22.20   <2e-16 ***
I(bascre^3)  -0.85318    0.05054  -16.88   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.22 on 1245 degrees of freedom
Multiple R-squared:  0.7493,	Adjusted R-squared:  0.7487 
F-statistic:  1241 on 3 and 1245 DF,  p-value: < 2.2e-16
```

The AIC for these two models are 10438.02 and 10182.57

```
Call:
lm(formula = gfr ~ bascre + I(bascre^2) + I(bascre^3) + I(bascre^4),
    data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max 
-38.773  -7.786  -1.362   6.558  66.441 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  161.58874    3.94501  40.960  < 2e-16 ***
bascre      -111.33557    5.52474 -20.152  < 2e-16 ***
I(bascre^2)   31.08876    2.42465  12.822  < 2e-16 ***
I(bascre^3)   -3.74174    0.40567  -9.224  < 2e-16 ***
I(bascre^4)    0.16031    0.02235   7.174 1.25e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 13.94 on 1244 degrees of freedom
Multiple R-squared:  0.7593,	Adjusted R-squared:  0.7585 
F-statistic: 981.1 on 4 and 1244 DF,  p-value: < 2.2e-16
```

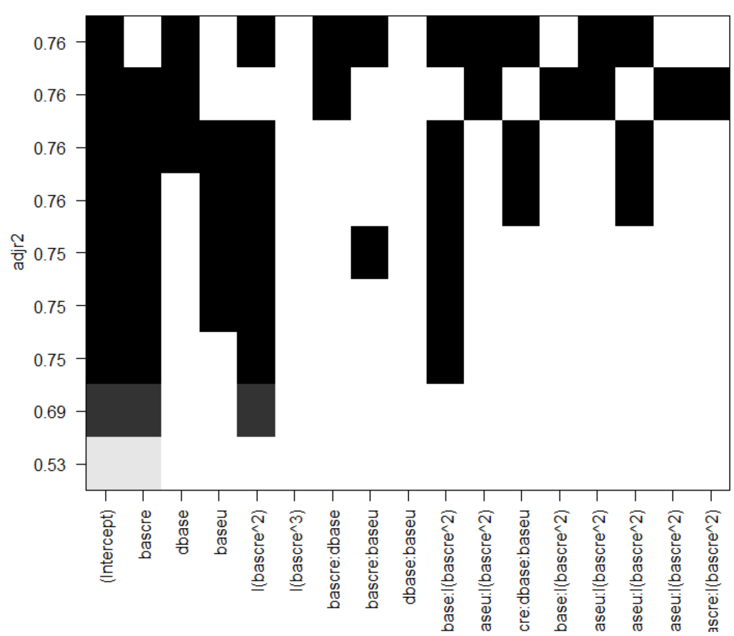
This model had an AIC=10133.93 and did not improve significantly on adjusted R squared. For our convenience, we just introduced third order term in our model. Also, we keep the “bascre” to respect the hierarchy.




Then we implemented full subset regression with these predictors. The graph showed us all predictors could be taken into consideration.

```
Call:
lm(formula = gfr ~ bascre + sbase + dbase + baseu + AGE + I(bascre^2) +
    I(bascre^3) + I(bascre^4), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-40.939  -7.598  -1.557   6.758  66.271

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  161.22230    5.97844  26.967  < 2e-16 ***
bascre      -109.90523    5.71591 -19.228  < 2e-16 ***
sbase         -0.05303    0.02605  -2.035 0.042036 *
dbase          0.12415    0.04878   2.545 0.011050 *
baseu         -0.61487    0.18429  -3.336 0.000874 ***
AGE           -0.06448    0.03454  -1.867 0.062188 .
I(bascre^2)   30.60733    2.47532  12.365  < 2e-16 ***
I(bascre^3)   -3.66541    0.41114  -8.915  < 2e-16 ***
I(bascre^4)    0.15590    0.02254   6.915 7.48e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 13.84 on 1240 degrees of freedom
Multiple R-squared:  0.7638,    Adjusted R-squared:  0.7623
F-statistic: 501.2 on 8 and 1240 DF,  p-value: < 2.2e-16

Correlation of Coefficients:
            (Intercept) bascre sbase dbase baseu AGE   I(bascre^2) I(bascre^3)
bascre      -0.79
sbase       -0.07        0.08
dbase       -0.40        0.06  -0.70
baseu        0.04       -0.15  -0.07  0.05
AGE         -0.28        0.08  -0.38  0.19  0.17
I(bascre^2)  0.75       -0.98  -0.07 -0.06  0.12 -0.06
I(bascre^3) -0.70        0.95   0.06  0.06 -0.09  0.05 -0.99
I(bascre^4)  0.66       -0.91  -0.05 -0.06  0.08 -0.04  0.97       -0.99
```

Also, we found out “sbase” and “dbase” has negative pairwise correlation and the coefficient and p-value of “sbase” are smaller than those of “dbase”. And “AGE” was not quite significant. We decided to remove “sbase” and “AGE” in our future analysis.



```
Call:
lm(formula = gfr ~ -1 + (bascre + dbase + baseu)^3 + I(bascre^2) +
    I(bascre^3), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-42.899  -7.858  -1.065   7.881  70.716

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
bascre             -18.55656    2.38459  -7.782 1.50e-14 ***
dbase                1.29311    0.02320  55.740  < 2e-16 ***
baseu               24.76714    2.95939   8.369  < 2e-16 ***
I(bascre^2)          9.35611    0.68627  13.633  < 2e-16 ***
I(bascre^3)         -0.53325    0.05442  -9.799  < 2e-16 ***
bascre:dbase        -0.40198    0.02026 -19.843  < 2e-16 ***
bascre:baseu        -7.86003    1.02841  -7.643 4.23e-14 ***
dbase:baseu         -0.29432    0.03219  -9.143  < 2e-16 ***
bascre:dbase:baseu   0.09168    0.01106   8.288 2.97e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.87 on 1240 degrees of freedom
Multiple R-squared:  0.9162,    Adjusted R-squared:  0.9155
F-statistic:  1505 on 9 and 1240 DF,  p-value: < 2.2e-16
```

The scatter plot of “bascre” looks like the nike function. Thus, we introduce the reciprocal term.

```
Call:
lm(formula = gfr ~ -1 + (dbase + baseu + bascre)^3 + I(bascre^-1) +
    I(bascre^2) + I(bascre^3), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-40.994  -7.764  -1.499   6.953  65.123

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
dbase               0.42662    0.08343   5.113 3.67e-07 ***
baseu               5.72459    3.33745   1.715  0.08655 .
bascre             -6.68978    2.53313  -2.641  0.00837 **
I(bascre^-1)       53.13856    4.93242  10.773  < 2e-16 ***
I(bascre^2)         3.04381    0.87993   3.459  0.00056 ***
I(bascre^3)        -0.16498    0.06228  -2.649  0.00817 **
dbase:baseu        -0.08125    0.03660  -2.220  0.02660 *
dbase:bascre       -0.13450    0.03150  -4.270 2.10e-05 ***
baseu:bascre       -1.99294    1.12446  -1.772  0.07658 .
dbase:baseu:bascre  0.02634    0.01220   2.159  0.03103 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.23 on 1239 degrees of freedom
Multiple R-squared:  0.9233,    Adjusted R-squared:  0.9227
F-statistic:  1492 on 10 and 1239 DF,  p-value: < 2.2e-16
```

The adjusted R squared and AIC=10188.87 looks great. Then we add one more to see what will be going on in the model.

```
Call:
lm(formula = gfr ~ -1 + (dbase + baseu + bascre)^3 + I(bascre^-1) +
    I(bascre^-2) + I(bascre^2) + I(bascre^3), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-40.115  -7.431  -1.491   6.848  64.900

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
dbase               0.050065   0.093832   0.534    0.594
baseu              -3.056722   3.432253  -0.891    0.373
bascre            -18.693186   2.884774  -6.480 1.32e-10 ***
I(bascre^-1)      136.059472  11.361087  11.976  < 2e-16 ***
I(bascre^-2)      -46.985173   5.832090  -8.056 1.83e-15 ***
I(bascre^2)         4.708798   0.882615   5.335 1.13e-07 ***
I(bascre^3)        -0.311916   0.063411  -4.919 9.87e-07 ***
dbase:baseu         0.013456   0.037575   0.358    0.720
dbase:bascre       -0.007419   0.034527  -0.215    0.830
baseu:bascre        0.879167   1.153030   0.762    0.446
dbase:baseu:bascre -0.004572   0.012498  -0.366    0.715
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.87 on 1238 degrees of freedom
Multiple R-squared:  0.9272,    Adjusted R-squared:  0.9265
F-statistic:  1432 on 11 and 1238 DF,  p-value: < 2.2e-16
```

We find out all the interaction terms are not significant now. But the AIC of this model is 10127.05. And the coefficients of "bascre^-1", "bascre", "baseu", "bascre^2" and ""bascre^-2" are both quite large.

```
Call:
lm(formula = gfr ~ -1 + (bascre + baseu) + I(bascre^-1) + I(bascre^-2) +
    I(bascre^2), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-39.972  -7.934  -1.472   6.917  62.275

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
bascre        -5.3957     0.8356  -6.458 1.52e-10 ***
baseu         -0.6841     0.1843  -3.713 0.000214 ***
I(bascre^-1) 107.1286     4.3446  24.658  < 2e-16 ***
I(bascre^-2) -25.3455     3.6997  -6.851 1.15e-11 ***
I(bascre^2)    0.6950     0.1293   5.373 9.22e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.13 on 1244 degrees of freedom
Multiple R-squared:  0.9241,    Adjusted R-squared:  0.9238
F-statistic:  3029 on 5 and 1244 DF,  p-value: < 2.2e-16
```

This model had an AIC with 10166.36

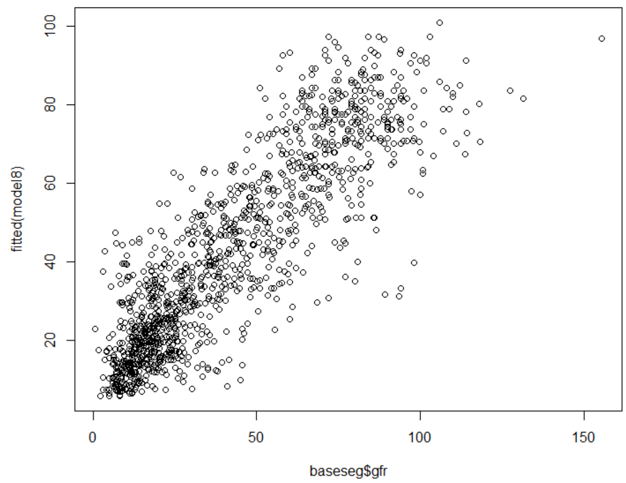
What's more, we decided to add an interaction term.

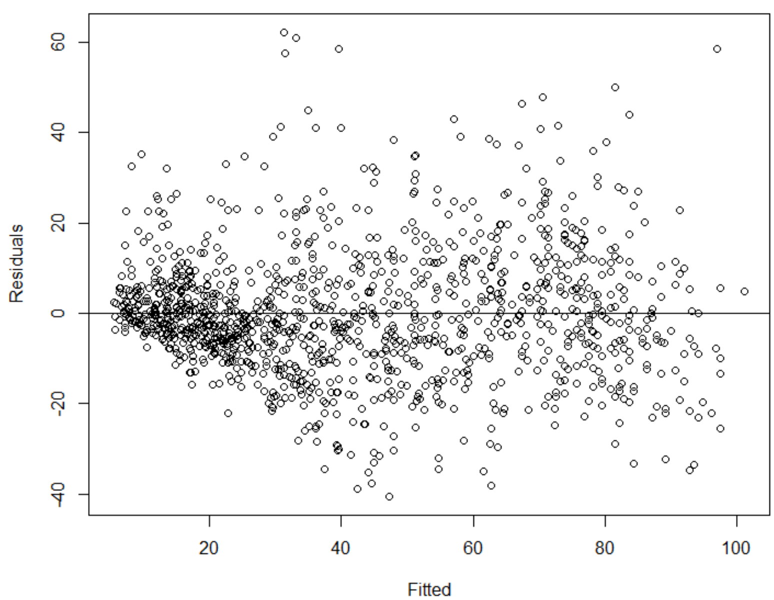
```
Call:
lm(formula = gfr ~ -1 + (bascre * baseu) + I(bascre^-1) + I(bascre^-2) +
    I(bascre^2), data = baseseg)

Residuals:
    Min      1Q  Median      3Q     Max
-40.621  -7.826  -1.362   6.724  62.221

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
bascre        -6.3086     0.8498  -7.424 2.11e-13 ***
baseu         -2.4672     0.4139  -5.960 3.27e-09 ***
I(bascre^-1) 115.0094     4.6089  24.954  < 2e-16 ***
I(bascre^-2) -31.2750     3.8698  -8.082 1.50e-15 ***
I(bascre^2)    0.6969     0.1282   5.435 6.59e-08 ***
bascre:baseu   0.6675     0.1391   4.800 1.78e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14 on 1243 degrees of freedom
Multiple R-squared:  0.9255,    Adjusted R-squared:  0.9251
F-statistic:  2573 on 6 and 1243 DF,  p-value: < 2.2e-16
```

This model had an AIC with 10145.28.

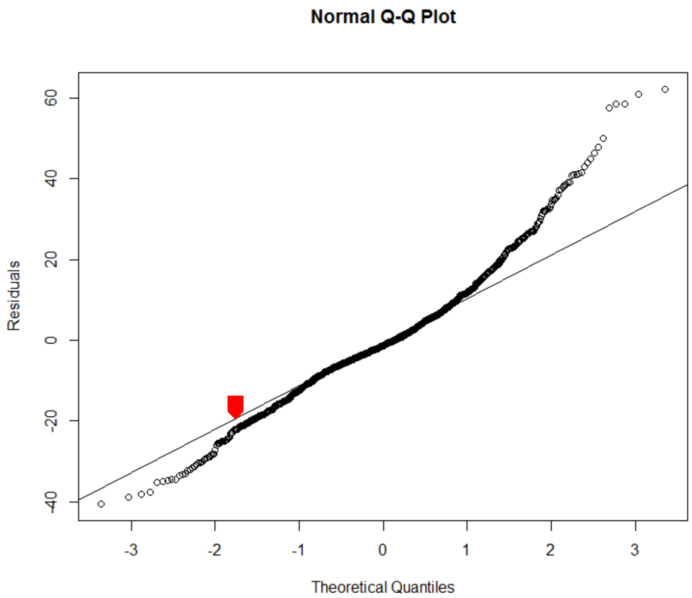
In the end, our model is

$$\mathrm{gfr} = -6.31 \times \mathrm{bascre} - 2.47 \times \mathrm{baseu} + 115.01 \times \frac{1}{bascre} - 31.28 \times \frac{1}{bascre^2} + 0.70 \times bascre^2 + 0.67 \times bascre \times baseu$$

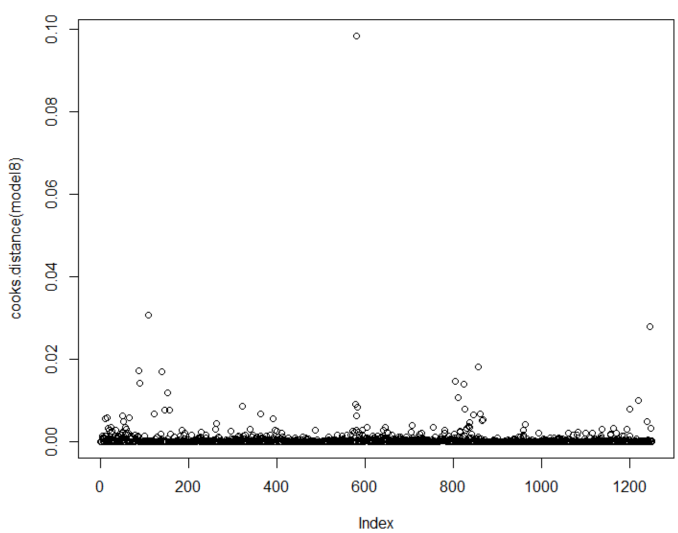



The first graph shows that the predicted values of gfr is following the line y=x, which means the model fits the data quite well. The second one shows that points lie symmetrically above and below the 0 horizontal line. And the variance is not constant.




The fitted values approximately follows the normal distribution since the curve mostly follow the line within (-2, 2).




This graph shows that there is no influential point.

Part B

1.

c. The p-value should be above 0.05

e. The proportion of times the p-value is less than 0.05 is 0.05 which matchs with my intuition. The slope could not be significant because they are from different distribution. Type I error is simulated in my simulation.

f. The proportion of times the p-value is less than 0.05 is 1. y is simulated given x so that the slope should be smaller than 0.05. 1 - Type II error is simulated here.

2.

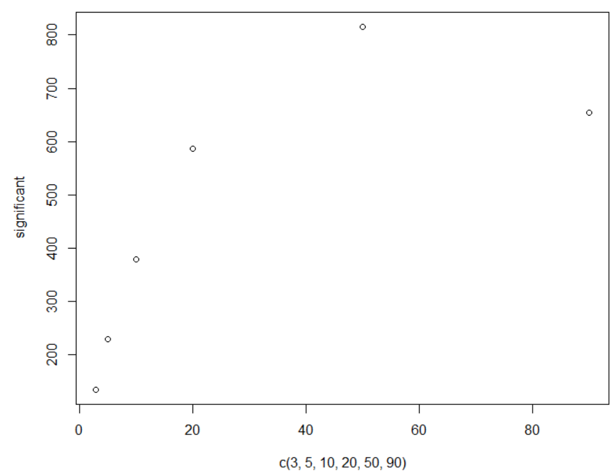
c. There are 51, 51, 55 p-values for each variable are significant at the 0.05 level. Each variable has approximately 0.05 probability to be significant.

d. There are 146 minimum p-values for each regression are significant at the 0.05 level. Strictly speaking, these three variables are uncorrelated based on the assumption. Each variable have 0.05 probability to be significant according to question c. The problem now is there is at least one p-value for each regression will be less than 0.05, $1-(0.95)^3=0.143$. This is what we simulate with R.

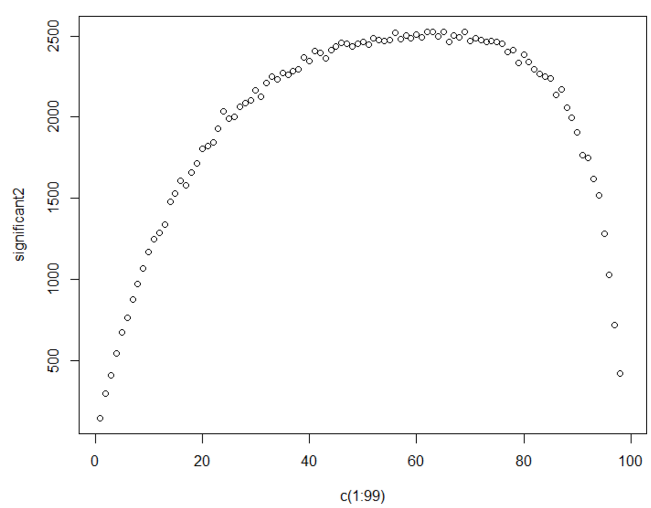
e. We find the number of times finding at least one significant variable for each P will go down at about 50 predictors. Before this, the curve perform well with

$$\frac{\#\ of\ times\ finding\ at\ least\ one\ significant\ variable}{\#\ of\ simulation\ times}=1-(0.95)^n$$

There are few points to show us the exact trend of this kind of simulation. And the curve should be approximately closed to 1.




f. It shows obvious pattern in the graph below with 3000 simulation and predictors from 1 to 99. The trend sudden decreases into 0. It may be due to the overfit or some problem in the simulation design.




Appendix
# Homework 2
setwd("D:/Google Drive/Fall2017/PHP2550/HW2")
baseseg <- read.csv("baseseg.csv")
potential_predictors <- c("bascre", "sbase", "dbase", "baseu", "AGE", "SEX", "black")
#eliminate NAs in gfr
baseseg <- baseseg[-which(is.na(baseseg$gfr)),]

```
# extract the significant predictors which have p-values < 0.05
potentials <- c()
for (i in potential_predictors){
  if (summary(lm(paste0("baseseg$gfr~baseseg$",i)))[[4]][8] < 0.05){
    potentials <- c(potentials, i)
  }
}
potentials
par(mfrow=c(2,3))
plot(baseseg$bascre, baseseg$gfr)
plot(baseseg$sbase, baseseg$gfr)
plot(baseseg$dbase, baseseg$gfr)
plot(baseseg$baseu, baseseg$gfr)
plot(baseseg$AGE, baseseg$gfr)
# fit all the potentials
model1 <- lm(gfr~bascre+sbase+dbase+baseu+AGE+I(bascre^2)+I(bascre^3)+I(bascre^4),data =
baseseg)
summary(model1, cor=T) # sbase and dbase has negative pairwise correlation
library(leaps)
leaps <-
regsubsets(gfr~bascre+sbase+dbase+baseu+AGE+I(bascre^2)+I(bascre^3)+I(bascre^4),data =
baseseg,nbest = 4)
plot(leaps,scale = "adjr2")
# drop sbase
leaps2 <- regsubsets(gfr~bascre*sbase*dbase*baseu*AGE*I(bascre^2)+I(bascre^3),data =
baseseg,nbest = 1)
plot(leaps2,scale = "adjr2")
leaps2$xnames[c(1,2,3,4,6,7,9,12)]

leaps3 <- regsubsets(gfr~bascre*dbase*baseu*I(bascre^2)+I(bascre^3),data = baseseg,nbest = 1)
plot(leaps3,scale = "adjr2")
# drop AGE
model22 <- lm(gfr~bascre*dbase*baseu-1,data = baseseg)
summary(model22)
plot(fitted(model22), residuals(model22),xlab = "Fitted", ylab = "Residuals")
abline(h=0)


model2 <- lm(gfr~-1+(bascre+dbase+baseu)^3+I(bascre^2)+I(bascre^3),data = baseseg)
summary(model2) # The coefficients themselves do not change residual SE does not change
```

```
# constant variance
plot(fitted(model2), residuals(model2),xlab = "Fitted", ylab = "Residuals")
abline(h=0) # Nonlinear which indicates some change in the model is necessary


full_model <- lm(gfr~bascre*dbase*baseu*I(bascre^2)*I(bascre^3)*I(bascre^4),data = baseseg)
reduced_model <- step(full_model,direction = "backward")

m1 <- lm(gfr~bascre,data = baseseg) # nonlinearity
plot(fitted(m1), residuals(m1),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
m2 <- lm(gfr~dbase,data = baseseg)
plot(fitted(m2), residuals(m2),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
m3 <- lm(gfr~baseu,data = baseseg)
plot(fitted(m3), residuals(m3),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
# Look like Nike function, add reciprocal term (respect the hierarchy)

m22 <- lm(gfr~bascre+I(bascre^(2)), data = baseseg)
plot(fitted(m22), residuals(m22),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
summary(m22)

m23 <- lm(gfr~bascre+I(bascre^2)+I(bascre^3), data = baseseg)
summary(m23)
plot(fitted(m23), residuals(m23),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
AIC(m22)
AIC(m23)
AIC(m24)
m24 <- lm(gfr~bascre+I(bascre^2)+I(bascre^3)+I(bascre^4), data = baseseg)
summary(m24)
plot(fitted(m24), residuals(m24),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
m25 <- lm(gfr~bascre+I(bascre^2)+I(bascre^3)+I(bascre^4)+I(bascre^5), data = baseseg)
summary(m25)
#add polynomial term of order 4
model3 <- lm(gfr~-1+(dbase+baseu+bascre)^3+I(bascre^-1)+I(bascre^2)+I(bascre^3),data = 
baseseg)
summary(model3)
```

```
plot(fitted(model3), residuals(model3),xlab = "Fitted", ylab = "Residuals")
abline(h=0)

model4 <- lm(gfr~-1+(dbase+baseu+bascre)^3+I(bascre^-1)+I(bascre^-2)+I(bascre^2)+I(bascre^3),data = baseseg)
summary(model4)
plot(fitted(model4), residuals(model4),xlab = "Fitted", ylab = "Residuals")
abline(h=0)

model5 <- lm(gfr~-1+(dbase+baseu+bascre)+I(bascre^-1)+I(bascre^-2)+I(bascre^2)+I(bascre^3),data = baseseg)
summary(model5)
plot(fitted(model5), residuals(model5),xlab = "Fitted", ylab = "Residuals")
abline(h=0)

model6 <- lm(gfr~-1+(bascre+baseu)+I(bascre^-1)+I(bascre^-2)+I(bascre^2),data = baseseg)
summary(model6)
plot(fitted(model6), residuals(model6),xlab = "Fitted", ylab = "Residuals")
abline(h=0)

plot(model6)

model8 <- lm(gfr~-1+(bascre*baseu)+I(bascre^-1)+I(bascre^-2)+I(bascre^2),data = baseseg)
summary(model8)

plot(fitted(model8), residuals(model8),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
######
model7 <- lm(gfr~bascre+dbase+I(bascre^2)+I(bascre^3)+bascre:dbase+dbase:I(bascre^2)+bascre:I(bascre^3)+

dbase:I(bascre^3)+I(bascre^2):I(bascre^3)+bascre:dbase:I(bascre^3)+bascre:I(bascre^2):I(bascre^3)+

dbase:I(bascre^2):I(bascre^3)+bascre:dbase:I(bascre^2):I(bascre^3)+I(bascre^-1),data = baseseg)
summary(model7)

plot(fitted(model7), residuals(model7),xlab = "Fitted", ylab = "Residuals")
abline(h=0)
# Normality
```

```
qqnorm(residuals(model5), ylab = "Residuals")
qqline(residuals(model5))

qqnorm(residuals(model8), ylab = "Residuals")
qqline(residuals(model8))
shapiro.test(residuals(model8))
#outlier
library(car)
outlierTest(model8)

plot(cooks.distance(model8))
# PART B
# 1
y <- rnorm(100, mean = 10, sd=2)
x <- rnorm(100, mean = 3, sd=1)
model_B <- lm(y~x)
summary(model_B)[[4]][2,4]
p_values <- c()
for (i in c(1:1000)){
   y <- rnorm(100, mean = 10, sd=2)
   x <- rnorm(100, mean = 3, sd=1)
   model_B1 <- lm(y~x)
   p_values <- c(p_values,summary(model_B1)[[4]][2,4])
}
sum(p_values<0.05)/1000

#
p_values2 <- c()
for (i in 1:1000){
   y <- rep(0, 100)
   x <- rnorm(100,mean = 3,sd=1)
   for (j in 1:100){
      y[j] <- rnorm(1, x[j]+10,1)
   }
   model_B2 <- lm(y~x)
   p_values2 <- c(p_values2, summary(model_B2)[[4]][2,4])

}
sum(p_values2<0.05)/1000
```

```
#2
library(MASS)
y <- rnorm(100,10,4)
X <- mvrnorm(n=100,c(1,2,3),diag(3))
set.seed(4)
p_values3 <- c()
for (i in 1:1000){
  y <- rnorm(100,10,4)
  X <- mvrnorm(n=100,c(1,2,3),diag(3))
  model_B3 <- lm(y~X)
  p_values3 <- rbind(p_values3, summary(model_B3)$coefficients[-1,4])
}
sum(p_values3[,1] <0.05)
sum(p_values3[,2] <0.05)
sum(p_values3[,3] <0.05)

sum(apply(p_values3,1,min)<0.05)


significant <- c()
for (n in c(3,5,10,20,50,90)){
  p_values4 <- c()
  for (i in 1:1000){
    y <- rnorm(100,10,4)
    X <- mvrnorm(n=100,c(1:n),diag(n))
    model_B4 <- lm(y~X)
    p_values4 <- rbind(p_values4, summary(model_B4)$coefficients[-1,4])
  }
  significant <- c(significant, sum(apply(p_values4,1,min)<0.05))
}

plot(x=c(3,5,10,20,50,90), y=significant)



significant2 <- c()
for (n in c(1:200)){
  p_values5 <- c()
  for (i in 1:2000){
    y <- rnorm(100,10,4)
    X <- mvrnorm(n=100,c(1:n),diag(n))
```

```
        model_B5 <- lm(y~X)
        p_values5 <- rbind(p_values5, summary(model_B5)$coefficients[-1,4])
    }
    significant2 <- c(significant2, sum(apply(p_values5,1,min)<0.05))
}

plot(x=c(1:99), y=significant2)
```