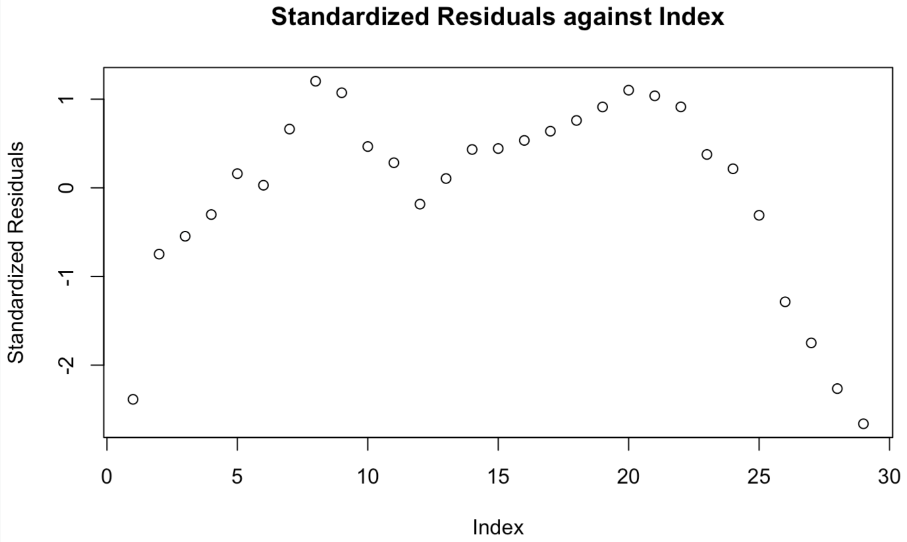
MATH3424 HW4
Name: Leung Ko Tsun
SID: 20516287

1a

```
> #Q1a
> q1data <- read.table("./Downloads/MATH3424HW4Data/Crude-Oil-Production.txt", header = TRUE, s
ep = ",")
> q1data$logBarrels <- log(q1data$Barrels)
> q1model <- lm(q1data$logBarrels ~ q1data$Year, data = q1data)
> plot(1:29, rstudent(q1model), xlab = "Index" , ylab = "Standardized Residuals", main = "Stand
ardized Residuals against Index")
```



The residual plot shows a strong trend of M shape, so the errors should be autocorrelated.

1b

```
> #Q1b
> library(lmtest)
> dwtest(q1model)

	Durbin-Watson test

data:  q1model
DW = 0.19454, p-value = 3.503e-14
alternative hypothesis: true autocorrelation is greater than 0
```

Since d = 0.19454 is close to 0 and p-value is 3.503e-14 which is smaller than 0.05, so it shows as an evidence of autocorrelation.

1c

```
> #Q1c
> count <- 1
> for (i in c(2:29)) {
+   if (q1model$residuals[i] * q1model$residuals[i-1] < 0) {
+     count <- count + 1
+   }
+ }
> count
[1] 5
> q1c1 <- length(which(q1model$residuals > 0))
> q1c2 <- length(which(q1model$residuals < 0))
> q1cmu <- 2 * q1c1 * q1c2 / (q1c1 + q1c2) + 1
> q1cstd <- sqrt(2 * q1c1 * q1c2 * (2 * q1c1 * q1c2 - q1c1 - q1c2) / (q1c1 + q1c2)^2 / (q1c1 + q1c2 - 1))
> q1cZscore <- (count - q1cmu) / q1cstd
> q1cZscore
[1] -3.825054
> pnorm(q1cZscore)
[1] 6.537168e-05
> |
```

Since p-value = 6.537e-5 which is smaller than 0.05, we can claim that there exist autocorrelation

1d

```
> #Q1d
> hatrho <- sum(q1model$residuals[-1] * q1model$residuals[-29]) / sum(q1model$residuals^2)
> hatrho
[1] 0.7337842
> q1ddata <- data.frame(y = log(q1data$Barrels)[-1] - log(q1data$Barrels)[-29] * hatrho, x <- q1data$Year[-1] - q1data$Year[-29] * hatrho)
> q1dmodel <- lm(y~x, data = q1ddata)
> dwtest(q1dmodel)

	Durbin-Watson test

data:  q1dmodel
DW = 0.80175, p-value = 6.909e-05
alternative hypothesis: true autocorrelation is greater than 0
```

After doing Cochrane and Orcutt procedure for 1 iteration, the statistic d increases from 0.7338 to 0.80175, but the p-value is 6.909e-5, which is still smaller than 0.05, so that autocorrelation still exists.

2a

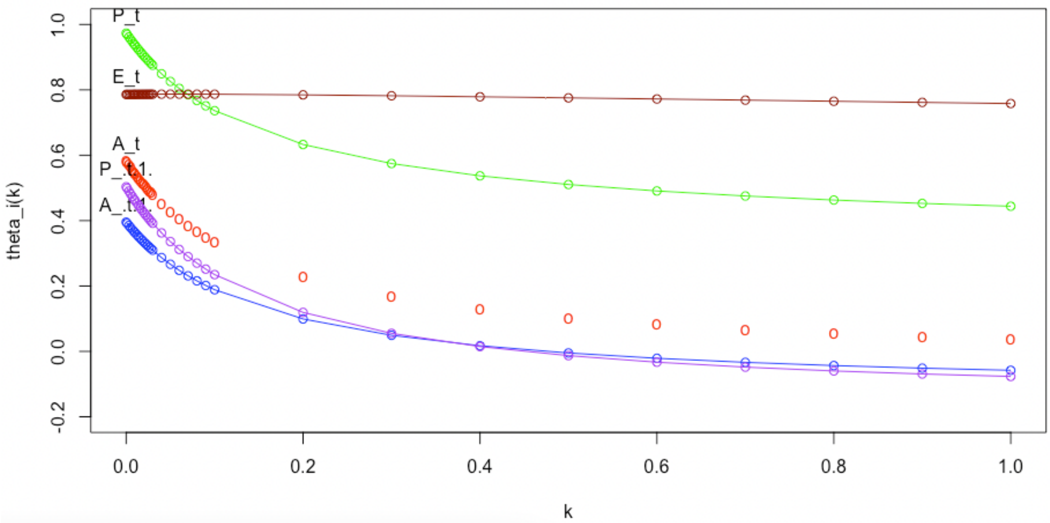
```
> #Q2
> library(regclass)
> q2data <- read.table("./Downloads/MATH3424HW4Data/Advertising.txt", header = TRUE)
> q2ad <- q2data[-1,]
> rownames(q2ad) <- 1:nrow(q2ad)
> q2ad$S_.t.1 <- q2data$S_t[-22]
> q2data <- q2ad
> q2model1 <- lm(q2data$S_t ~ q2data$E_t + q2data$A_t + q2data$P_t + q2data$A_.t.1., data = q2data)
> VIF(q2model1)
    q2data$E_t      q2data$A_t      q2data$P_t q2data$A_.t.1.
      1.034196        1.316052        1.431257        1.282855
> q2model2 <- lm(q2data$S_t ~ q2data$E_t + q2data$A_t + q2data$P_t + q2data$S_.t.1, data = q2data)
> VIF(q2model2)
   q2data$E_t    q2data$A_t    q2data$P_t q2data$S_.t.1
     5.440452      2.239380      2.035060      6.644790
> q2model3 <- lm(q2data$S_t ~ q2data$E_t + q2data$A_t + q2data$A_.t.1. + q2data$S_.t.1, data = q2data)
> VIF(q2model3)
    q2data$E_t      q2data$A_t q2data$A_.t.1.  q2data$S_.t.1
      3.415835        1.320897        1.051241        3.829540
> q2model4 <- lm(q2data$S_t ~ q2data$E_t + q2data$P_t + q2data$A_.t.1. + q2data$S_.t.1, data = q2data)
> VIF(q2model4)
    q2data$E_t      q2data$P_t q2data$A_.t.1.  q2data$S_.t.1
      3.455814        1.309504        1.146808        3.490918
```

All models have VIF_j less than 10 which means collinearity has been removed in all models.

3.

a).

```
> #### Q3a
> normalize <- function(c) {
+   return ((c - mean(c)) / sd(c))
+ }
> q3data <- apply(q2data, 2, normalize)
> q3y <- q3data[,1]
> q3x <- q3data[,-1]
>
> q3_vec1 <- c(0.000, 0.001, 0.003, 0.005, 0.007, 0.009)
> q3_vec2<- seq(from=0.01, to=0.03, by=0.002)
> q3_vec3 <- seq(from=0.04, to=0.09, by=0.01)
> q3_vec4 <- seq(from=0.1, to=1, by=0.1)
> q3_vec <- c(q3_vec1, q3_vec2, q3_vec3, q3_vec4)
>
>
> q3_rec <- matrix(data=NA, nrow=length(q3_vec), ncol=5)
> for (i in (1:length(q3_vec)))
+ {
+   q3k <- q3_vec[i]
+   q3_theta <- solve(t(q3x) %*% q3x + q3k*diag(5)) %*% t(q3x) %*% q3y
+   q3_rec[i,] <- q3_theta
+ }
> offset <- 0.05
> colours = c("red", "green", "dark red", "blue", "purple")
> legends = c("A_t", "P_t", "E_t", "A_.t.1.", "P_.t.1.")
> plot(q3_vec, q3_rec[,1], col=colours[1], pch="o", xlab="k", ylab="theta_i(k)", ylim=c(-0.2,1))
> text(q3_vec[1], q3_rec[1,1]+offset, legends[1])
> for (i in (2:5))
+ {
+   points(q3_vec, q3_rec[,i], col=colours[i], lty=1)
+   lines(q3_vec, q3_rec[,i], col=colours[i], lty=1)
+   text(q3_vec[1], q3_rec[1,i]+offset, legends[i])
+ }
```



All variables becomes more stable at k = 0.65

```
> #### Q3b
> q3bY <- as.matrix(q3data[,1])
> q3bX <- as.matrix(q3data[,-1])
> theta <- function(i) {
+   return (solve(t(q3bX) %*% q3bX + i * diag(5)) %*% t
(q3bX) %*% q3bY)
+ }
> q3model <- lm(q3bY ~ ., data = as.data.frame(q3bX))
> q3b_rhosq <- sum((q3model$residuals)^2)/ 16
> q3b_k <- c()
> q3b_k_1 <- 0
> for (i in c(0:5)) {
+   k <- 5 * q3b_rhosq / sum((theta(q3b_k_1)) ^ 2)
+   q3b_k_1 <- k
+   q3b_k <- c(q3b_k, k)
+ }
> q3b_k
[1] 0.2356103 0.5242140 0.6304654 0.6505368
[5] 0.6538152 0.6543381
```

As from the above R result, k1 = 0.2356, k2 = 0.5242, k3 = 0.6304, k4 = 0.6505, so it converges to k = 0.65 after 5 iterations

OLS Result:

```
> q3b_k[6]
[1] 0.6543381
> q3_sd_all <- apply(q2data[,-1], 2, sd)
> q3_mean_all <- apply(q2data[,-1], 2, mean)
> beta_1_to_j <- theta(q3b_k[6])*sd(q2data$S_t)/q3_sd_all
> beta_0 <- mean(q2data$S_t) - sum(q3_mean_all*beta_1_to_j)
> beta_original <- rbind(beta_0, beta_1_to_j)
> beta_original
              [,1]
beta_0   8.0123932
A_t      0.6773945
P_t      4.1452662
E_t     22.0707010
A_.t.1. -0.2751787
P_.t.1. -0.3419280
```

```
> summary(lm(S_t ~ ., data=q2data))

Call:
lm(formula = S_t ~ ., data = q2data)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.8601 -0.9848  0.1323  0.7017  2.2046 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -14.194     18.715  -0.758   0.4592    
A_t            5.361      4.028   1.331   0.2019    
P_t            8.372      3.586   2.334   0.0329 *  
E_t           22.521      2.142  10.512 1.36e-08 ***
A_.t.1.        3.855      3.578   1.077   0.2973    
P_.t.1.        4.125      3.895   1.059   0.3053    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.32 on 16 degrees of freedom
Multiple R-squared:  0.9169,	Adjusted R-squared:  0.8909 
F-statistic:  35.3 on 5 and 16 DF,  p-value: 4.289e-08
```

4.

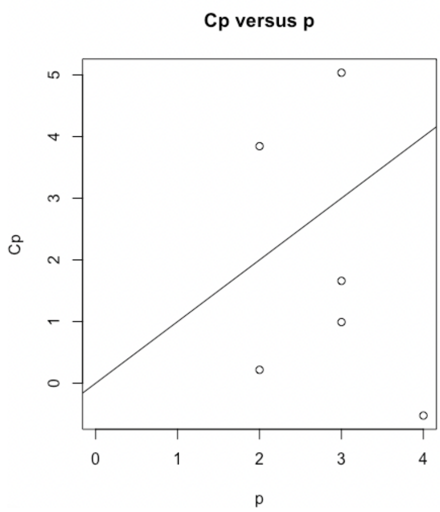
```
> ###q4a
> q4data <- read.table("./Downloads/MATH3424HW4Data/Gas
oline-Consumption.txt", header= TRUE)
> q4model <- lm(Y ~ . ,data = q4data)
> summary(q4model)

Call:
lm(formula = Y ~ ., data = q4data)

Residuals:
    Min      1Q  Median      3Q     Max 
-5.3498 -1.6236 -0.6002  1.5155  5.2815 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 17.773204  30.508775   0.583   0.5674
X_1         -0.077946   0.058607  -1.330   0.2001
X_2         -0.073399   0.088924  -0.825   0.4199
X_3          0.121115   0.091353   1.326   0.2015
X_4          1.329034   3.099535   0.429   0.6732
X_5          5.975989   3.158647   1.892   0.0747
X_6          0.304178   1.289094   0.236   0.8161
X_7         -3.198576   3.105435  -1.030   0.3167
X_8          0.185362   0.129252   1.434   0.1687
X_9         -0.399146   0.323812  -1.233   0.2336
X_.10.      -0.005193   0.005893  -0.881   0.3898
X_.11.       0.598655   3.020681   0.198   0.8451
```

The p-value of every predictor is larger than 0.05. So we should not include all variables

4b.

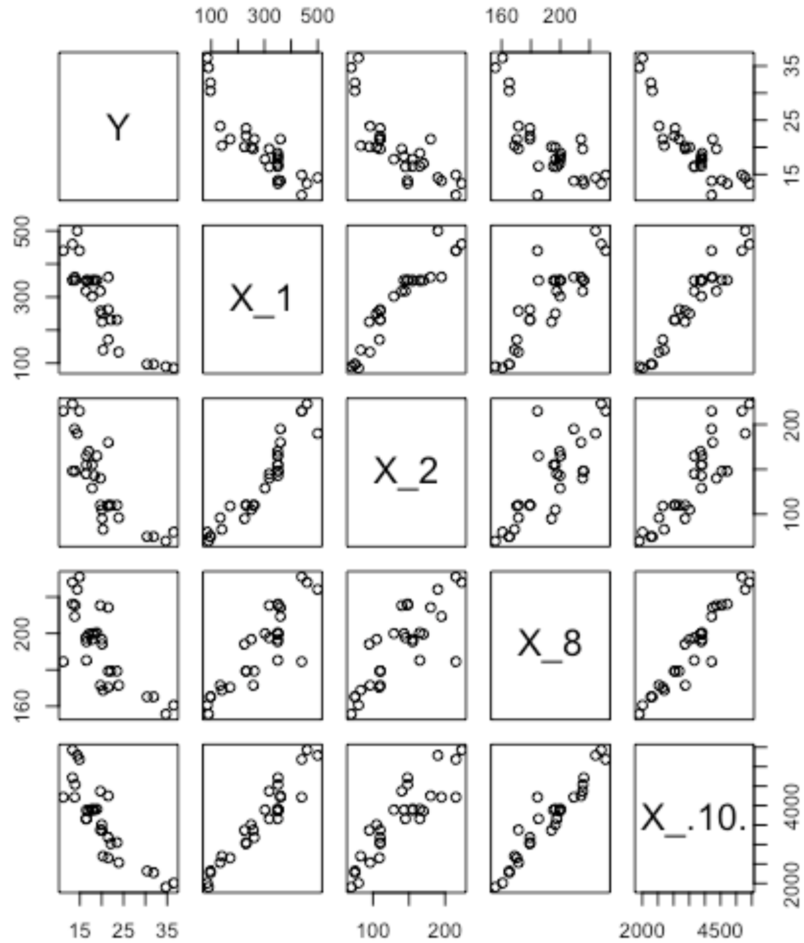
```
>
> q4model1 <- lm(Y~q4data$X_1, data = q4data)
> q4model2 <- lm(Y~q4data$X_.10., data = q4data)
> q4model3 <- lm(Y ~ q4data$X_1 + q4data$X_.10., data = q4data)
> q4model4 <- lm(Y ~ q4data$X_2 + q4data$X_.10., data = q4data)
> q4model5 <- lm(Y ~ q4data$X_8 + q4data$X_.10., data = q4data)
> q4model6 <- lm(Y ~ q4data$X_8 + q4data$X_5 + q4data$X_.10., data = q4data)
> crit <- as.data.frame(matrix(data = NA, 4, 6), row.names=c("adj R^2", "Mallow's Cp", "AIC", "BIC"))
> crit[1,] <- c(summary(q4model1)$adj.r.squared,
+               summary(q4model2)$adj.r.squared,
+               summary(q4model3)$adj.r.squared,
+               summary(q4model4)$adj.r.squared,
+               summary(q4model5)$adj.r.squared,
+               summary(q4model6)$adj.r.squared)
> crit[2,] <- c(ols_mallows_cp(q4model1, q4model),
+               ols_mallows_cp(q4model2, q4model),
+               ols_mallows_cp(q4model3, q4model),
+               ols_mallows_cp(q4model4, q4model),
+               ols_mallows_cp(q4model5, q4model),
+               ols_mallows_cp(q4model6, q4model))
> crit[3,] <- c(AIC(q4model1),
+               AIC(q4model2),
+               AIC(q4model3),
+               AIC(q4model4),
+               AIC(q4model5),
+               AIC(q4model6))
> crit[4,] <- c(BIC(q4model1),
+               BIC(q4model2),
+               BIC(q4model3),
+               BIC(q4model4),
+               BIC(q4model5),
+               BIC(q4model6))
> colnames(crit) <- c("modelA", "modelB", "modelC", "modelD", "modelE", "modelF")
> crit
               modelA      modelB      modelC      modelD      modelE      modelF
adj R^2      0.751499   0.7171223   0.7477775   0.7145943   0.7543429   0.7807823
Mallow's Cp  0.217510   3.8443469   1.6597812   5.0356539   0.9918459  -0.5239606
AIC        157.374293 161.2613314 158.7292132 162.4372038 157.9379565 155.3896042
BIC        161.577886 165.4649235 164.3340028 168.0419934 163.5427461 162.3955911
```

From the above R result, we will select modelF. If we see adjusted R squared or AIC, modelF has the largest adjusted R square, smallest AIC, close distance in Cp = p, and the second smallest BIC.



4c).

pairs(q4data[,c(1,2,3,9,11)])



From the pairwise scatterplots, it indicated that there is a strong linear relationship between Y and X1, X2, X8, X10. Therefore, it suggests that there may be linear relationships between Y and the 11 predictors.

4d

Step1 (X1 is selected):

```
> ###q4d - X1 is selected
> q4dX1 <-  q4data[,c(2,3,6,9,11)]
> q4dSelected1 <- colnames(q4dX1)[1]
> for( x in colnames(q4dX1)[-1]) {
+   if(abs(cor(q4data$Y, q4data[x])) > abs(cor(q4data$Y, q4data[q4dSel
ected1]))) {
+     q4dSelected1 <- x
+   }
+ }
> q4dmodel1 <- lm(as.formula(paste("Y~", q4dSelected1, sep="")), data
 = q4data)
> summary(q4dmodel1)

Call:
lm(formula = as.formula(paste("Y~", q4dSelected1, sep = "")),
    data = q4data)

Residuals:
    Min      1Q  Median      3Q     Max 
-6.6000 -2.0240 -0.2681  1.4684  7.0261 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 33.487803   1.537109  21.786  < 2e-16 ***
X_1         -0.047056   0.004996  -9.418 3.55e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.122 on 28 degrees of freedom
Multiple R-squared:  0.7601,	Adjusted R-squared:  0.7515 
F-statistic:  88.7 on 1 and 28 DF,  p-value: 3.555e-10
```

The model is significant.

Step 2: X5 is selected

```
> ###q4d - X5 is selected
> q4dX2 <-  q4data[,c(3,6,9,11)]
> q4dSelected2 <- colnames(q4dX2)[1]
> for(x in colnames(q4dX2)[-1]) {
+   if(abs(cor(q4dmodel1$residuals, q4data[x])) > abs(cor(q4dmodel1$re
siduals, q4data[q4dSelected2]))) {
+     q4dSelected2 <- x
+   }
+ }
> q4dmodel2 <- lm(Y~X_1+X_5, data = q4data)
> q4dmodel2 <- lm(as.formula(paste("Y~X_1+", q4dSelected2, sep="")), d
ata = q4data)
> summary(q4dmodel2)

Call:
lm(formula = as.formula(paste("Y~X_1+", q4dSelected2, sep = "")),
    data = q4data)

Residuals:
    Min      1Q  Median      3Q     Max
-6.4870 -1.8186  0.2525  1.5600  6.7924

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 29.259779   6.068399   4.822 4.92e-05 ***
X_1         -0.043765   0.006801  -6.435 6.77e-07 ***
X_5          1.074811   1.491453   0.721    0.477
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.149 on 27 degrees of freedom
Multiple R-squared:  0.7646,    Adjusted R-squared:  0.7472
F-statistic: 43.85 on 2 and 27 DF,  p-value: 3.307e-09
```
From the result above, p-value of X5 is 0.477 > 0.05, and t-value is also not significant, so we should not include X5. Final model is Y = X_1 + epsilon