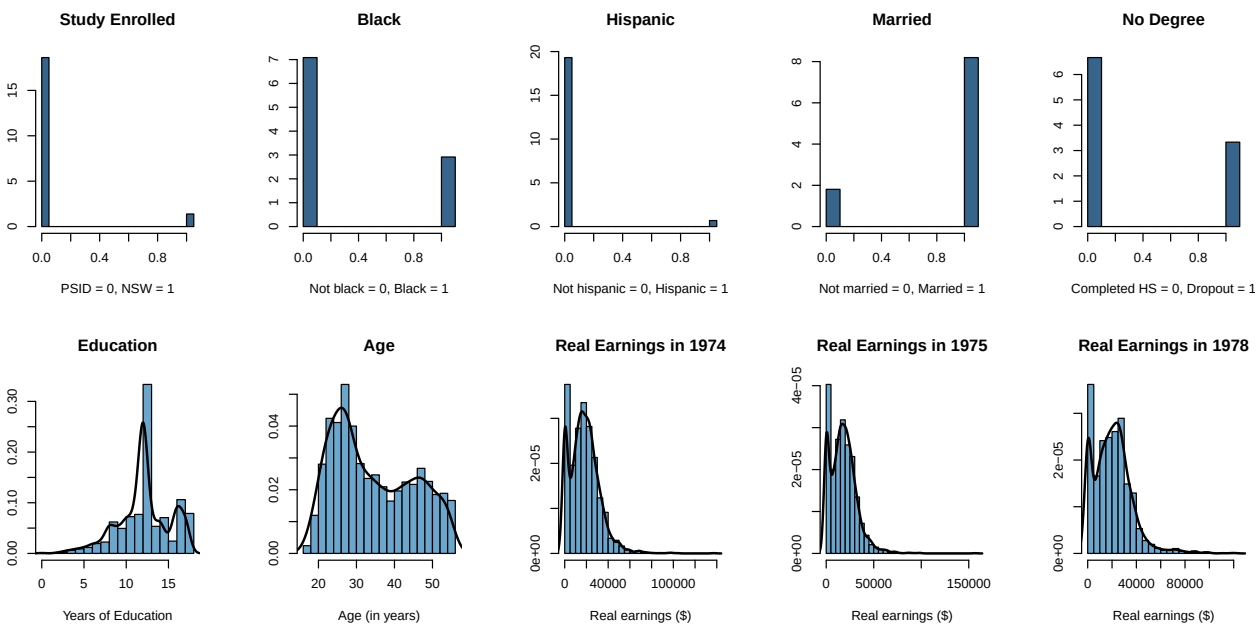
Econ 144: Homework #1 Solution

Problem 1:

1a. Plot a histogram of each variable.

There are a total of 10 variables; 5 are binary, 5 are not. For the ones that are not binary, we find that there is a large degree of skewness in all five of the variables. The estimated density functions are also provided for reference.



1b. Estimate the full regression model.

There are several variables that do not appear to have any statistical significance in estimating real earnings in 1978. Namely, the study enrolled (*trt*), whether or not the individual was black or not (*black*), and whether or not the individual was a high school graduate or not, are not statistically significant in the regression model at a 5% significance level.

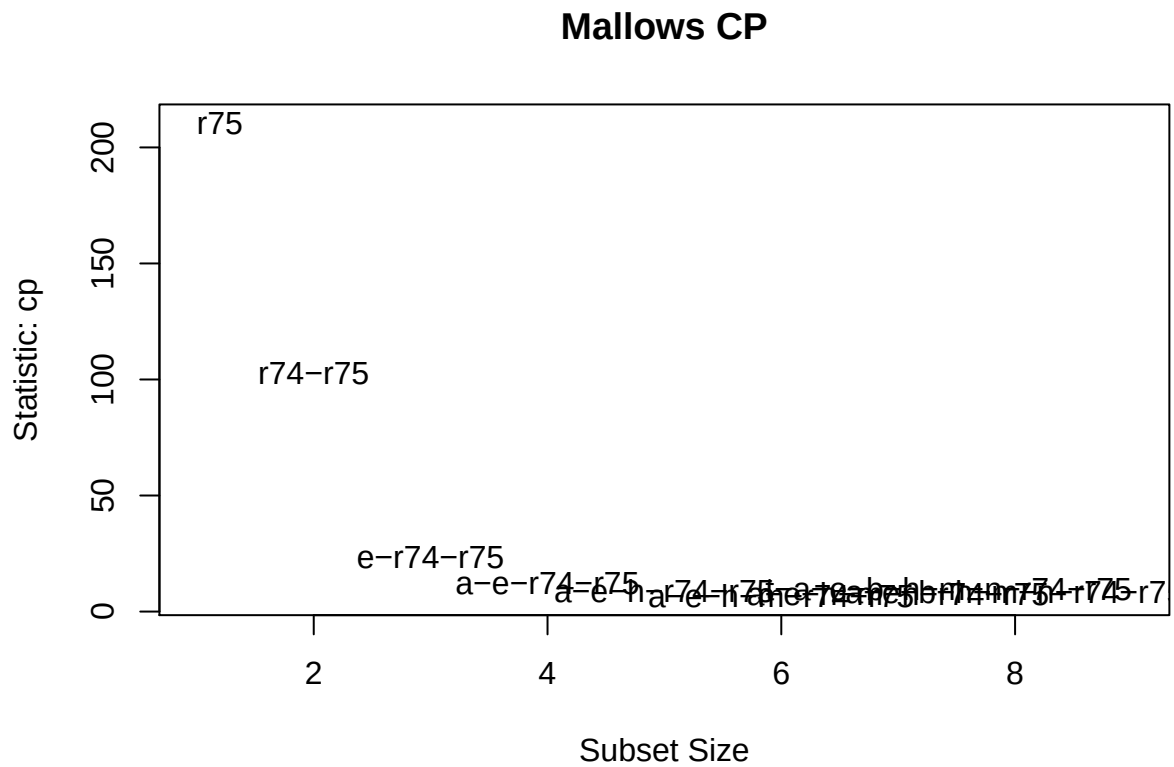
```
full_model <- lm(re78 ~ ., data = nsw74psid1)
summary(full_model)
```

```
##
## Call:
## lm(formula = re78 ~ ., data = nsw74psid1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -64870  -4302   -435   3786 110412
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -129.74276 1688.51706  -0.077   0.9388
## trt          751.94643  915.25723   0.822   0.4114
## age          -83.56559   20.81380  -4.015 6.11e-05 ***
## educ         592.61020  103.30278   5.737 1.07e-08 ***
## black       -570.92797  495.17772  -1.153   0.2490
## hisp        2163.28118 1092.29036   1.981   0.0478 *
## marr        1240.51952  586.25391   2.116   0.0344 *
## nodeg        590.46695  646.78417   0.913   0.3614
## re74           0.27812    0.02792   9.960  < 2e-16 ***
## re75           0.56809    0.02756  20.613  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10070 on 2665 degrees of freedom
## Multiple R-squared:  0.5864, Adjusted R-squared:  0.585
## F-statistic: 419.8 on 9 and 2665 DF,  p-value: < 2.2e-16
```

1c. Mallows C_p.

```r
reg.cp = regsubsets(re78 ~ ., method = c("exhaustive"), nbest = 1,
    data = nsw74psid1, nvmax = 9)
subsets(reg.cp, statistic = "cp", legend = F, main = "Mallows CP")
```




```
##       Abbreviation
## trt              t
## age              a
## educ             e
## black            b
## hisp             h
## marr             m
## nodeg            n
## re74           r74
## re75           r75
```

We compute Mallows C_p for the various subsets of variables and find the combination of variables that will yield the minimum value. There exists several ways to interpret the following plot. We can look at the absolute minima that Mallows C_p achieves and determine which variables are included in the minimum. In this case, Mallows C_p achieves a minimum with 6 variables. However, a close look at the plot above shows that using 5 variables may yield comparable performance.

We look at the 6 variables that, according to Mallows C_p should be included:

```r
summary.cp <- summary(reg.cp)
summary.cp$which[which.min(summary.cp$cp), ]  #6 variables
```

```
## (Intercept)         trt         age        educ       black        hisp
##        TRUE       FALSE        TRUE        TRUE       FALSE        TRUE
##        marr       nodeg        re74        re75
##        TRUE       FALSE        TRUE        TRUE
```

```r
# AGE, EDUC, HISP, MARR, RE74, RE75
```

For simplicity, we stick with 6 variables and are left with the following equation:

$$re78 = \alpha + \beta_1 age + \beta_2 educ + \beta_3 hisp + \beta_4 marr + \beta_5 re74 + \beta 6 re75 + \varepsilon$$

```r
reduc_model <- lm(re78 ~ age + educ + hisp + marr + re74 + re75,
    data = nsw74psid1)
summary(reduc_model)
```

```
##
## Call:
## lm(formula = re78 ~ age + educ + hisp + marr + re74 + re75, data = nsw74psid1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -64914  -4316   -498   3722 110303
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  517.84209 1182.98921   0.438   0.6616
## age          -81.42078   20.55047  -3.962 7.63e-05 ***
## educ         547.71773   71.70329   7.639 3.04e-14 ***
## hisp        2405.49742 1074.08566   2.240   0.0252 *
## marr        1124.59319  541.07870   2.078   0.0378 *
## re74           0.27729    0.02789   9.941  < 2e-16 ***
## re75           0.56817    0.02747  20.681  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10070 on 2668 degrees of freedom
## Multiple R-squared:  0.586,  Adjusted R-squared:  0.585
## F-statistic: 629.4 on 6 and 2668 DF,  p-value: < 2.2e-16
```

Computing AIC and BIC on both the full model and reduced model converge to show that the reduce model is preferred.

```r
AIC(full_model, reduc_model)
```

```
##             df      AIC
## full_model  11 56916.33
## reduc_model  8 56912.93
```

```r
BIC(full_model, reduc_model)
```

```
##             df      BIC
## full_model  11 56981.14
## reduc_model  8 56960.06
```

A quick RESET test yields that the model is misspecified. However, the inclusion of higher order terms shows that the model continues to be misspecified, which may hint at the fact that we are

missing some key variables that may help us estimate our dependent variable more. For simplicity, we continue with our analysis using model proposed above.

```
resettest(reduc_model, power = 2)
```

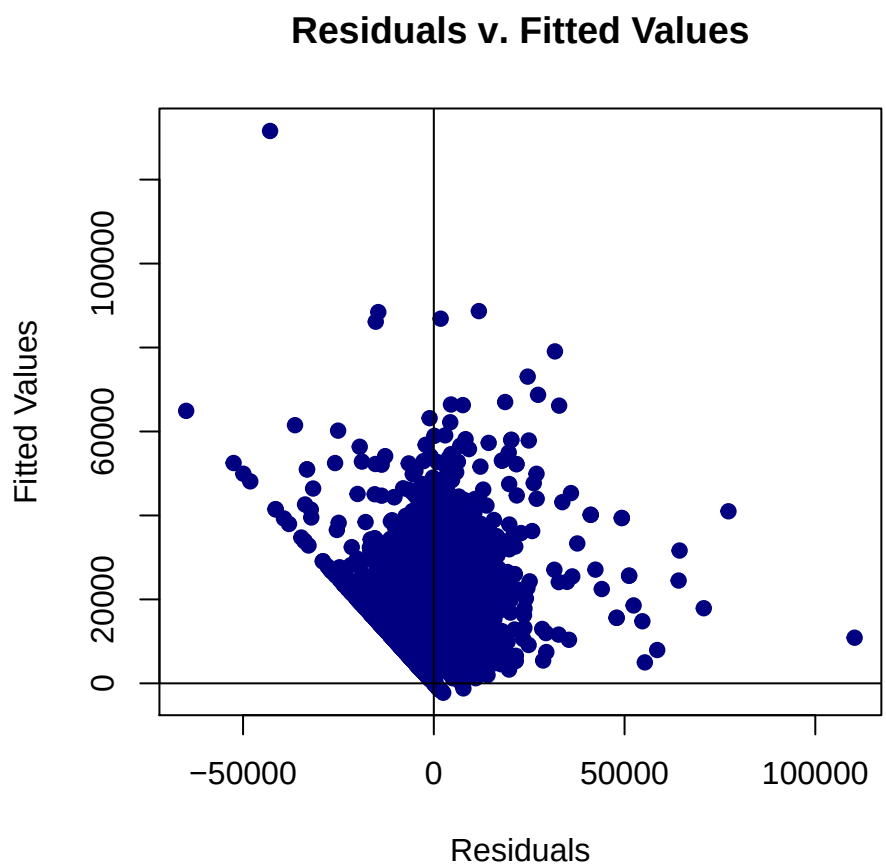
```
##
##  RESET test
##
## data:  reduc_model
## RESET = 5.7841, df1 = 1, df2 = 2667, p-value = 0.01624
```

```
resettest(reduc_model, power = 3)
```

```
##
##  RESET test
##
## data:  reduc_model
## RESET = 13.353, df1 = 1, df2 = 2667, p-value = 0.0002629
```

1d. Residuals v. Fitted Values

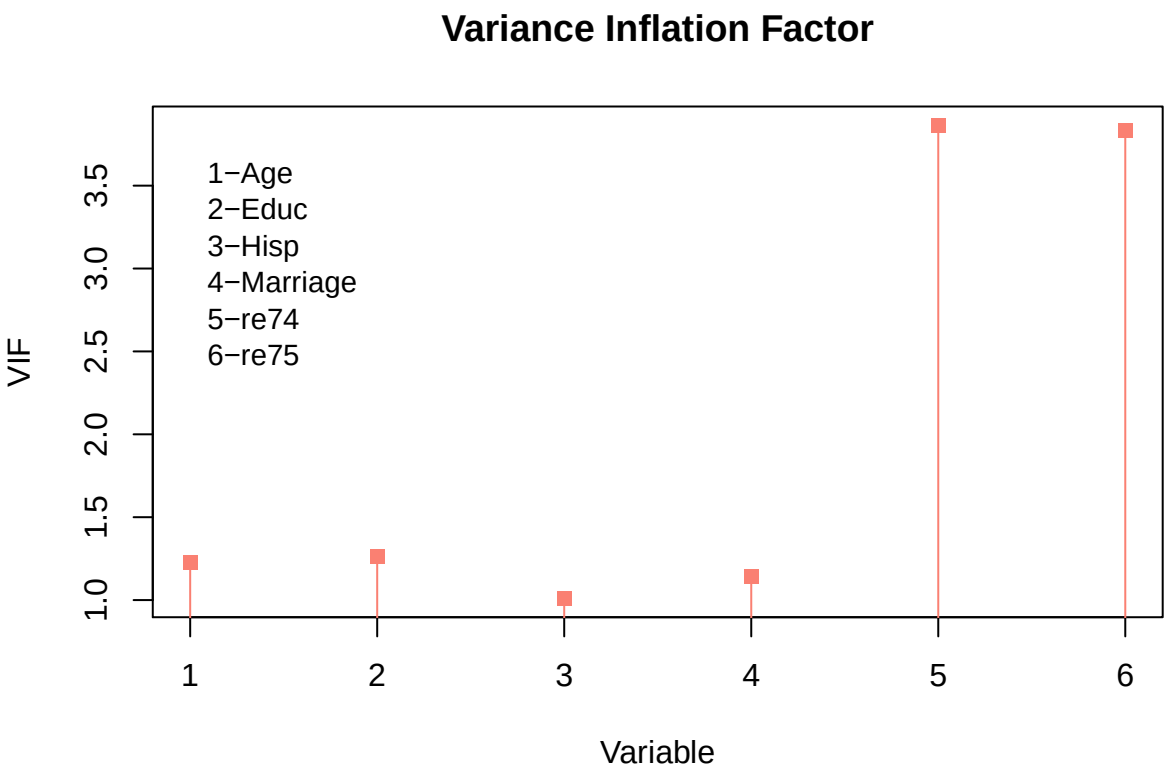
There are more residuals to the right of the zero line than the left; hence, there appears to be some underestimation within our model.



1e. VIF plot

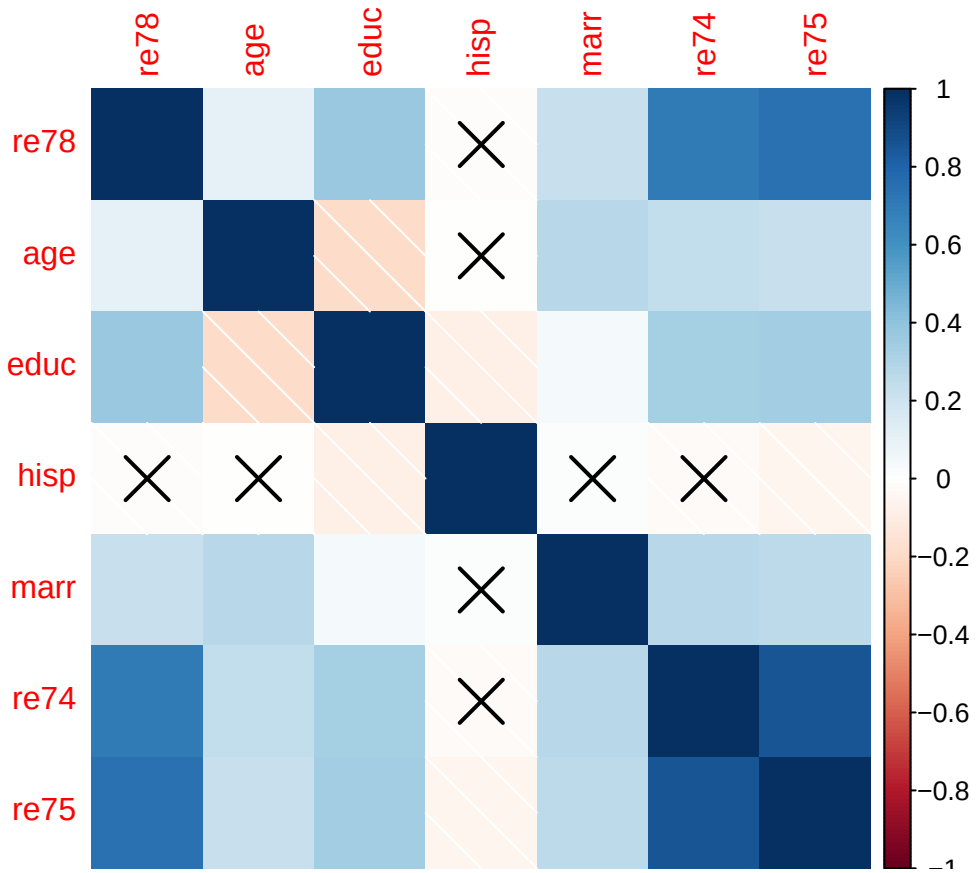
The variance inflation factors show that *re*74 and *re*75 introduce a large degree of variation into our model.

```
##      age     educ     hisp     marr     re74     re75
## 1.227765 1.264142 1.010674 1.142693 3.863304 3.833144
```



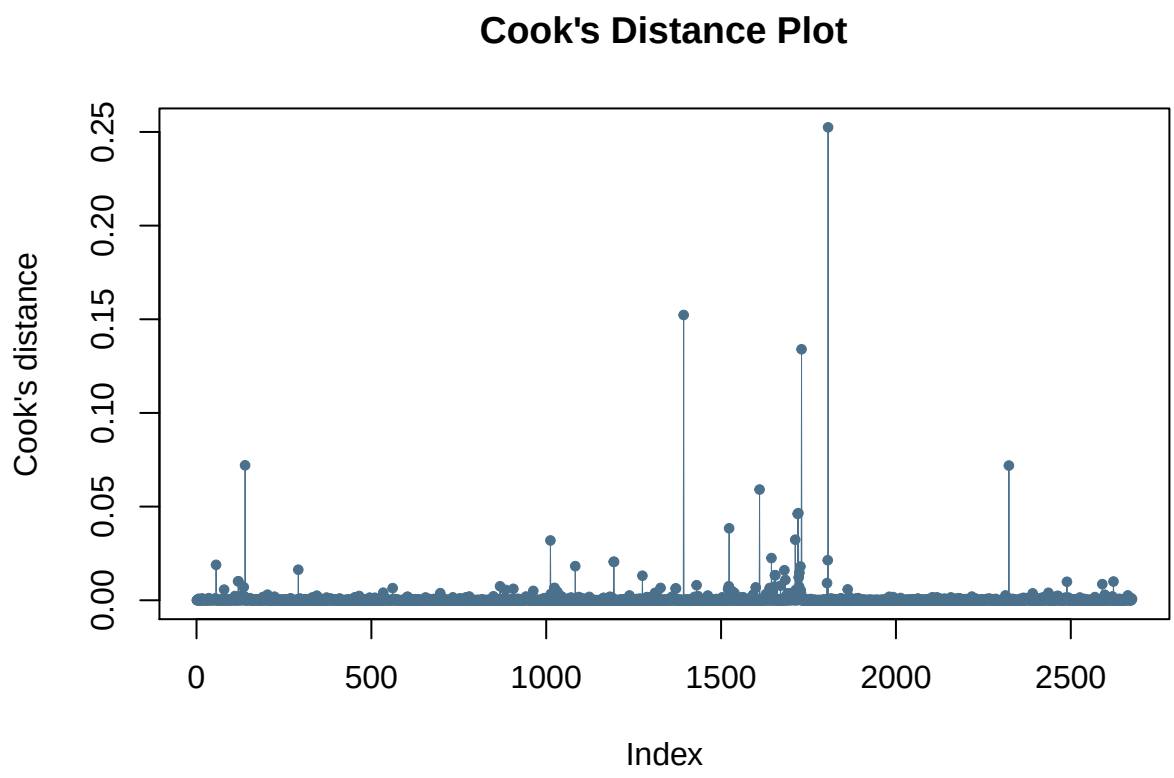
1f. Correlation plot

We compute the correlations between the variables, and display the correlation matrix as a plot below. The crosses denote statistically insignificant correlations. It is apparent that the variable *hisp* is uncorrelated with the majority of the variables in our model. Additionally, the high degree of correlation between *re*74 and *re*75 confirms our VIF computation above, as there may exist some collinearity between these two variables.



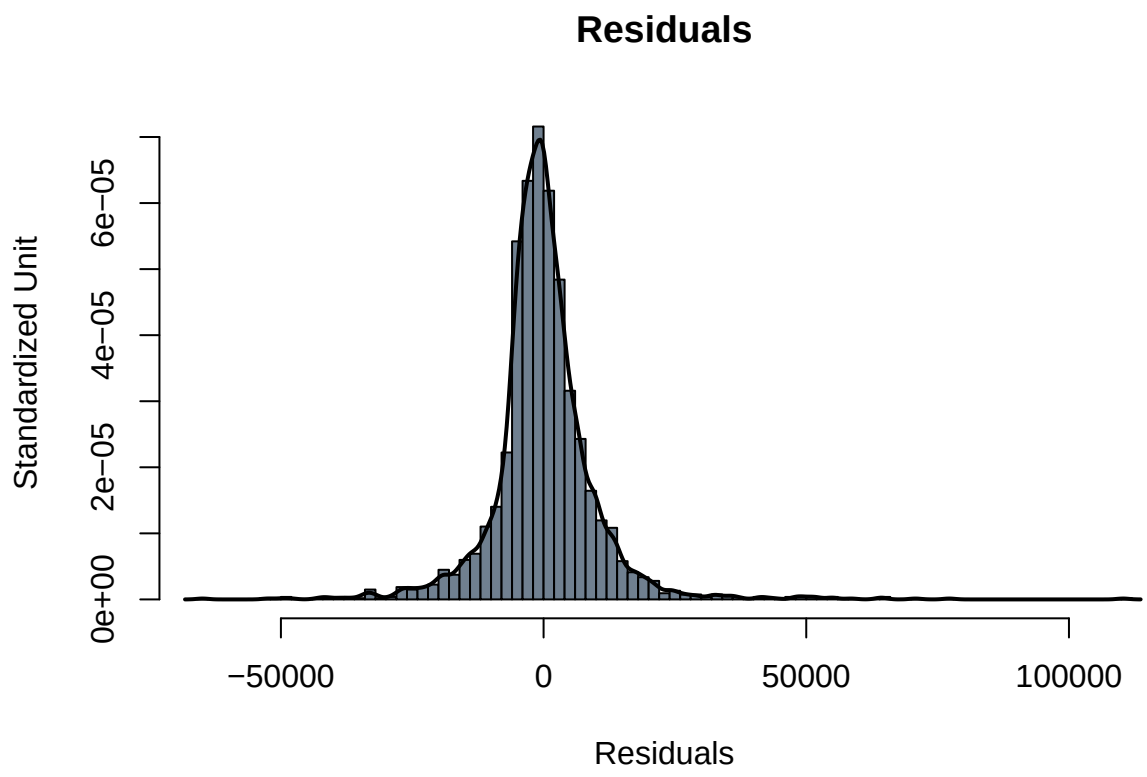
1g. Cooks Distance plot

There appear to be roughly five outliers. We could remove them from the data set and run the regression again to see if we get different results.



1h. Histogram of the residuals

The residuals are symmetric; however, the tails do not appear to match that of a normal distribution.



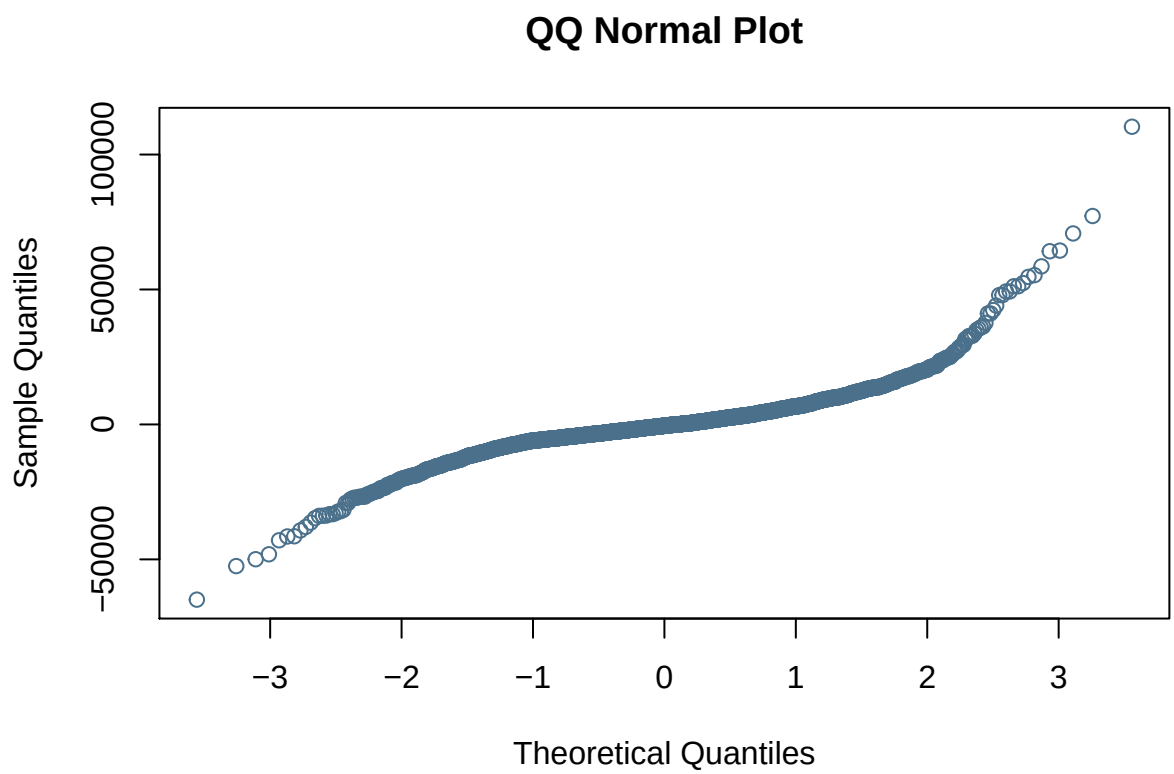
We check for normality in our residuals by using Jacque-Berra's Test for normality, and find that the residuals are not normally distributed.

```
jarque.bera.test(resid(reduc_model)) #Fails normality
```

```
##
##  Jarque Bera Test
##
## data:  resid(reduc_model)
## X-squared = 21372, df = 2, p-value < 2.2e-16
```

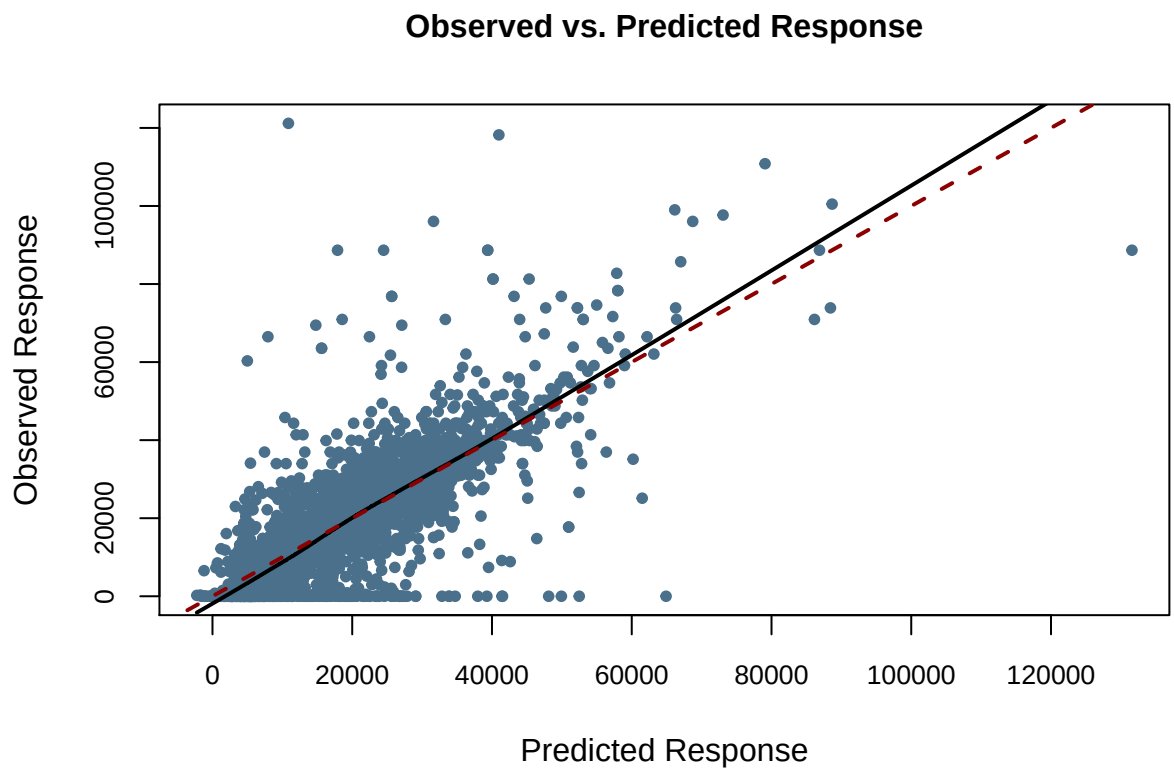
1i. Plot the QQ Normal Plot.

The QQ Normal plot shows that for most of the data, the model is actually fairly strong; however, in the lower quantiles, there appears to be a departure from the straight line. Therefore, our model may not be very representative for quantities of smaller values, and is weaker for quantities of larger values.



1j. Plot the observed v. predicted values, overlay a Lowess smoother

The Lowess smoother appears somewhat close to the line y=x; however, it diverges as the observed/prediced responses increase in value.



Problem 2:

```
## Warning in strptime(xx, f <- "%Y-%m-%d", tz = "GMT"): unknown timezone
## 'zone/tz/2018c.1.0/zoneinfo/America/Los_Angeles'
```

We begin by looking at the summary statistics. We use data from the St. Louis Federal Reserve with the `getSymbols` function from `Quantmod`. The FRED data returns GDP quarterly growth rates that are already seasonally adjusted.

Looking at the boxplot and the summary statistics, we can see that the S&P 500 returns have greater spread than the GDP growth rates.

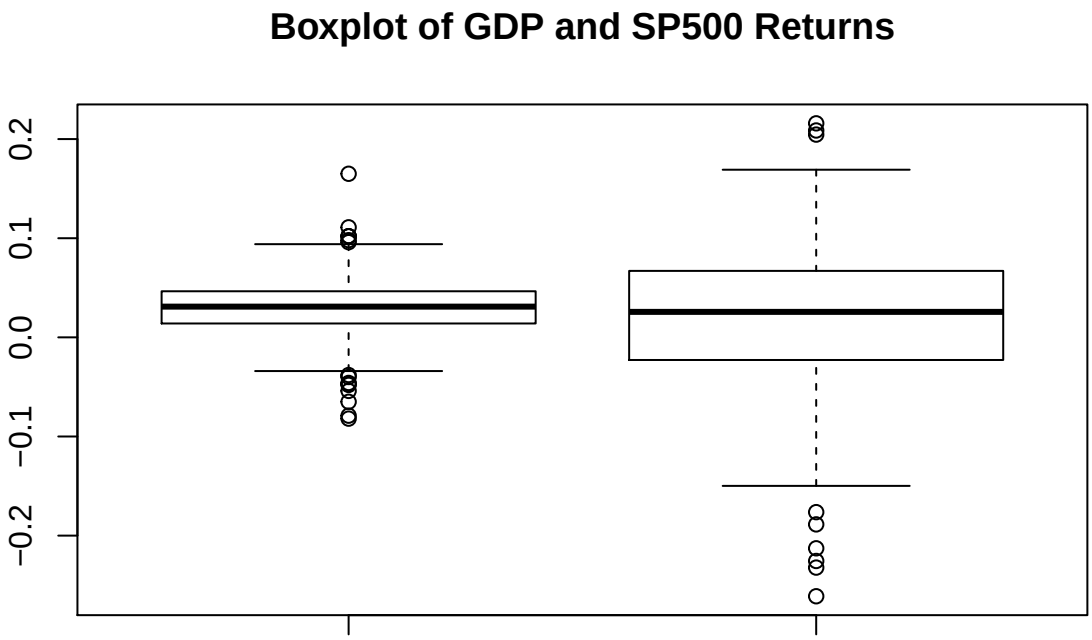
```r
# Sumamry Statistics
summary(gdp)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.08200  0.01400  0.03100  0.03061  0.04650  0.16500
```

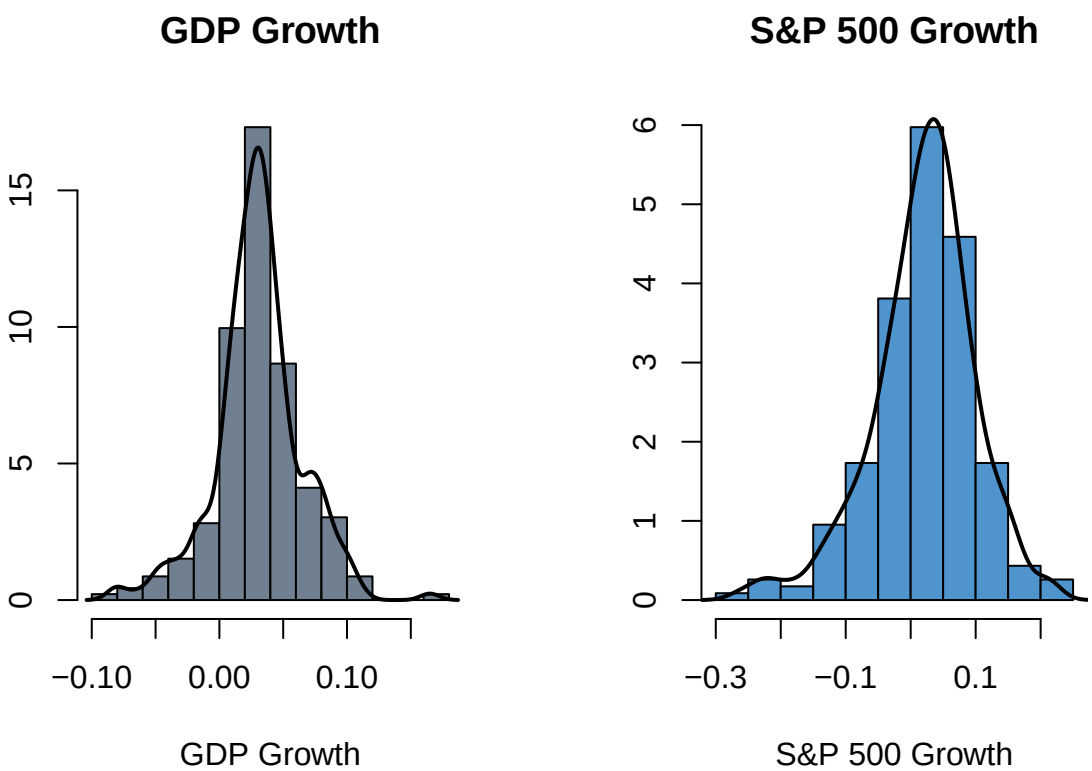
```r
summary(gspc)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.26116 -0.02282  0.02569  0.01947  0.06704  0.21587
```

```r
boxplot(gdp, gspc, main = "Boxplot of GDP and SP500 Returns")
```



Both distributions appear slightly skewed.

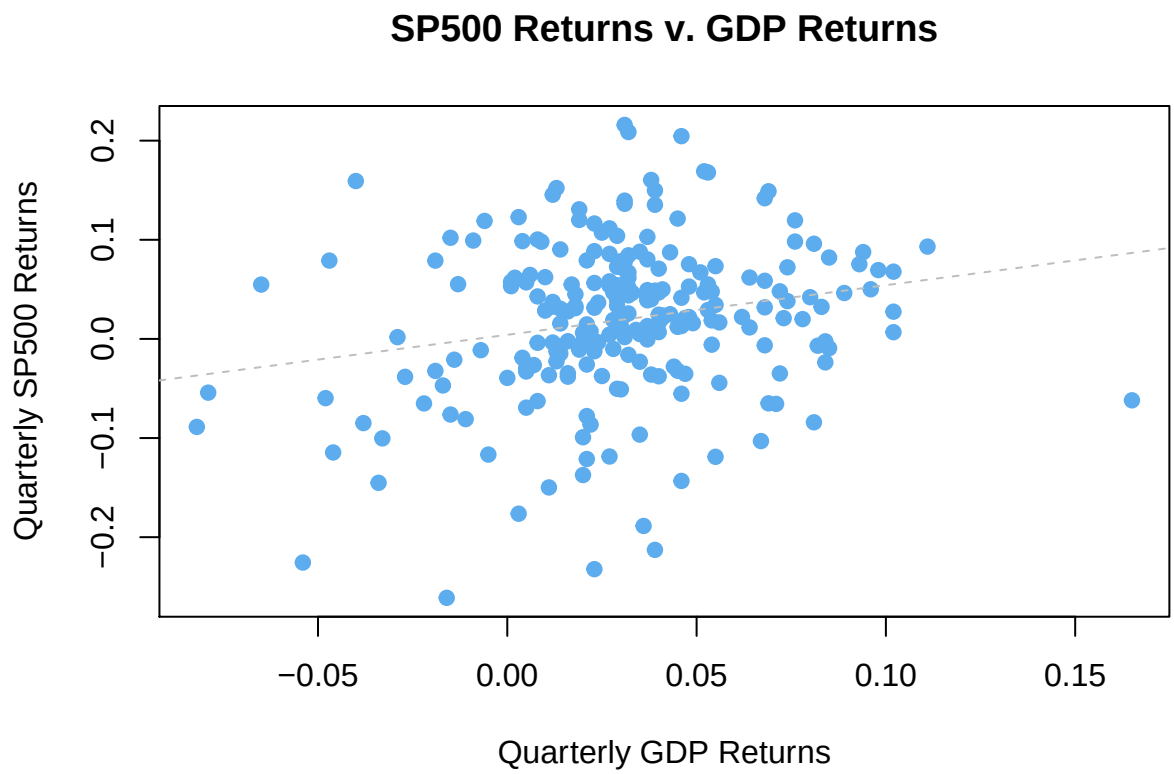


We compute a correlation of 0.2138981 between the contemporaneous series. There does not appear to be a lot of linear dependence between the two series. However, intuitively, we would hypothesize that as the economy performs better (GDP increases), the stock market would also be performing well (S&P increases). Therefore, there should exist some relationship between the two. What would be more interesting to look at would be the relationship between lagged values of the two series on contemperaneous observations.

```r
cor(gdp, gspc)  #Correlation value of 0.2138981
```

```
## [1] 0.2144277
```

```r
# Sanity check:
plot(gdp, gspc, pch = 19, col = "steelblue2", ylab = "Quarterly SP500 Returns",
    xlab = "Quarterly GDP Returns", main = "SP500 Returns v. GDP Returns")
abline(lm(gspc ~ gdp), lty = 2, col = "gray")
```



Problem 3:

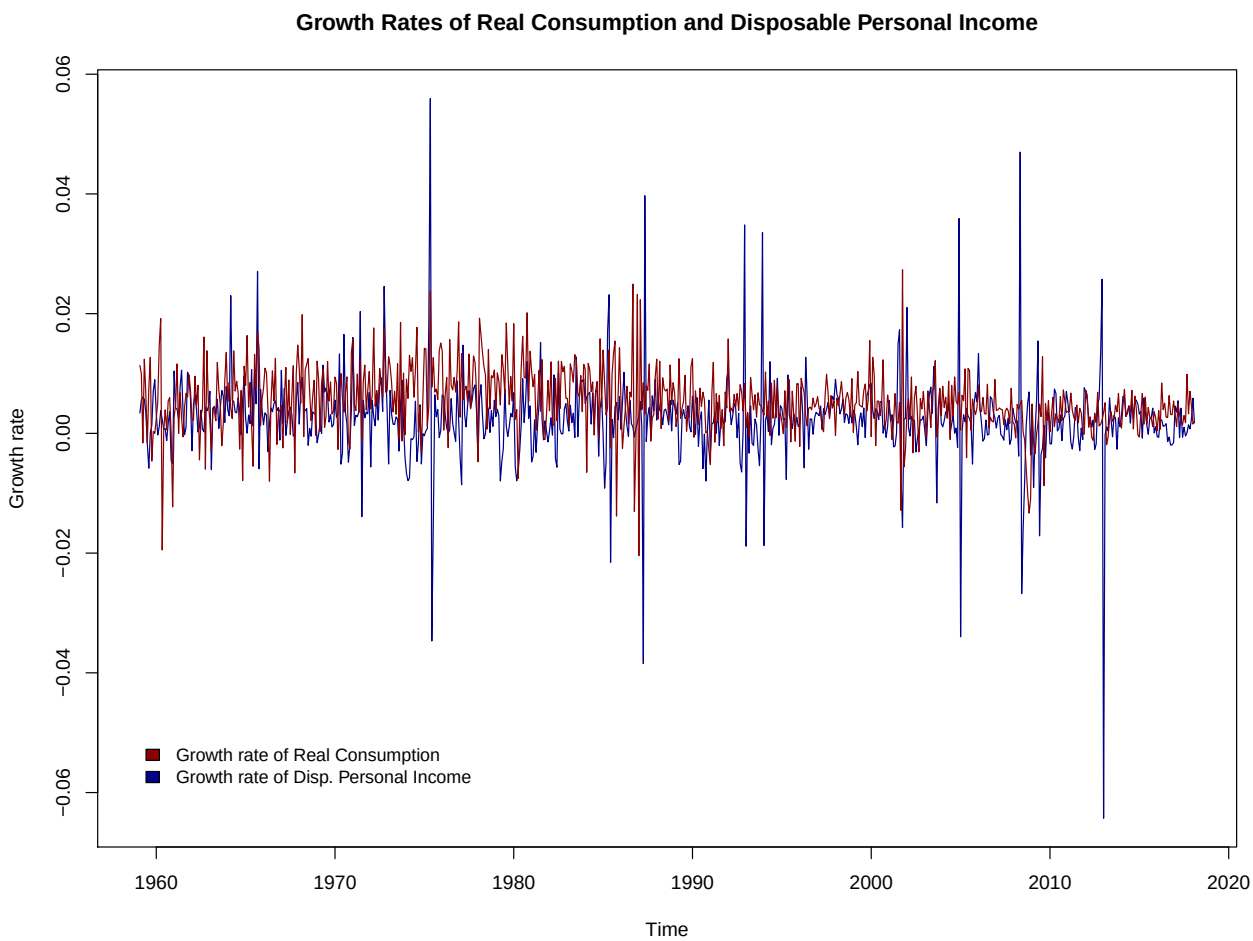
3a. Calculate growth rates of real consumption and disposable personal income and plot the data.

We calculate the growth rates by first taking the log of each series, and taking the first difference between observations.

```r
# Personal consumption expenditures
getSymbols("PCE", src = "FRED")
pce <- as.data.frame(PCE)
pce_ts <- ts(pce$PCE, start = 1959, freq = 12)
# Growth rates:
log.pce <- log(pce_ts)
r_pce <- diff(log.pce)
```

```r
# Disposable personal income:
getSymbols("DSPIC96", src = "FRED")
dpi <- as.data.frame(DSPIC96)
dpi_ts <- ts(dpi$DSPIC96, start = 1959, freq = 12)
# Growth rates:
log.dpi <- log(dpi_ts)
r_dpi <- diff(log.dpi)
```

We can see from the plot that there exists greater fluctuation in disposable personal income than in consumption. From an economic perspective, we can explain this with the permanent income hypothesis, which argues that people base consumption on an average of their income over time, rather than their current income. This is because people prefer "smooth" paths of consumption. So, while we expect the two series to be related, we would expect large fluctuations in current disposable income to result in smaller fluctuations in consumption expenditure.



3b. Regress consumption growth on disposable income growth.

From the regression results, we find that for a 100% increase in the return in disposable personal income, we would expect real consumption growth to increase by 0.1024 (10.24%).

```r
model<-lm(r_pce~r_dpi)
summary(model)
```

```
##
## Call:
## lm(formula = r_pce ~ r_dpi)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0258263 -0.0031126 -0.0004632  0.0031402  0.0238081
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0051047  0.0002165  23.582  < 2e-16 ***
## r_dpi       0.1012722  0.0283099   3.577 0.000371 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005422 on 707 degrees of freedom
## Multiple R-squared:  0.01778,    Adjusted R-squared:  0.01639
## F-statistic:  12.8 on 1 and 707 DF,  p-value: 0.0003709
```

3c. Add a lag of growth in disposable income to the equation.

Adding a lag to the equation does not help explain the change in real consumption. The p-value associated with the lagegd value of disposable personal income is 0.248820, which is greater than $\alpha = 0.05$. Therefore, we conclude that the lagged value is statistically insignificant.

```r
lag.r_dpi<-append(NA, r_dpi[-length(r_dpi)])
model2<-lm(r_pce~r_dpi+lag.r_dpi)
summary(model2)
```

```
##
## Call:
## lm(formula = r_pce ~ r_dpi + lag.r_dpi)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0257946 -0.0030111 -0.0004243  0.0031844  0.0242324
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0049985  0.0002329  21.460  < 2e-16 ***
## r_dpi       0.1062206  0.0286463   3.708 0.000225 ***
## lag.r_dpi   0.0327655  0.0286463   1.144 0.253096
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.00542 on 705 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.01958,    Adjusted R-squared:  0.0168
## F-statistic: 7.041 on 2 and 705 DF,  p-value: 0.0009385
```

Problem 4

Note: while the problem does not specifically ask for this, a good check for stationarity would be to look at the ACF/PACF's of the individual series. Additionally, for each series, the average of the series is provided on the plots as a gray line.

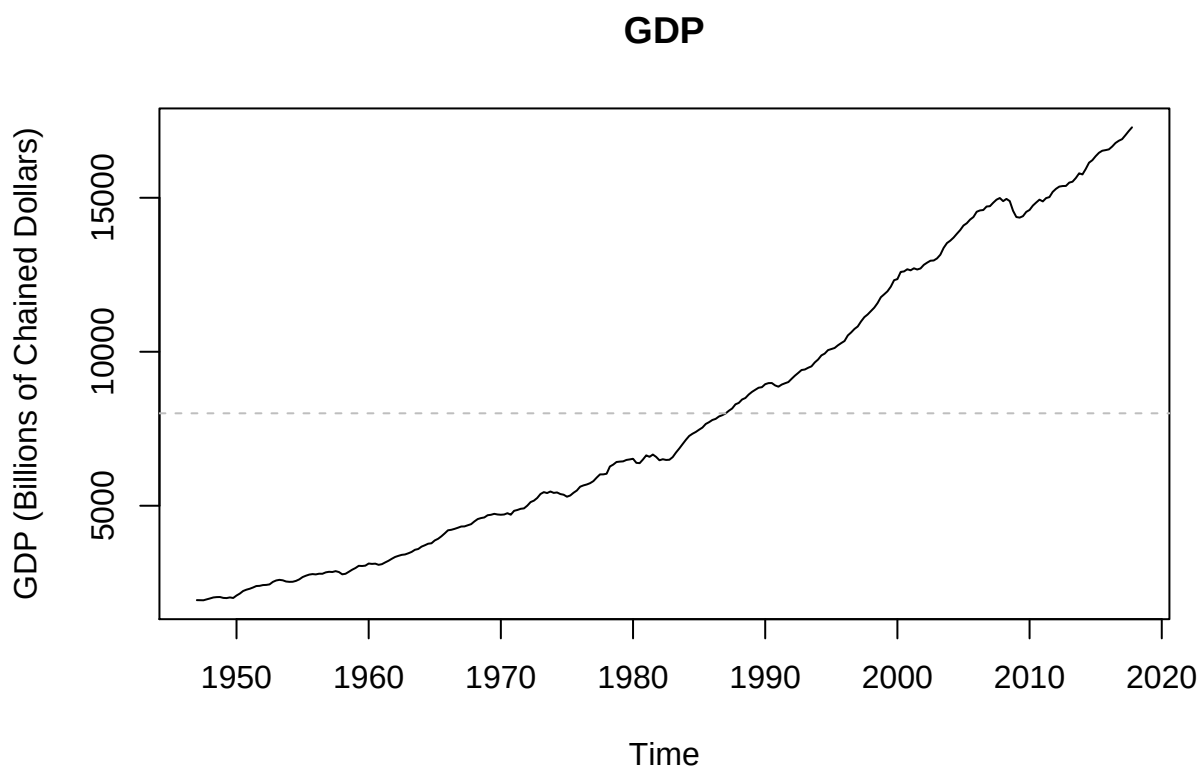
4a. GDP

GDP is formally defined as the gross domestic product, which is the inflation adjusted value of goods and services produced by labor and property located in the United States. The data begins at January 1, 1947, and extends through October 1, 2016. The data is quarterly. From the plot, we can see that the the series is not mean reverting, and we therefore conclude that the series is not stationary. There exists a strong upward trend, with the exception of several recession periods, in which, consistent with economic theory, GDP dips.

```r
# GDP:
getSymbols("GDPC1", src = "FRED")
```

```
## [1] "GDPC1"
```

```r
gdp <- ts(GDPC1$GDPC1, start = 1947, freq = 4)
plot(gdp, ylab = "GDP (Billions of Chained Dollars)", main = "GDP")
abline(h = mean(gdp), col = "gray", lty = 2)
```



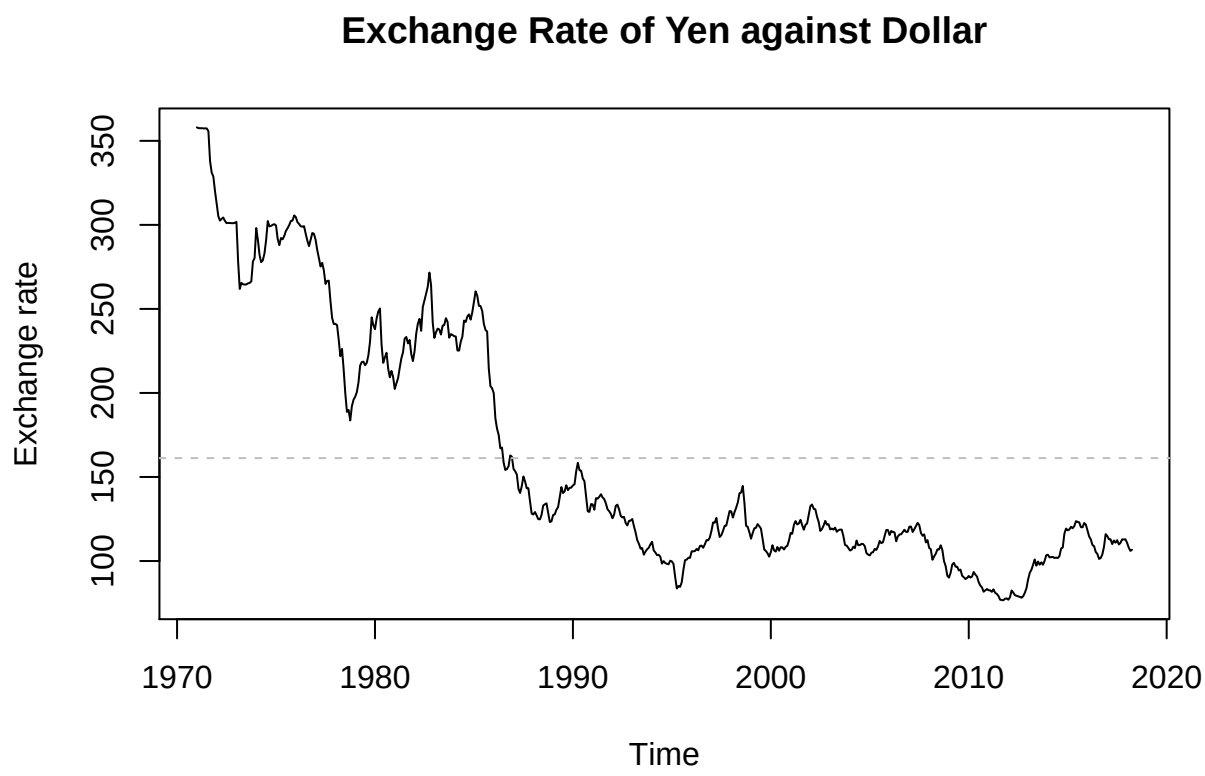
4b. Exchange rate of Japanese yen to US Dollar

The foreign exchange rate of Japanese yen to US dollar refers to the number of yen that we can exchange for $1 (US dollar). The date range of this data set is January 1971 to April 2017, and the data exists on a monthly frequency. There exists a downward trend in the series. The data is not mean reverting, so it cannot be first order stationary.

```
# EXPJ
getSymbols("EXJPUS", src = "FRED")
```

```
## [1] "EXJPUS"
```

```
exj <- ts(EXJPUS$EXJPUS, start = 1971, freq = 12)
plot(exj, ylab = "Exchange rate", main = "Exchange Rate of Yen against Dollar")
abline(h = mean(exj), col = "gray", lty = 2)
```



4c. 10-year US Treasury constant maturity yield

This refers to the yields on actively traded non-inflation-indexed issues adjusted to constant maturities. The data ranges from April 1953 to March 2017 at a monthly level. There appears to be the presence of some cycles or seasonality in the series. There may be a quadratic trend within the series. Taking the first difference should yield a stationary process.

```
# GS10
getSymbols("GS10", src = "FRED")
```

```
## [1] "GS10"
```

```
gs10 <- ts(GS10$GS10, start = 1953 + (1/4), freq = 12)
plot(gs10, ylab = "Yield", main = "10 year US treasury constant maturity yield")
abline(h = mean(gs10), col = "gray", lty = 2)
```

4d. Unemployment Rate

The unemployment rate refers to the percentage of people who are unemployed in the labor force (which is defined as people 16 years of age or older). The data ranges from January 1948 to March 2017 at a monthly frequency. The series does exhibit some stationarity. There can be observed cycles and seasonal factors as well.

```
# Unemployment Rate
getSymbols("UNRATE", src = "FRED")
```

```
## [1] "UNRATE"
```

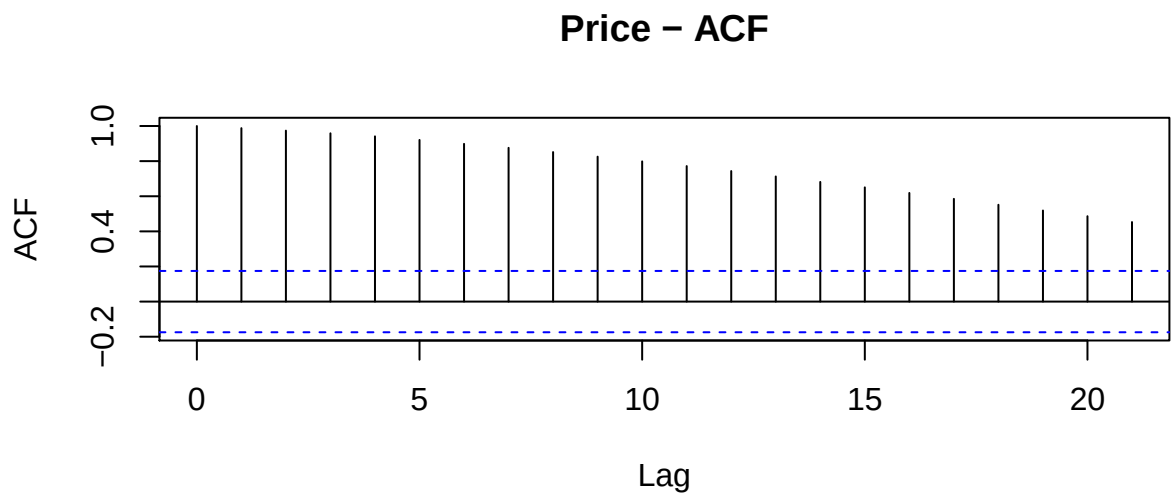
```
unemploy <- ts(UNRATE$UNRATE, start = 1948, freq = 12)
plot(unemploy, ylab = "Unemployment Rate", main = "US Unemployment Rate")
abline(h = mean(unemploy), col = "gray", lty = 2)
```

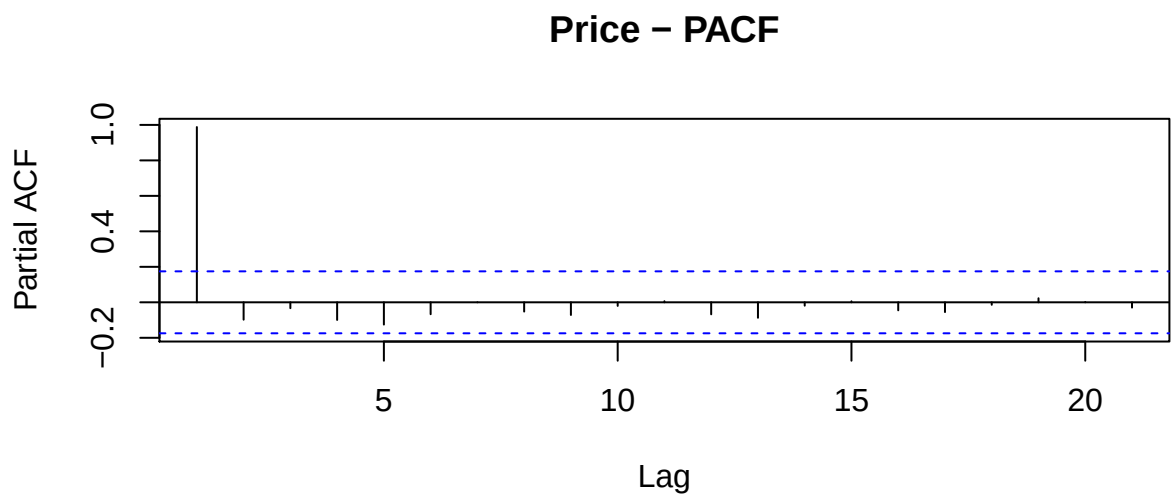
Problem 5

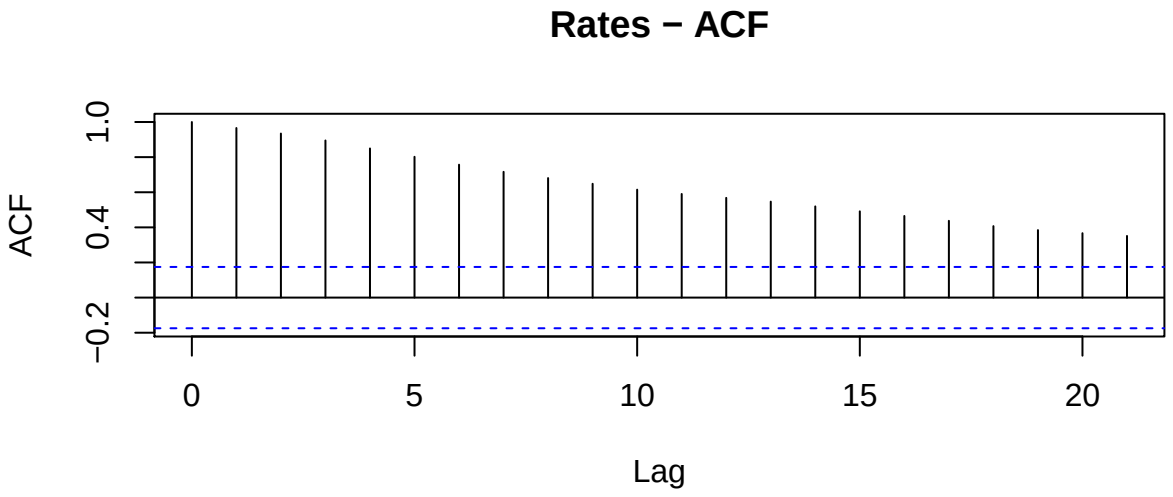
We find that in both the ACF/PACF of the price and interest rate series, there exists a decay in the autocorrelation function, and a large spike in the partial autocorrelation function. In the housing price growth series, we see that there exists some fluctuation in the ACF, hinting at potential cycles, and several significant spikes in the partial autocorrelation. The autocorrelation functions for the interest rate changes series shows no significant spikes in either the ACF or the PACF, which means that it will be very difficult to model interest rate changes using lagged values of the original series and lagged innovations.

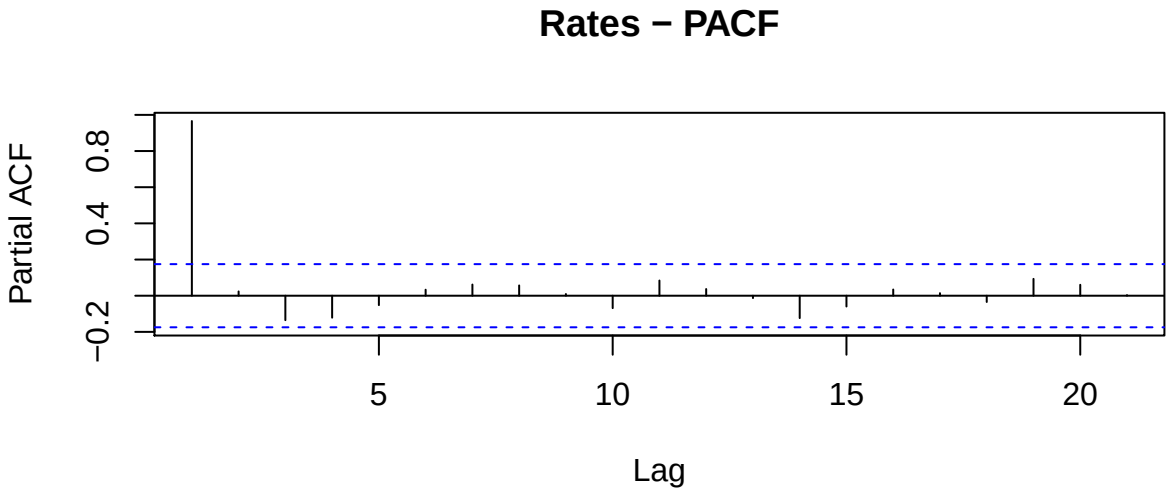
We look also at the autocorrelation functions of the annual data. There exists a faster decay in the autocorrelation function (ACF) of the housing price and the interest rates. This makes sense, because we are dealing with longer periods of time. In both cases, we see the same pattern as the quarterly data: a spike in the PACF, and a decay in the ACF. This would hint at an autoregressive process.

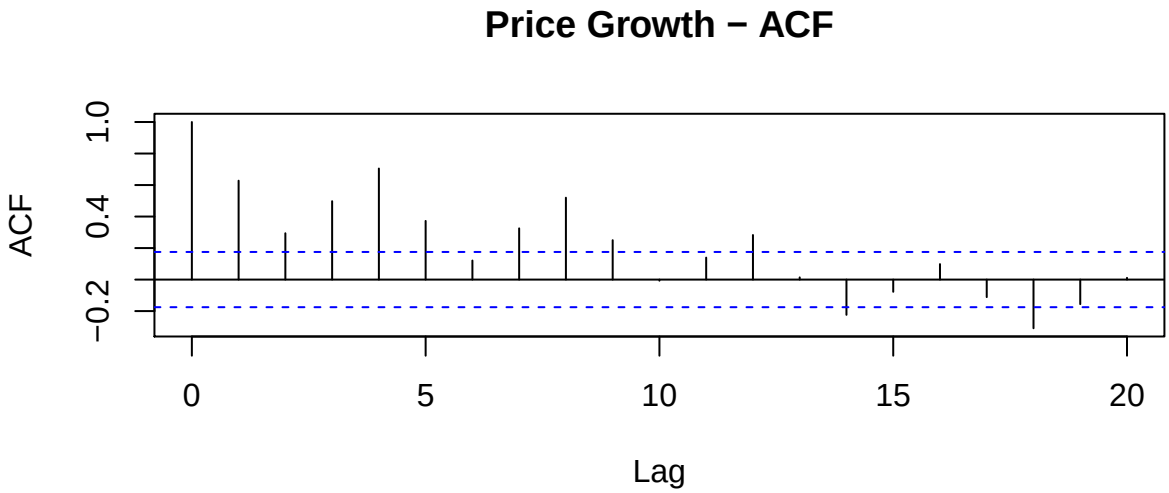
In the ACF/PACF of the annual price growth rates, we see that this too follows an autoregressive process, with a significant spike at lag 1 in the PACF, and a decay in the spikes in the ACF. This contrasts with the ACF/PACF functions of the price growth rates on a quarterly frequency. The interest rate changes at an annual level do not show much dependence upon previous values, much like the quarterly level data.

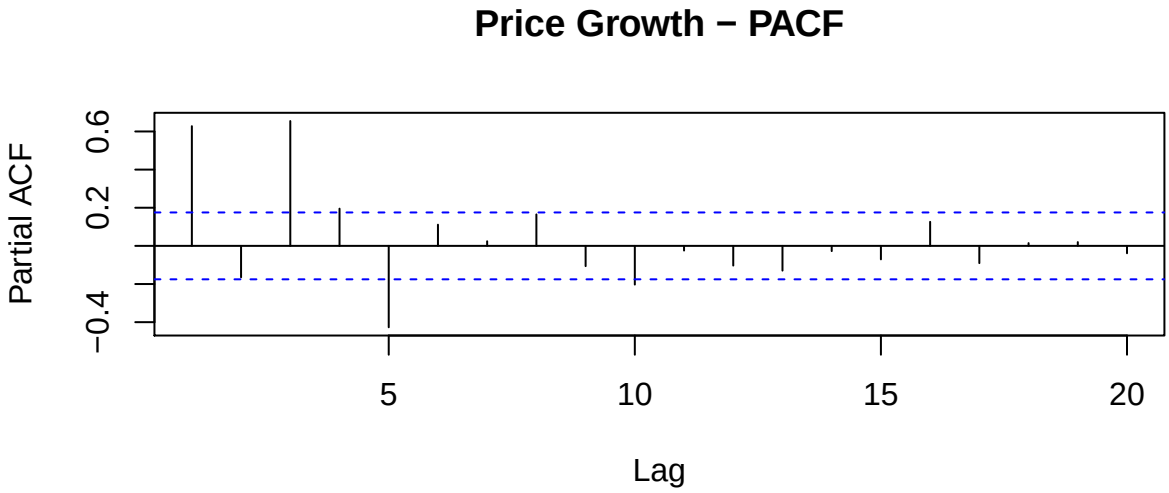

Price − ACF
ACF
1.0
0.4
−0.2
0
5
10
15
20
Lag

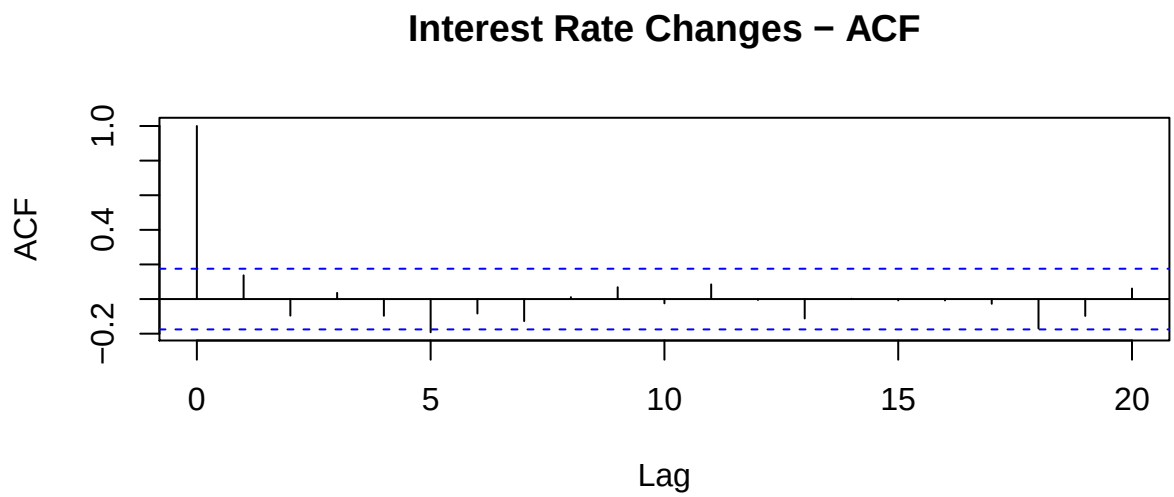

Price − PACF
Partial ACF
1.0
0.4
−0.2
5
10
15
20
Lag

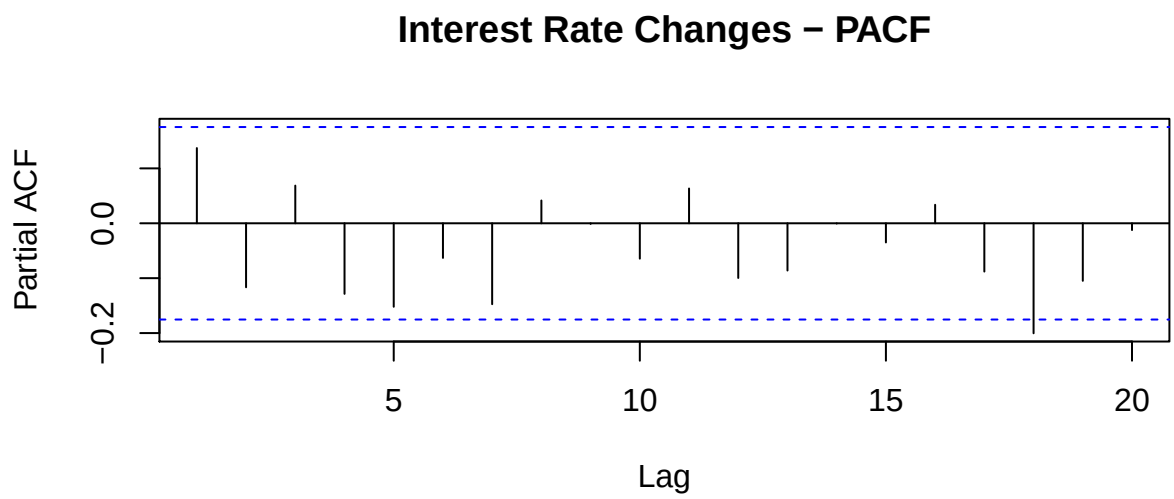

Rates − ACF
ACF
1.0
0.4
−0.2
0
5
10
15
20
Lag

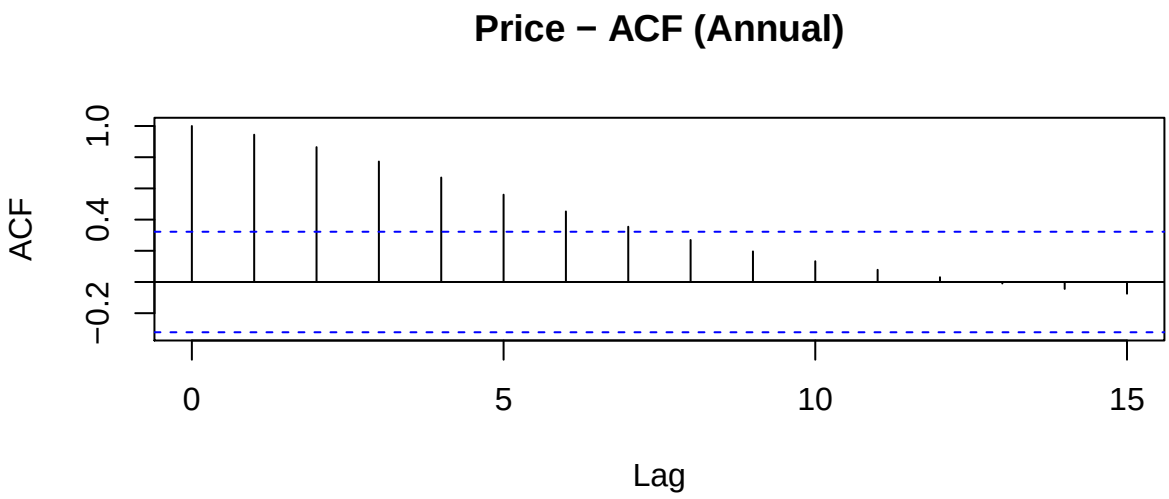

Rates − PACF
Partial ACF
0.8
0.4
−0.2
5
10
15
20
Lag

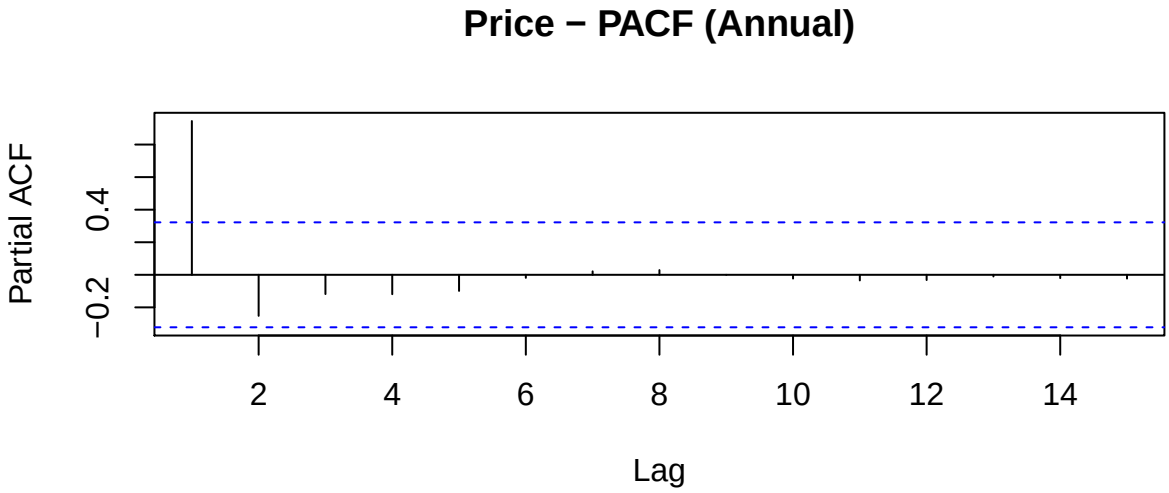

Price Growth − ACF
ACF
1.0
0.4
−0.2
0
5
10
15
20
Lag

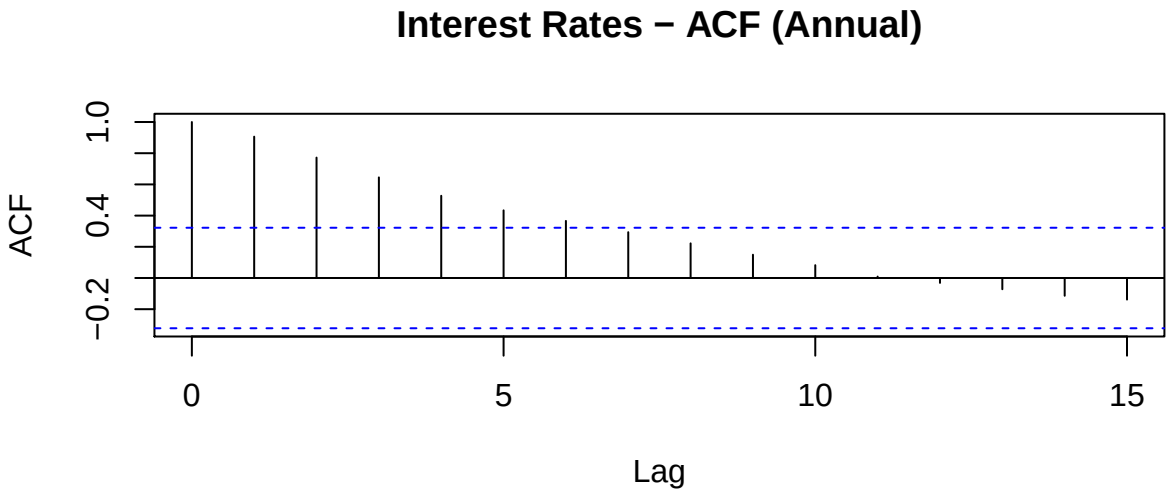

Price Growth − PACF
Partial ACF
0.6
0.2
−0.4
5
10
15
20
Lag

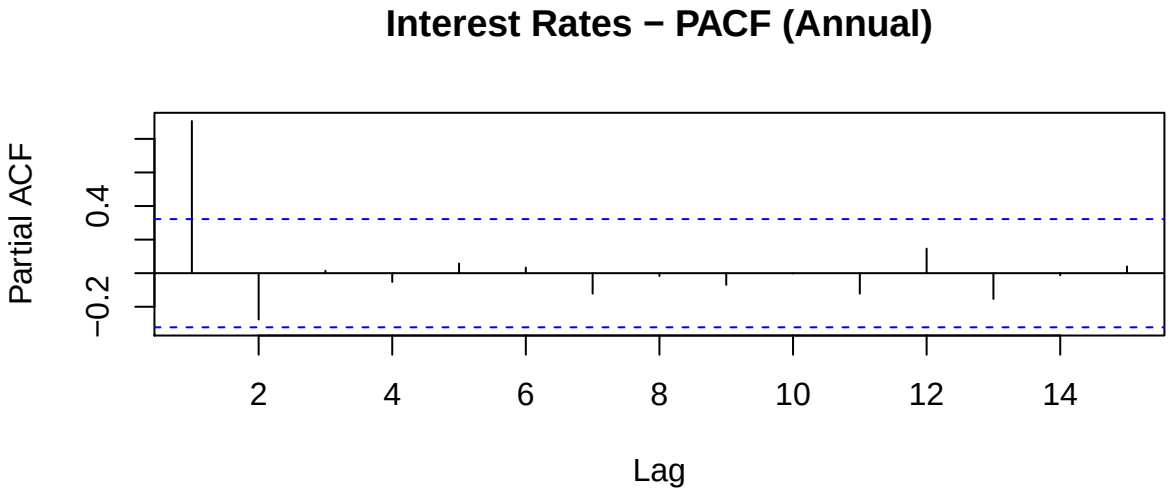

Interest Rate Changes − ACF
ACF
1.0
0.4
−0.2
0
5
10
15
20
Lag

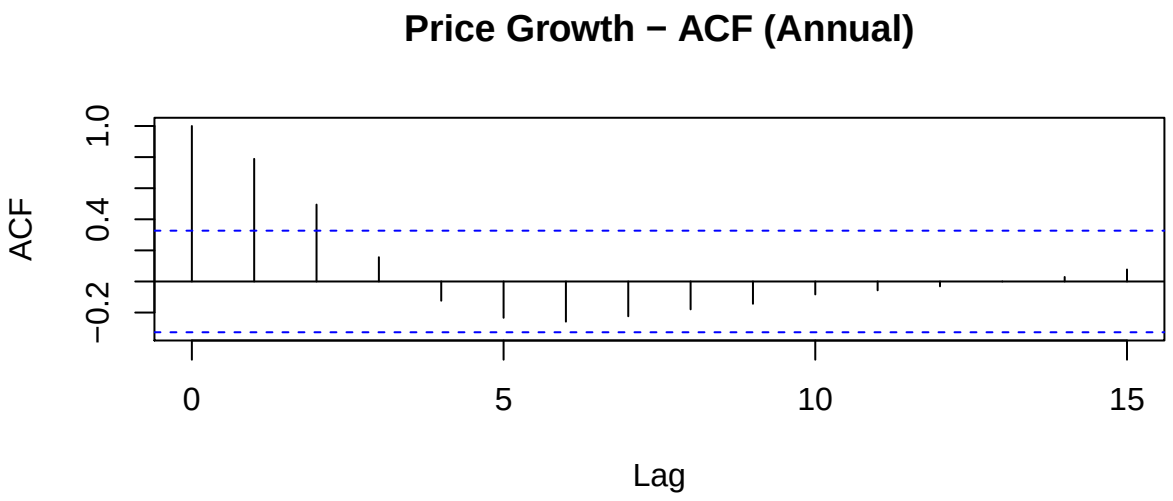

Interest Rate Changes − PACF
Partial ACF
0.0
−0.2
5
10
15
20
Lag

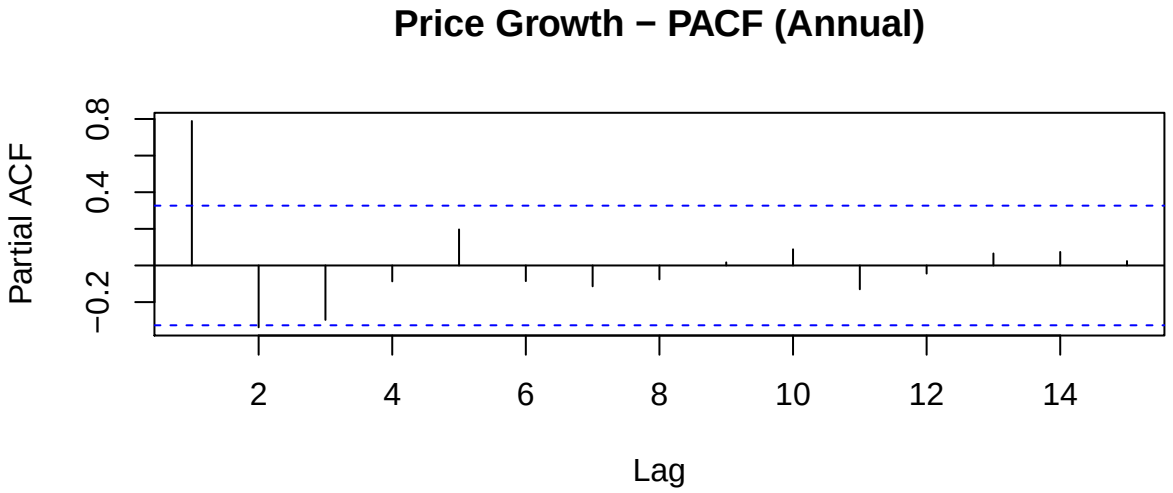

Price − ACF (Annual)
ACF
1.0
0.4
−0.2
0
5
10
15
Lag

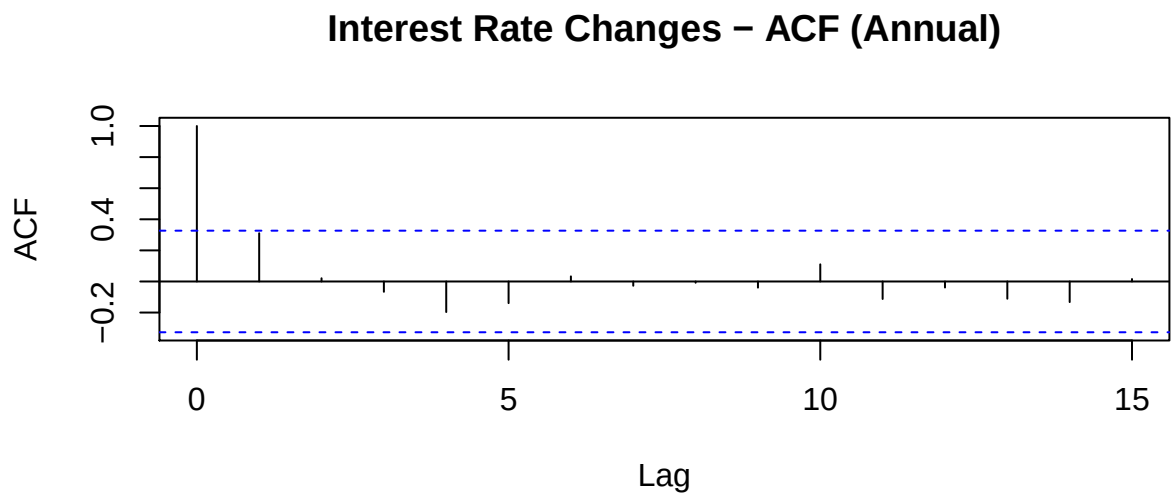

Price − PACF (Annual)
Partial ACF
0.4
−0.2
2
4
6
8
10
12
14
Lag

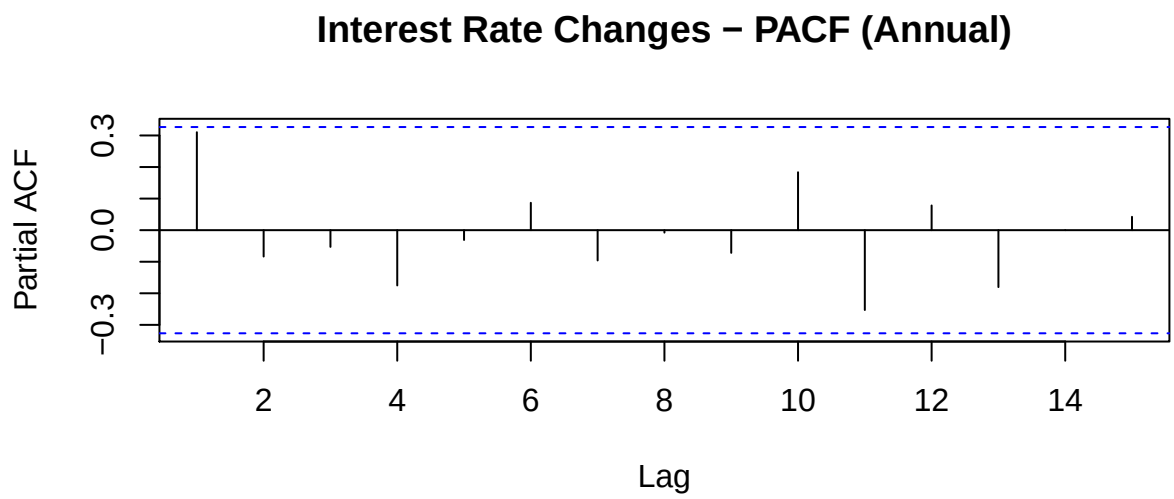

Interest Rates − ACF (Annual)
ACF
1.0
0.4
−0.2
0
5
10
15
Lag


Interest Rates − PACF (Annual)
Partial ACF
0.4
−0.2
2
4
6
8
10
12
14
Lag


Price Growth − ACF (Annual)
ACF
1.0
0.4
−0.2
0
5
10
15
Lag


Price Growth − PACF (Annual)
Partial ACF
0.8
0.4
−0.2
2
4
6
8
10
12
14
Lag





R-Code:

```
library(DAAG)
library(MASS)
library(leaps)
library(car)
library(lmtest)
library(corrplot)
library(tseries)
library(quantmod)
# Load data
data(nsw74psid1)
names(nsw74psid1)

# Problem 1: 1a: Plot a histogram of each variable
quartz(height = 5.5, width = 11)
```

```
par(mfrow = c(2, 5))
# Binary variables
truehist(nsw74psid1$trt, xlab = "PSID = 0, NSW = 1", main = "Study Enrolled",
    col = "steelblue4")
truehist(nsw74psid1$black, xlab = "Not black = 0, Black = 1",
    main = "Black", col = "steelblue4")
truehist(nsw74psid1$hisp, xlab = "Not hispanic = 0, Hispanic = 1",
    main = "Hispanic", col = "steelblue4")
truehist(nsw74psid1$marr, xlab = "Not married = 0, Married = 1",
    main = "Married", col = "steelblue4")
truehist(nsw74psid1$nodeg, xlab = "Completed HS = 0, Dropout = 1",
    main = "No Degree", col = "steelblue4")
# Non-binary variables
truehist(nsw74psid1$educ, xlab = "Years of Education", main = "Education",
    col = "skyblue3")
lines(density(nsw74psid1$educ), lwd = 2)
truehist(nsw74psid1$age, xlab = "Age (in years)", main = "Age",
    col = "skyblue3")
lines(density(nsw74psid1$age), lwd = 2)
truehist(nsw74psid1$re74, xlab = "Real earnings ($)", main = "Real Earnings in 1974",
    col = "skyblue3")
lines(density(nsw74psid1$re74), lwd = 2)
truehist(nsw74psid1$re75, xlab = "Real earnings ($)", main = "Real Earnings in 1975",
    col = "skyblue3")
lines(density(nsw74psid1$re75), lwd = 2)
truehist(nsw74psid1$re78, xlab = "Real earnings ($)", main = "Real Earnings in 1978",
    col = "skyblue3")
lines(density(nsw74psid1$re78), lwd = 2)

# 1b: Estimate the full regression model
full_model <- lm(re78 ~ ., data = nsw74psid1)
summary(full_model)

# 1c: Mallows CP

reg.cp = regsubsets(re78 ~ ., method = c("exhaustive"), nbest = 1,
    data = nsw74psid1, nvmax = 9)
subsets(reg.cp, statistic = "cp", legend = F, main = "Mallows CP",
    col = "steelblue4")
# There exists several ways to interpret this: We can look at
# the absolute minima that Mallows CP achieves: Determine
# which variables are included in minimum:
summary.cp <- summary(reg.cp)
summary.cp$which[which.min(summary.cp$cp), ]  #6 variables
# AGE, EDUC, HISP, MARR, RE74, RE75

# However, we see that using 5 variables also provides
# comprable performance For simplicity, we stick with using 6
```

```
# variables.

reduc_model <- lm(re78 ~ age + educ + hisp + marr + re74 + re75,
    data = nsw74psid1)
summary(reduc_model)
resettest(reduc_model, power = 2)
resettest(reduc_model, power = 3)
resettest(reduc_model, power = 4)
# A quick RESET test yields that the model is misspecified.
# However, the inclusion of higher order terms shows that the
# model continues to be misspecified, which may hint at the
# fact that we are missing some key variables that may help
# us estimate our dependent variable more. For simplicity, we
# continue with our analysis using model (2).

# 1d. Residuals v. Fitted Values
plot(resid(reduc_model), fitted(reduc_model), xlab = "Residuals",
    ylab = "Fitted Values", main = "Residuals v. Fitted Values",
    pch = 19, col = "navyblue")

# 1e. VIF plot
vif(reduc_model)
plot(vif(reduc_model), col = "salmon", ylab = "VIF", main = "Variance Inflation Factor",
    xlab = "Variable", pch = 15)
lines(vif(reduc_model), type = "h", col = "salmon")
legend(0.8, 3.8, c("1-Age", "2-Educ", "3-Hisp", "4-Marriage",
    "5-re74", "6-re75"), bty = "n", cex = 0.95)


# 1f. Correlation plot
df_corr <- data.frame(re78 = nsw74psid1$re78, age = nsw74psid1$age,
    educ = nsw74psid1$educ, hisp = nsw74psid1$hisp, marr = nsw74psid1$marr,
    re74 = nsw74psid1$re74, re75 = nsw74psid1$re75)
M <- cor(df_corr)
# To check for significance of correlations:
cor.mtest <- function(mat, conf.level = 0.95) {
    mat <- as.matrix(mat)
    n <- ncol(mat)
    p.mat <- lowCI.mat <- uppCI.mat <- matrix(NA, n, n)
    diag(p.mat) <- 0
    diag(lowCI.mat) <- diag(uppCI.mat) <- 1
    for (i in 1:(n - 1)) {
        for (j in (i + 1):n) {
            tmp <- cor.test(mat[, i], mat[, j], conf.level = conf.level)
            p.mat[i, j] <- p.mat[j, i] <- tmp$p.value
            lowCI.mat[i, j] <- lowCI.mat[j, i] <- tmp$conf.int[1]
            uppCI.mat[i, j] <- uppCI.mat[j, i] <- tmp$conf.int[2]
        }
```

```
    }
    return(list(p.mat, lowCI.mat, uppCI.mat))
}
res1 <- cor.mtest(df_corr, 0.95)
res2 <- cor.mtest(df_corr, 0.99)
## specialized the insignificant value according to the
## significant level
corrplot(M, p.mat = res1[[1]], sig.level = 0.05, method = "shade")
title("Correlation Plot")

# 1g. Cooks Distance plot
cook = cooks.distance(reduc_model)
quartz()
plot(cook, ylab = "Cook's distance", type = "o", main = "Cook's Distance Plot",
    col = "skyblue4", pch = 20, lwd = 0.25)

# 1h. Histogram of the residuals
quartz()
truehist(resid(reduc_model), main = "Residuals", col = "slategrey",
    xlab = "Residuals", ylab = "Standardized Unit")
lines(density(resid(reduc_model)), lwd = 2)

# Jaque-Berra Test for Normality:
jarque.bera.test(resid(reduc_model))  #Fails normality

# 1i. Plot the QQ Normal Plot.
quartz()
qqnorm(reduc_model$residuals, col = "skyblue4", main = "QQ Normal Plot")

# 1j. Plot the observed v. predicted values, overlay a Lowess
# smoother
quartz()
plot(reduc_model$fit, nsw74psid1$re78, pch = 20, col = "skyblue4",
    cex = 1, xlab = "Predicted Response", ylab = "Observed Response",
    main = "Observed vs. Predicted Response", cex.axis = 0.8,
    cex.main = 1)
lines(lowess(reduc_model$fit, nsw74psid1$re78), lwd = 2)
abline(0, 1, col = "darkred", lwd = 2, lty = 2)

# Problem 2: Download the US GDP quarterly growth rates and
# the S&P 500 quarterly returns For both series compute the
# descriptive statistics & their histograms Are the two
# series contemporaneously correlated?

# Pull data: Note: we pull FRED data, which is already
# seasonally adjusted.
getSymbols("A191RL1Q225SBEA", src = "FRED")
head(A191RL1Q225SBEA)
```

```
gdp <- as.data.frame(A191RL1Q225SBEA)
getSymbols("^GSPC", src = "yahoo", from = "1960-01-01")
gspc <- as.data.frame(quarterlyReturn(GSPC))
gspc <- gspc$quarterly.returns
index = which(row.names(gdp) == "1960-04-01") - 1
gdp <- gdp[-(1:index), ]
# GDP is currently returned as percentage return, so we
# divide by 100:
gdp <- gdp/100
gspc <- gspc[1:(length(gspc) - 3)]
length(gdp)
length(gspc)

# Sumamry Statistics
summary(gdp)
summary(gspc)
boxplot(gdp, gspc)

# Histograms:
par(mfrow = c(1, 2))
truehist(gdp, col = "slategray", xlab = "GDP Growth", main = "GDP Growth")
lines(density(gdp), lwd = 2)
truehist(gspc, col = "steelblue3", xlab = "S&P 500 Growth", main = "S&P 500 Growth")
lines(density(gspc), lwd = 2)


cor(gdp, gspc)  #Correlation value of 0.2138981
# Sanity check:
plot(gdp, gspc, pch = 19, col = "steelblue2", ylab = "Quarterly SP500 Returns",
    xlab = "Quarterly GDP Returns", main = "SP500 Returns v. GDP Returns")
abline(lm(gspc ~ gdp), lty = 2, col = "gray")

# Problem 3: Download monthly data on real personal
# consumption expenditures and real disposable personal
# income from the Federal Reserve Economic Database. Take a
# sample starting on 1959-01-01 to the present.  3a.
# Calculate growth rates of real consumption and disposable
# personal income and plot the data Personal consumption
# expenditures
getSymbols("PCE", src = "FRED")
pce <- as.data.frame(PCE)
pce_ts <- ts(pce$PCE, start = 1959, freq = 12)
# Growth rates:
log.pce <- log(pce_ts)
r_pce <- diff(log.pce)

# Disposable personal income:
getSymbols("DSPIC96", src = "FRED")
```

```
dpi <- as.data.frame(DSPIC96)
dpi_ts <- ts(dpi$DSPIC96, start = 1959, freq = 12)
# Growth rates:
log.dpi <- log(dpi_ts)
r_dpi <- diff(log.dpi)

# Plot
quartz(width = 11, height = 8.5)
plot(r_dpi, col = "darkblue", ylab = "Growth rate")
lines(r_pce, col = "darkred")
# More fluctuation in disposable personal income than that of
# real consumption

# 3b. Regress consumption growth on disposable income growth
model <- lm(r_pce ~ r_dpi)
summary(model)

# 3c. Add a lag of growth in disposable income to the
# equation
lag.r_dpi <- append(NA, r_dpi[-length(r_dpi)])
model2 <- lm(r_pce ~ r_dpi + lag.r_dpi)
summary(model2)

# Problem 4 (3.3) Download the following data: US Real GDP,
# Exchange rate of the Japanese Yen against the US Dollar, 10
# year US treasury constant maturity yield, US unemployment
# GDP:
getSymbols("GDPC1", src = "FRED")  #Starts 1947-01-01 - 2016-10-01, quarterly data
gdp <- ts(GDPC1$GDPC1, start = 1947, freq = 4)
plot(gdp, ylab = "GDP (Billions of Chained Dollars)", main = "GDP")
abline(h = mean(gdp), col = "gray", lty = 2)

# EXPJ
getSymbols("EXJPUS", src = "FRED")  #Starts 1971-01-01 - 2017-04-01, monthly data
exj <- ts(EXJPUS$EXJPUS, start = 1971, freq = 12)
plot(exj, ylab = "Exchange rate", main = "Exchange Rate of Yen against Dollar")
abline(h = mean(exj), col = "gray", lty = 2)

# GS10
getSymbols("GS10", src = "FRED")  #Starts 1953-04-01 - 2017-03-01, monthly data
gs10 <- ts(GS10$GS10, start = 1953 + (1/4), freq = 12)
plot(gs10, ylab = "Yield")
abline(h = mean(gs10), col = "gray", lty = 2)

# Unemployment Rate
getSymbols("UNRATE", src = "FRED")  #Starts 1948-01-01 - 2017-03-01
unemploy <- ts(UNRATE$UNRATE, start = 1948, freq = 12)
plot(unemploy, ylab = "Unemployment Rate", main = "US Unemployment Rate")
```

```
abline(h = mean(unemploy), col = "gray", lty = 2)

# Problem 5 (4.3) Quarterly freq. housing prices and interest
# rates Compute ACF/PACF of quarterly house prices, interest
# rates, house price growth, and interest rate changes
m_q <- read.csv("~/Desktop/Mortgage_quarterly.csv")
names(m_q) <- c("time", "price", "rates")
r_price_q <- diff(log(m_q$price))
r_rates_q <- diff(log(m_q$rates))

# Price:
par(mfrow = c(2, 1))
acf(m_q$price)
pacf(m_q$price)

# Interest rate:
par(mfrow = c(2, 1))
acf(m_q$rates)
pacf(m_q$rates)

# Price growth:
par(mfrow = c(2, 1))
acf(r_price_q)
pacf(r_price_q)

# Interest rate changes:
par(mfrow = c(2, 1))
acf(r_rates_q)
pacf(r_rates_q)

# Comment on differences across autocorrelation functions
# Which series do you find stornger time dependence

# Examine differences in autocorrelation functions of
# quarterly data v. annual data Annual analysis:
m <- read.csv("~/Documents/Econ 144/Data Sets/HW 1/Mortgage.csv")
head(m)
names(m) <- c("year", "price", "rates")
# Growth:
r_price <- diff(log(m$price))
r_rates <- diff(log(m$rates))

# Price:
par(mfrow = c(2, 1))
acf(m$price)
pacf(m$price)

# Interest rate:
```

```
par(mfrow = c(2, 1))
acf(m$rates)
pacf(m$rates)

# Price growth:
par(mfrow = c(2, 1))
acf(r_price)
pacf(r_price)

# Interest rate changes:
par(mfrow = c(2, 1))
acf(r_rates)
pacf(r_rates)
```