Project 2

Siyi Xie

For this project, we will use the `forecast` and `tseries` libraries.

```
library("forecast")
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package: timeDate
## This is forecast 5.9
```

```
library("tseries")
```

The dataset is the exchange rate for the Japanese Yen to 1 U.S. Dollar. The data is daily from January 1st, 2004 to April 30th, 2015.

The source of the data is http://www.oanda.com/currency/historical-rates/ (http://www.oanda.com/currency/historical-rates/).

```
data <- read.csv("e:/Forecasting/jpy.csv")
date <- as.Date(data$date, format="%m/%d/%Y")
time <- 1:length(date)
yen <- data$jpy
```

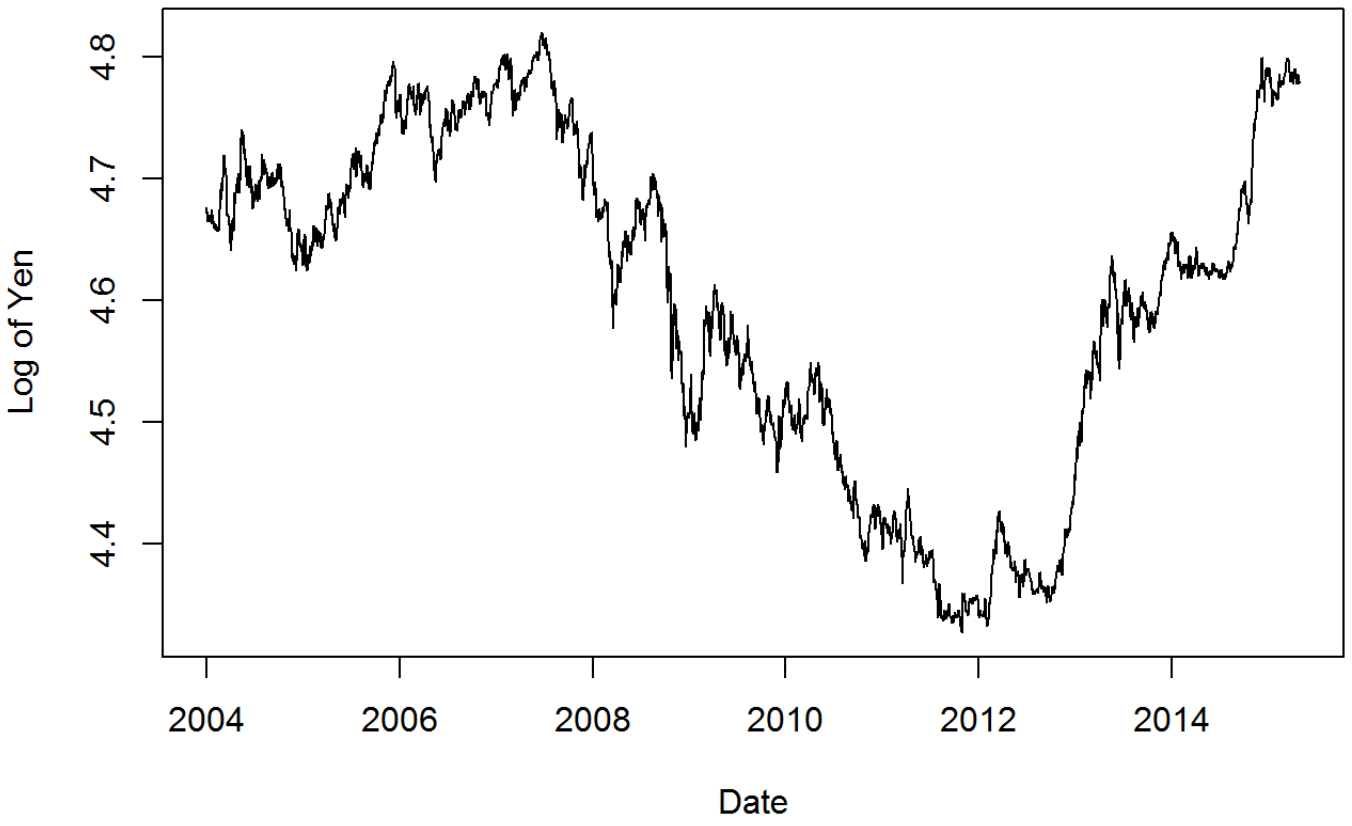
We will work with the logs of the exchange rates:

```
log.yen <- log(yen)
```

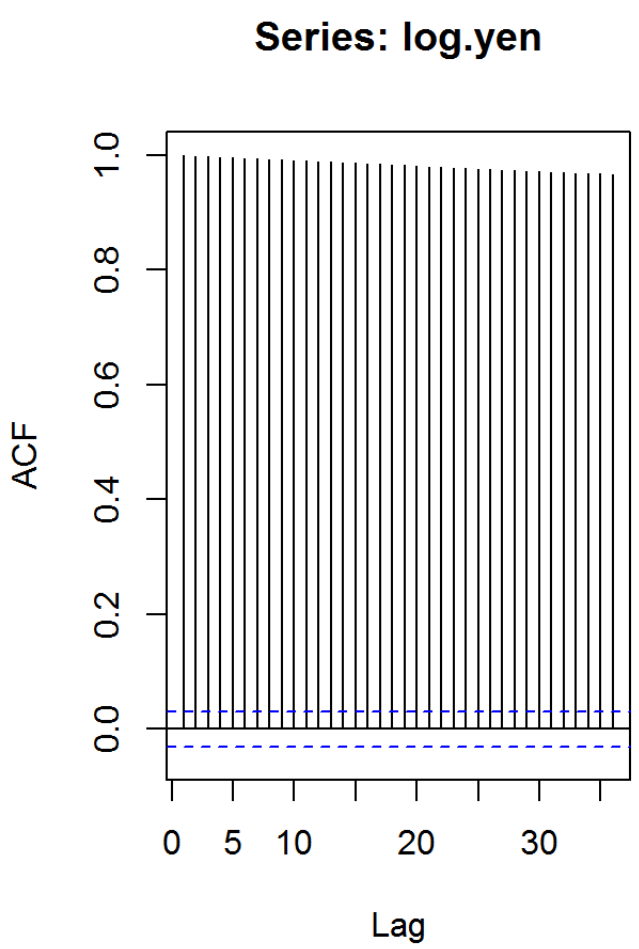
Problem 1

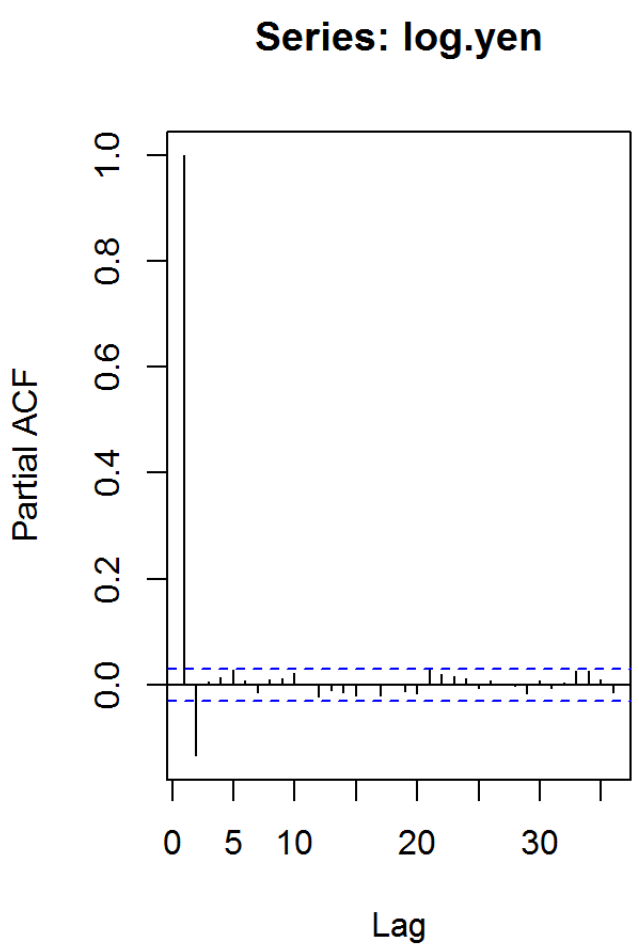
Here is a time series plot of log.yen, along with the ACF and PACF:

```
plot(date, log.yen, type="l", xlab="Date", ylab="Log of Yen")
```



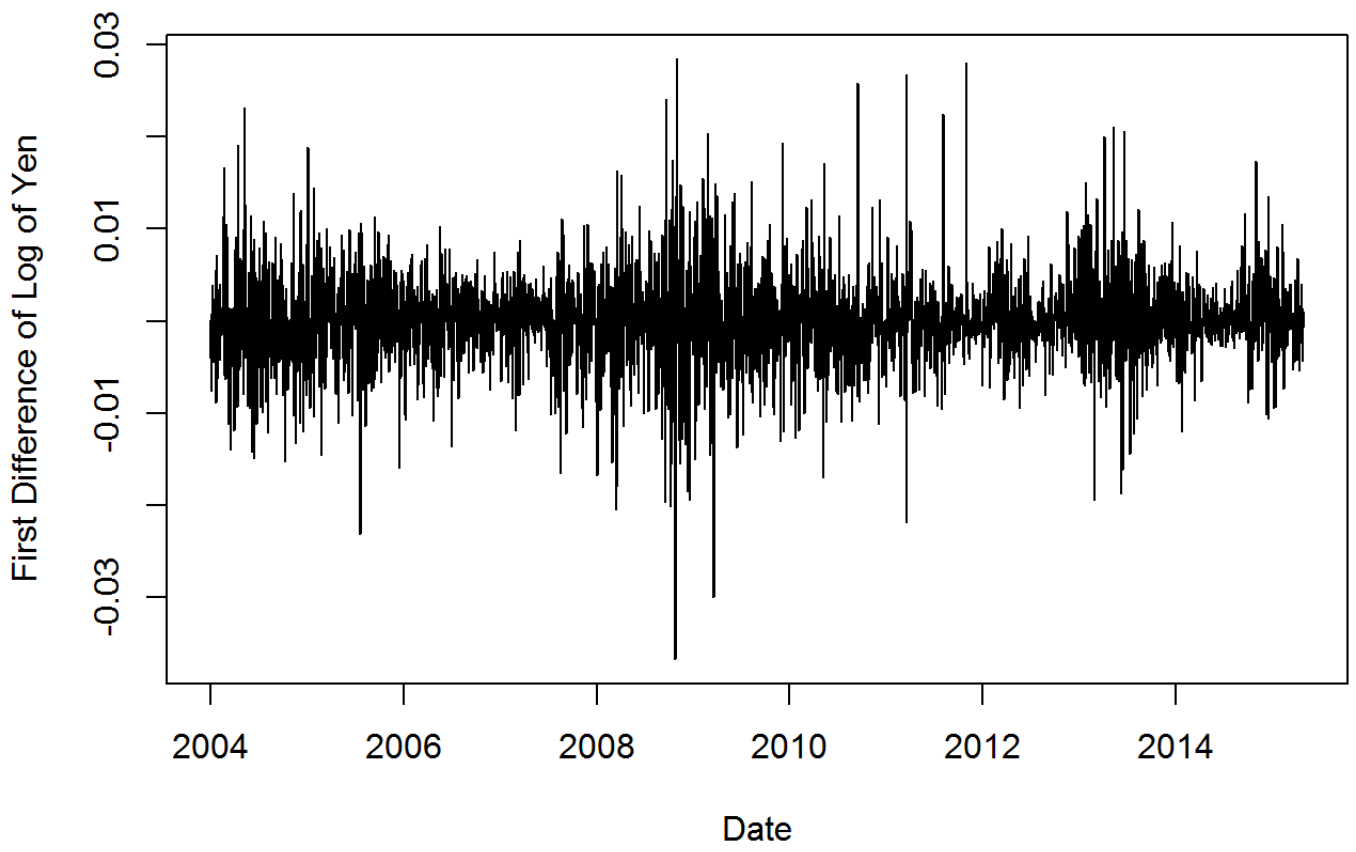

```
par(mfrow=c(1,2))
Acf(log.yen)
Pacf(log.yen)
```



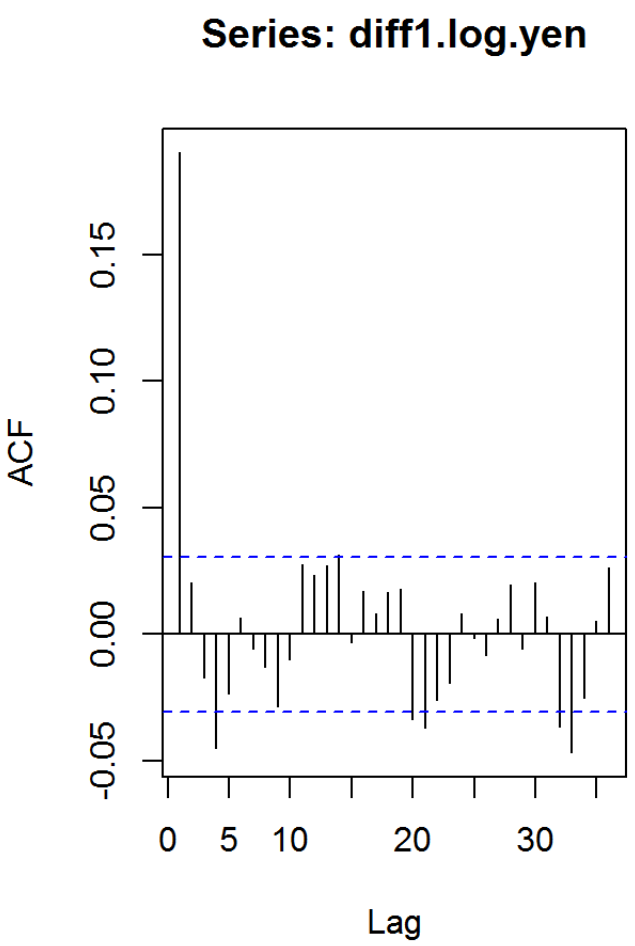


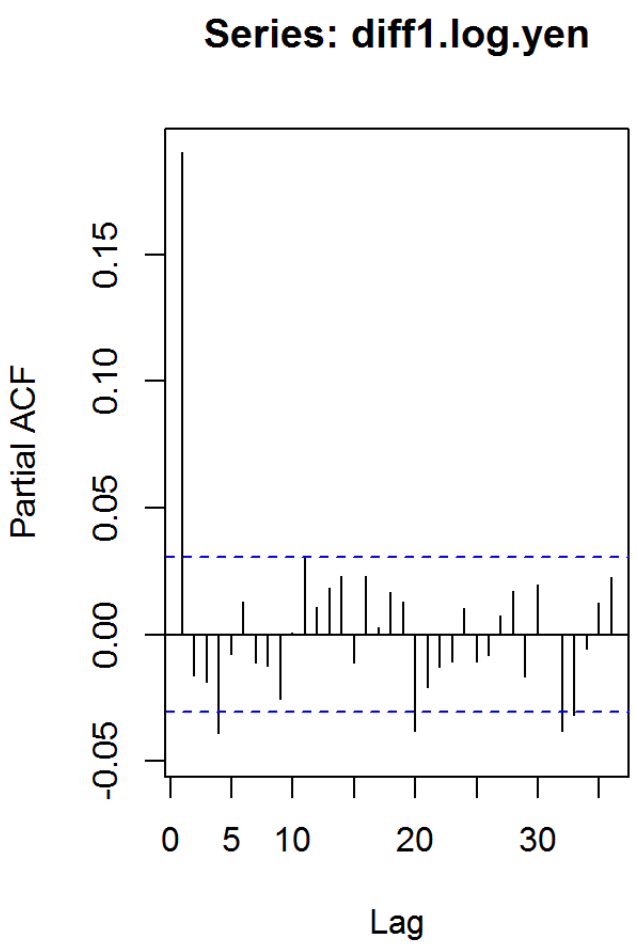
Here is the first difference, along with the ACF and PACF:

```
diff1.log.yen <- c(NA, diff(log.yen))
plot(date, diff1.log.yen,type="l",xlab="Date", ylab="First Difference of Log of Yen")
```



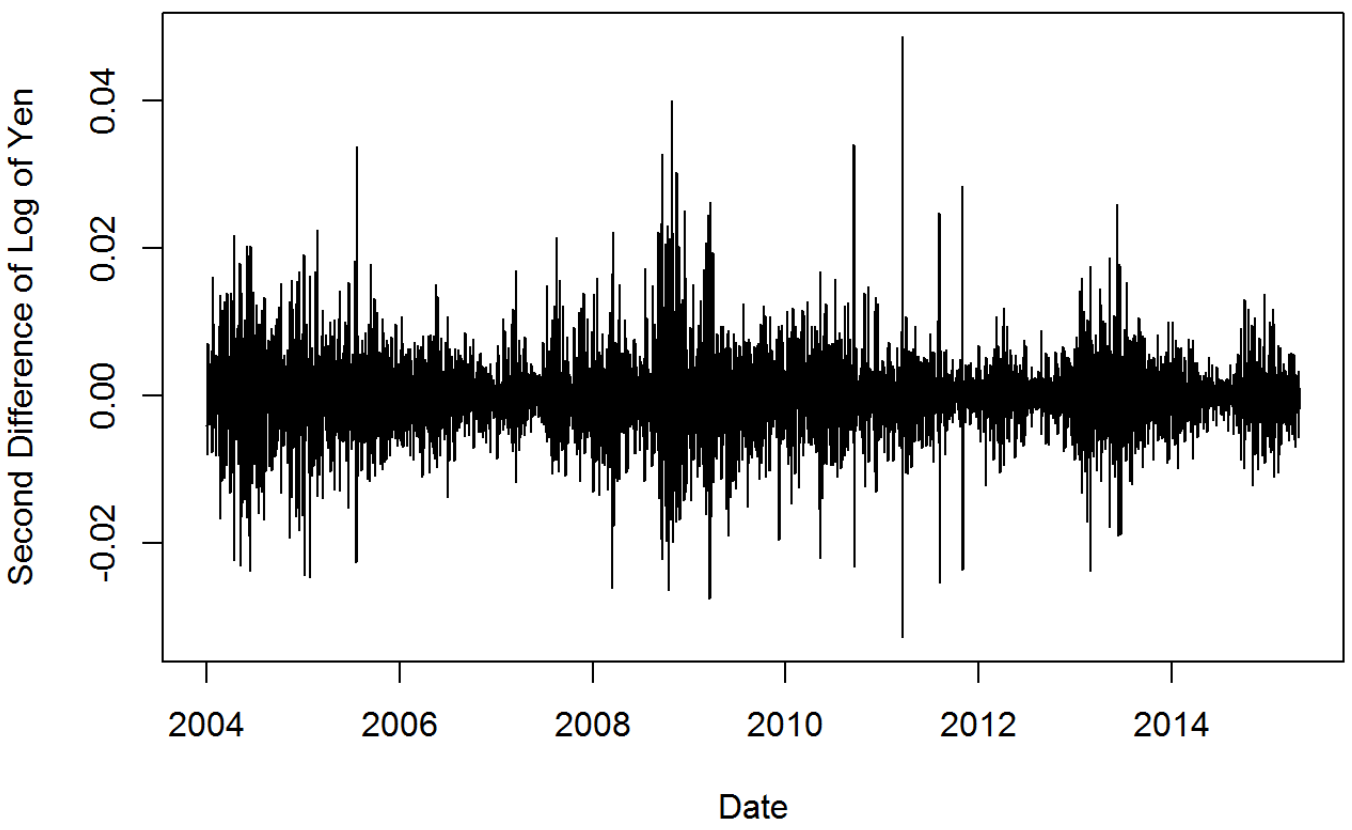

```
par(mfrow=c(1,2))
Acf(diff1.log.yen)
Pacf(diff1.log.yen)
```



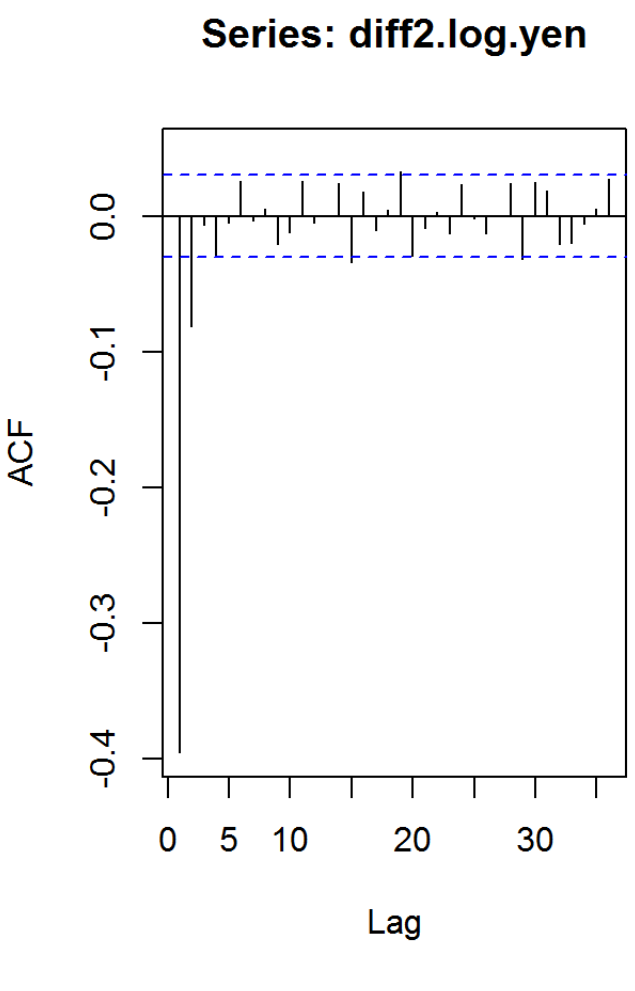


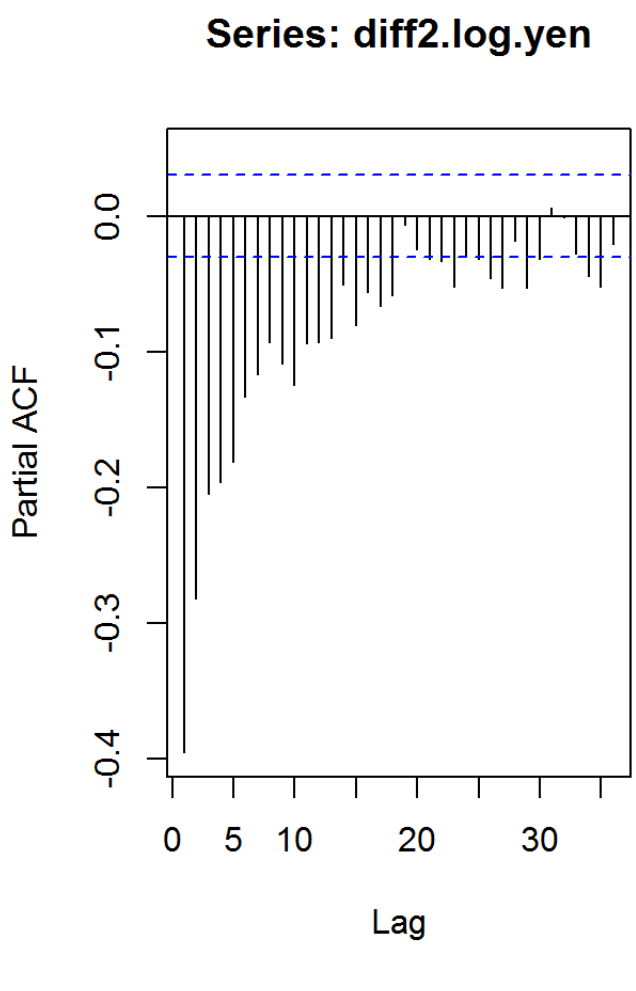
Here is the second difference, along with the ACF and PACF:

```
diff2.log.yen <- c(NA, diff(diff1.log.yen))
plot(date, diff2.log.yen,type="l",xlab="Date", ylab="Second Difference of Log of Yen")
```




```
par(mfrow=c(1,2))
Acf(diff2.log.yen)
Pacf(diff2.log.yen)
```





Does the series appear to be stationary?

The ACF plot of log of Yen shows a very slow decay pattern which is typical of a nonstationary time series. Clearly at least one order of differencing is needed to stationarize this series.

The series of the first difference of log of Yen appears approximately stationary with no long-term trend. Its ACF and PACF plots cut off after lag 1 although there are several spikes out of the bounds but not significantly.

As to the second difference of log of Yen, it looks overdifferenced because the ACF plot has a negative spike at lag 1 that is close to -0.5.

Can you identify an ARIMA(p,d,q) model from these plots?

Since the series of the first difference of log of Yen seems stationary, We decide to select 1 for d in this case. However, we cannot identify p and q for now. We need to use AICc to select p and q.

Problem 2

Since the first difference of the log of Yen does not have a non-zero average trend, there would not be a constant in the model. So we will only consider the ARIMA models without a constant here.

Here are the AICc for all ARIMA(p,1,q) models without constant for p and q ranging from 0 to 2:

```
d <- 1
for (include.constant in c(FALSE)) {
    for (p in 0:2) {
        for (q in 0:2) {
            fit <- Arima(diff1.log.yen, c(p,0,q),
                         include.constant=include.constant, method="ML")
            cat("ARIMA",
                "(", p, ",", d, ",", q, ")",
                "(constant=", include.constant, ")",
                " : ", fit$aicc, "\n", sep="")
        }
    }
}
```

```
## ARIMA(0,1,0)(constant=FALSE) : -32827.41
## ARIMA(0,1,1)(constant=FALSE) : -32976.49
## ARIMA(0,1,2)(constant=FALSE) : -32976.99
## ARIMA(1,1,0)(constant=FALSE) : -32977.85
## ARIMA(1,1,1)(constant=FALSE) : -32976.78
## ARIMA(1,1,2)(constant=FALSE) : -32974.85
## ARIMA(2,1,0)(constant=FALSE) : -32976.94
## ARIMA(2,1,1)(constant=FALSE) : -32974.74
## ARIMA(2,1,2)(constant=FALSE) : -32978.48
```

Select an ARIMA(p,1,q) model.

since the AICc of ARIMA(2,1,2) without constant is the smallest, I select this model here.

Here is code to fit the model, then compute residuals and the fitted values:

```
fit.mean <- Arima(log.yen, c(2,1,2), include.constant=FALSE)
```

Here are the residuals, with the last 10 residuals printed out:

```
resid <- residuals(fit.mean)
tail(resid, n=10)
```

```
##  [1]  0.0011234968  0.0036899292  0.0008623689  0.0011438917 -0.0044864987
##  [6] -0.0023057068  0.0006977118  0.0010923523 -0.0009475261 -0.0003821682
```

Here are the fitted values, with the last 10 fitted values printed out:

```
f <- fitted.values(fit.mean)
tail(f, n=10)
```

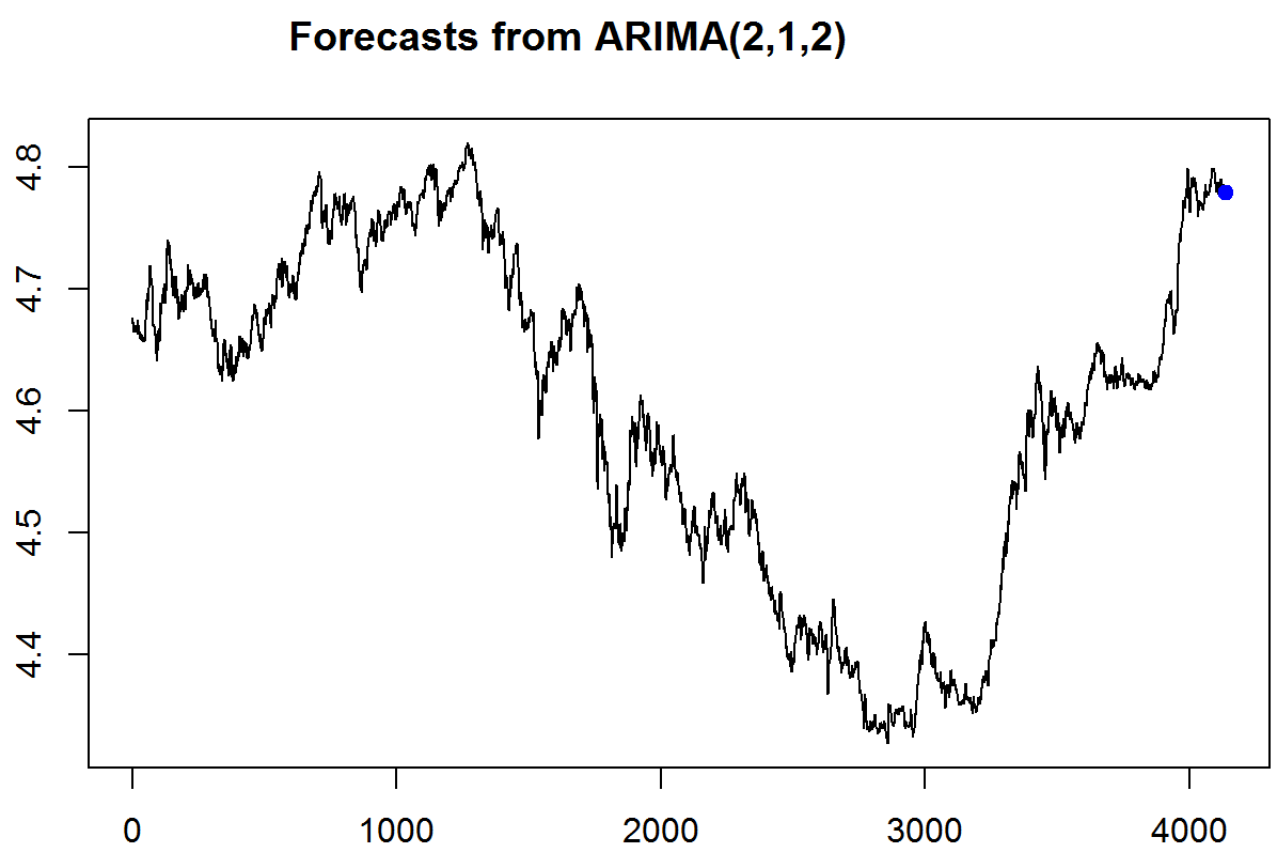
```
##  [1] 4.777916 4.779375 4.783876 4.785014 4.786295 4.780841 4.777837
##  [8] 4.778619 4.780071 4.779085
```

Here is the one step ahead forecast and 95% forecast interval:

```
forecast(fit.mean,h=1,level=95)
```

```
##      Point Forecast    Lo 95    Hi 95
## 4139       4.778641 4.769842 4.787441
```

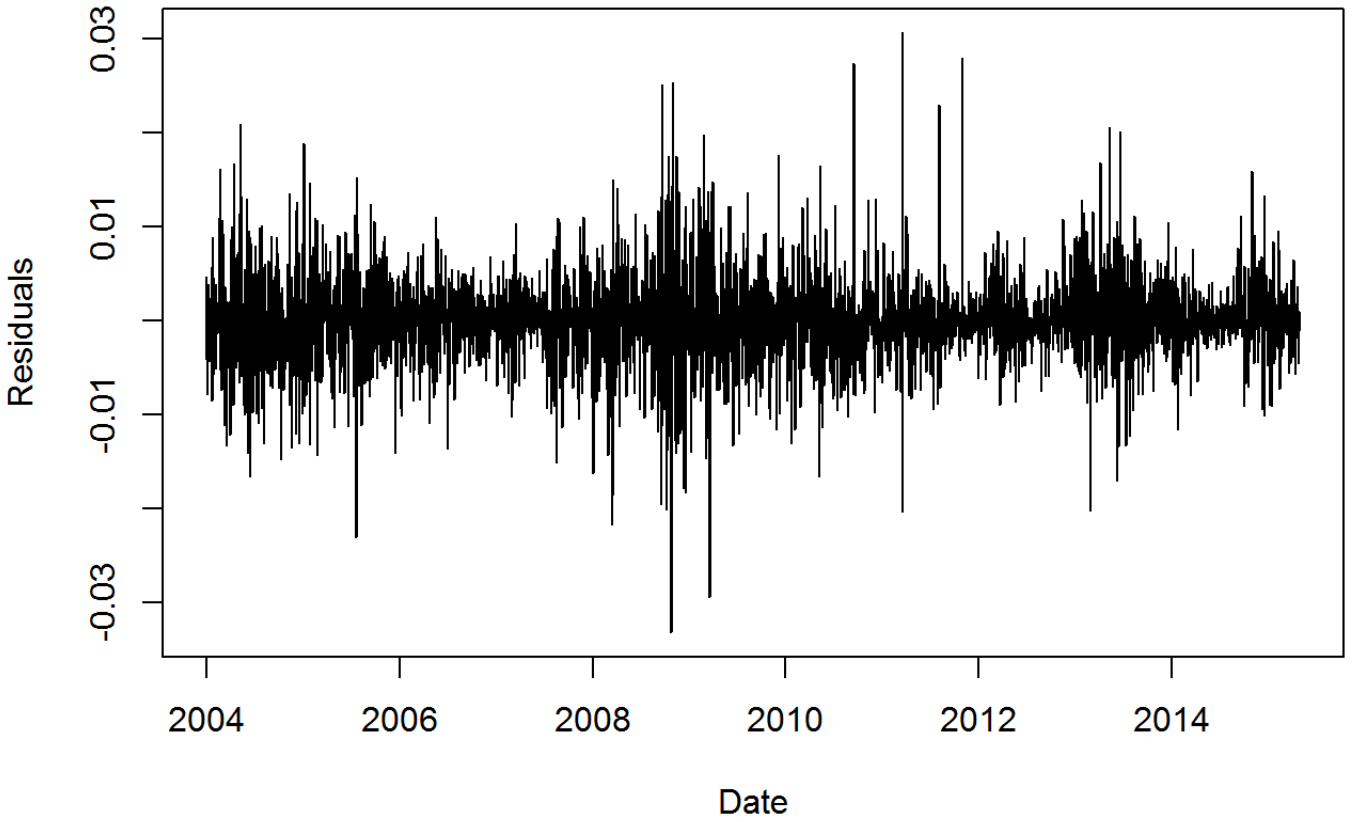
```
plot(forecast(fit.mean,h=1,level=95))
```



Problem 3

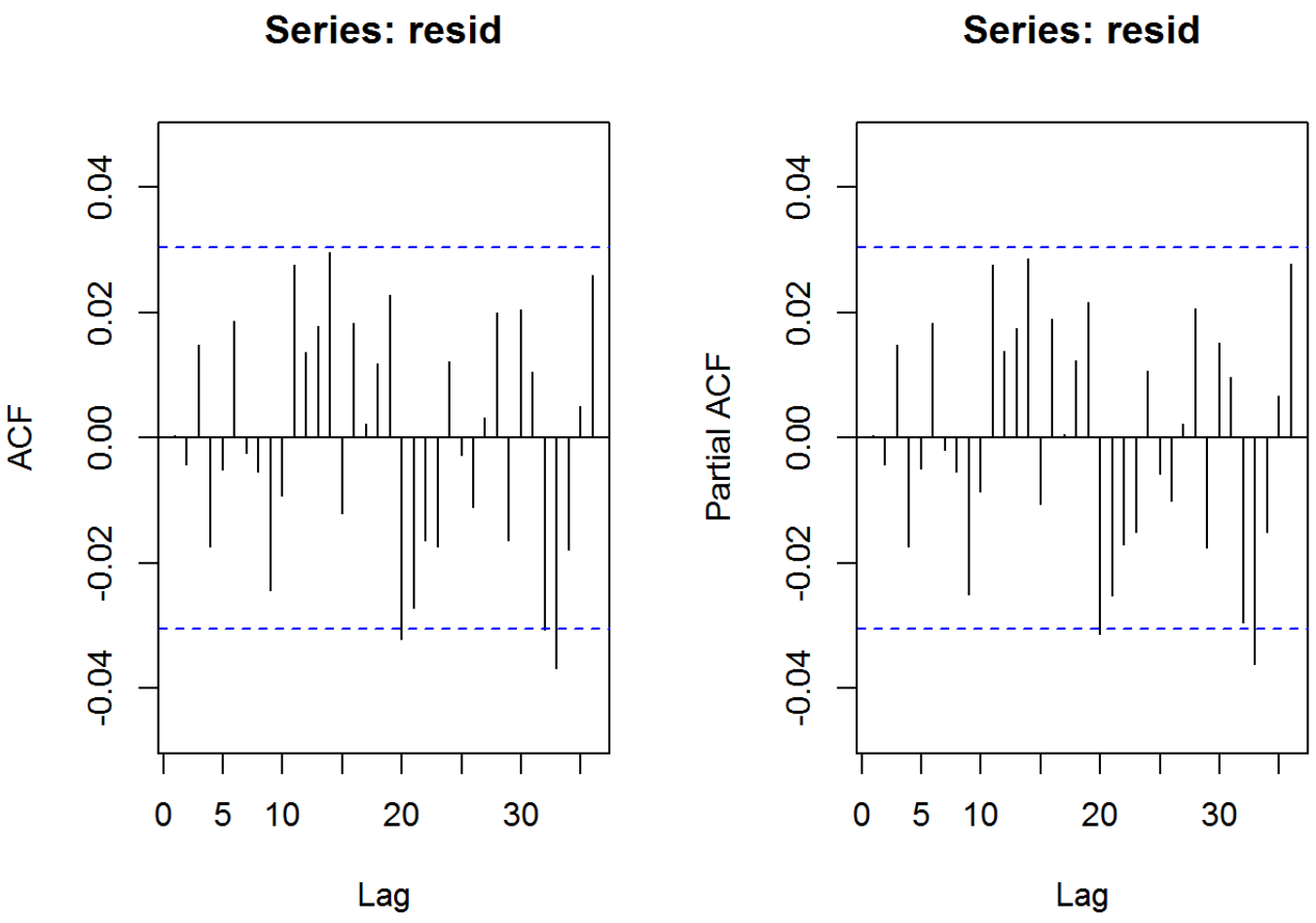
Here is a plot of the residuals:

```
plot(date, resid, type="l", xlab="Date", ylab="Residuals")
```



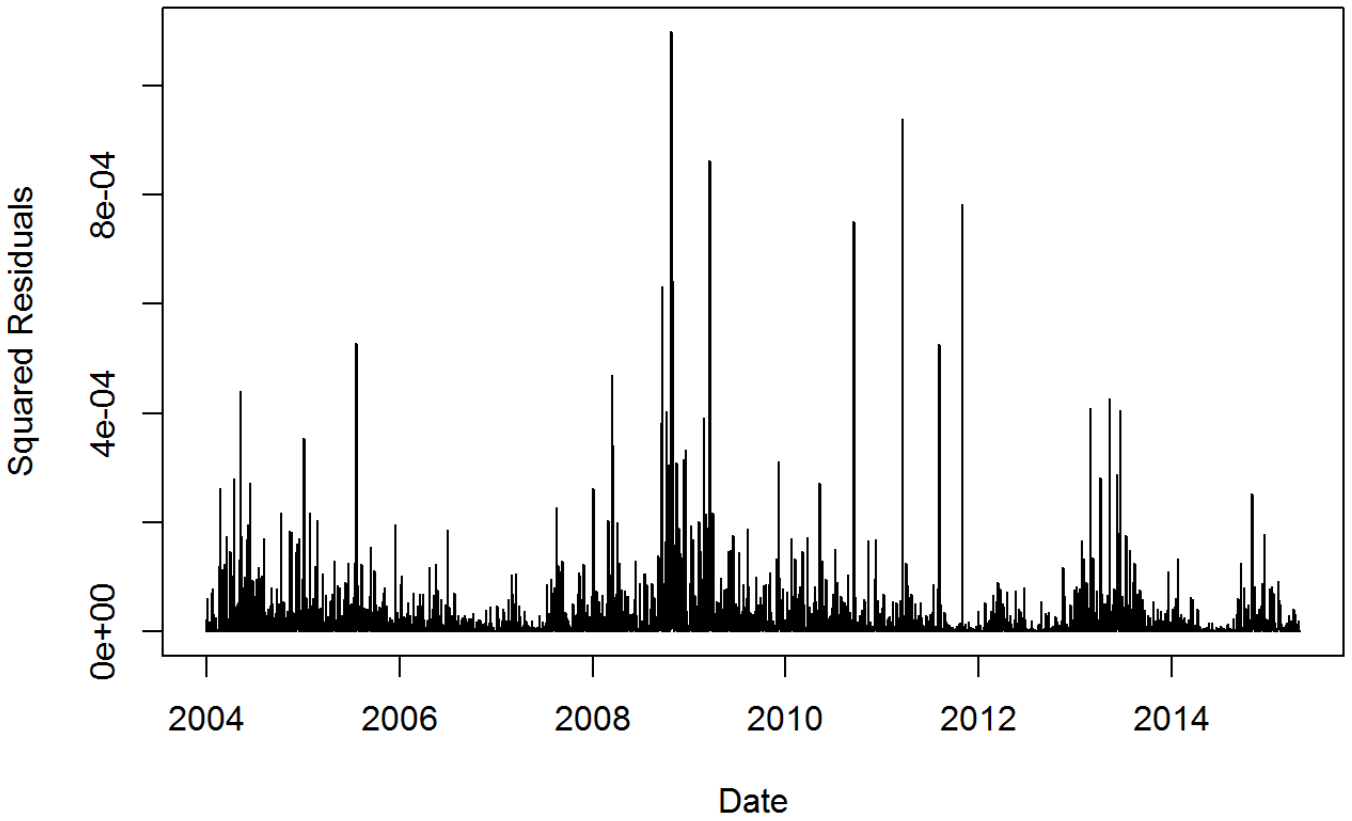
Here are an ACF and PACF of the residuals:

```
par(mfrow=c(1,2))
Acf(resid)
Pacf(resid)
```

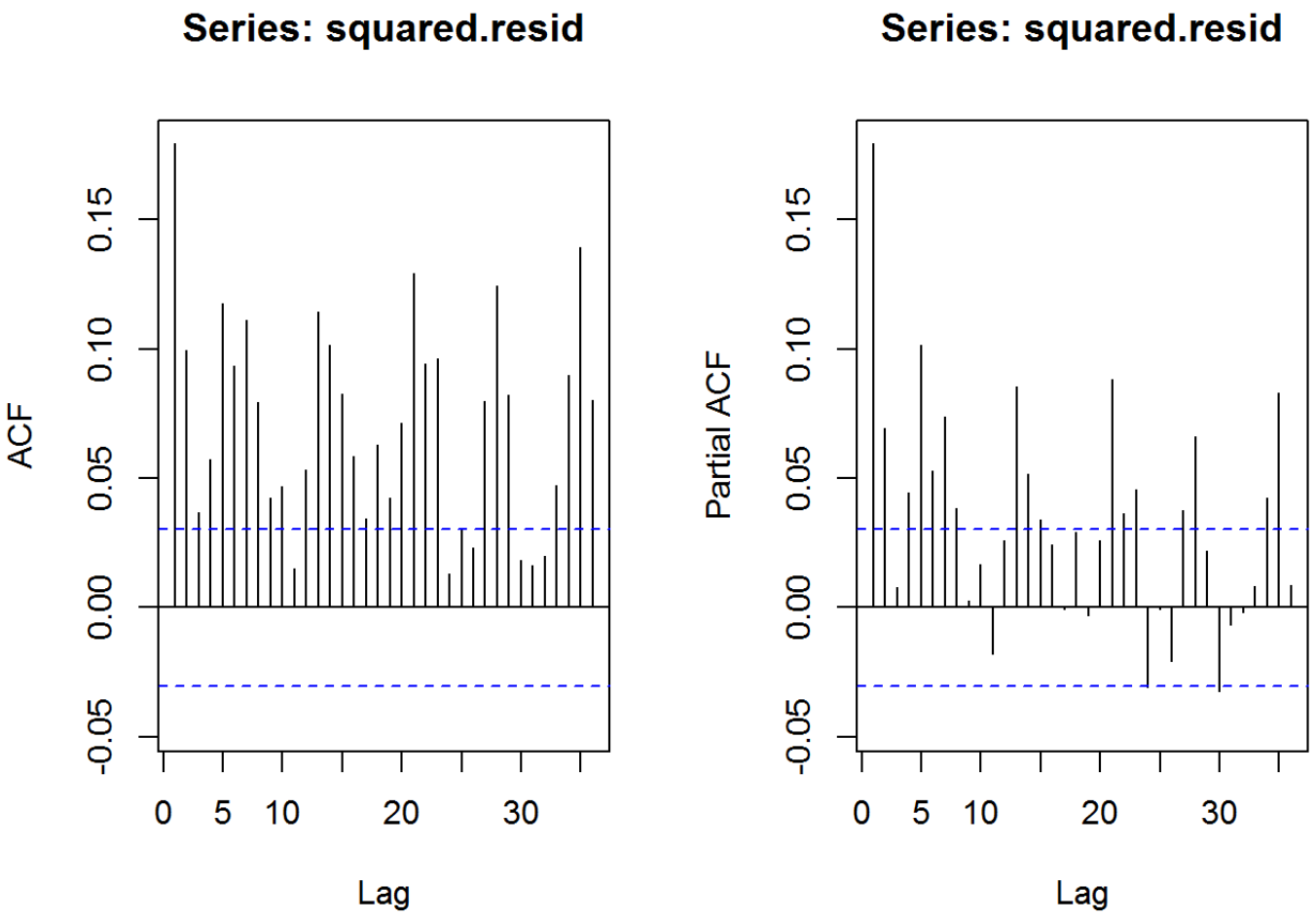


Here are an ACF and PACF of the squared residuals:

```
squared.resid <- resid^2
plot(date, squared.resid, type="l", xlab="Date", ylab="Squared Residuals")
```




```
par(mfrow=c(1,2))
Acf(squared.resid)
Pacf(squared.resid)
```



Use these plots to argue that the residuals, although approximately uncorrelated, are not independent; instead, they show evidence of conditional heteroscedasticity.

From the ACF of the residuals, we can find that the ACF values are almost all close to zero. This means that the residuals are approximately uncorrelated.

As to the ACF of the squared residuals, there are many spikes out of the bounds so that there are auto correlation in the squared residuals, which means that there is conditional heteroscedasticity and the residuals are not independent.

Problem 4

Here are the AICc values for the ARCH(q):

```
q <- 0:10
loglik <- rep(NA, length(q))
N <- length(resid)

for (i in 1:length(q)) {
    if (q[i] == 0) {
        loglik[i] <- -0.5 * N * (1 + log(2 * pi * mean(resid^2)))
    } else {
        fit <- garch(resid, c(0,q[i]), trace=FALSE)
        loglik[i] <- logLik(fit)
    }
}

k <- q + 1
aicc <- -2 * loglik  + 2 * k * N / (N - k - 1)

print(data.frame(q, loglik, aicc))
```

```
##     q   loglik      aicc
## 1   0 16498.24 -32994.48
## 2   1 16654.36 -33304.72
## 3   2 16665.00 -33324.00
## 4   3 16665.73 -33323.45
## 5   4 16662.82 -33315.63
## 6   5 16686.54 -33361.06
## 7   6 16742.96 -33471.90
## 8   7 16803.22 -33590.41
## 9   8 16828.95 -33639.86
## 10  9 16817.55 -33615.05
## 11 10 16816.68 -33611.30
```

Here is the AICc for the GARCH(1,1):

```
fit <- garch(resid, c(1,1), trace=FALSE)
loglik <- logLik(fit)
k <- 2
aicc <- -2 * loglik  + 2 * k * N / (N - k - 1)

print(data.frame(loglik, aicc))
```

```
##     loglik     aicc
## 1 16805.95 -33607.9
```

Here are the summary and log likelihood of the selected model:

```
fit.var <- garch(resid, order = c(0,8),trace=FALSE)
summary(fit.var)
```

```
##
## Call:
## garch(x = resid, order = c(0, 8), trace = FALSE)
##
## Model:
## GARCH(0,8)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -4.8750205 -0.4813876  0.0008636  0.4885874 10.5829462
##
## Coefficient(s):
##     Estimate  Std. Error  t value Pr(>|t|)
## a0 6.484e-06   1.253e-07   51.731  < 2e-16 ***
## a1 2.024e-01   1.393e-02   14.530  < 2e-16 ***
## a2 3.244e-02   5.969e-03    5.435 5.48e-08 ***
## a3 5.880e-03   5.449e-03    1.079   0.2805
## a4 1.044e-02   5.027e-03    2.077   0.0378 *
## a5 4.728e-02   8.213e-03    5.757 8.54e-09 ***
## a6 1.141e-01   1.375e-02    8.302  < 2e-16 ***
## a7 2.398e-01   1.942e-02   12.350  < 2e-16 ***
## a8 1.100e-01   1.187e-02    9.266  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostic Tests:
##  Jarque Bera Test
##
## data:  Residuals
## X-squared = 5230.592, df = 2, p-value < 2.2e-16
##
##
##  Box-Ljung test
##
## data:  Squared.Residuals
## X-squared = 0.0554, df = 1, p-value = 0.8139
```

```
logLik(fit.var)
```

```
## 'log Lik.' 16828.95 (df=9)
```

Comment on the statistical significance of the parameter values of your selected model.

We can find that the value of a3 is not statistically significant since the P value of it is 0.2805 which is significantly greater than 0.05. The value of a4 is statistically significant but its degree is weak as its p value is 0.0378 which is very near 0.05. However, other parameters are all statistically significant because their p values are obviously smaller than 0.05.

Write the complete form of the ARCH or GARCH model you have selected.

The ARCH(8) modle is:

h_t = 6.484e-06 + 0.2024*(e_t-1)*^2 + *0.03244*(e_t-2)^2 + 0.00588*(e_t-3)*^2 + *0.01044*(e_t-4)^2 + 0.04728*(e_t-5)*^2 + *0.1141*(e_t-6)^2 + 0.2398*(e_t-7)*^2 + *0.11*(e_t-8)^2

Problem 5

Construct a 95%one step ahead forecast interval for the log exchange rate, based on your ARIMA-ARCH model.

Based on the calculations in other problems, we can get relevant numbers to compute the conditional variance:

We have known the most recent residuals:

0.0011234968 0.0036899292 0.0008623689 0.0011438917 -0.0044864987 -0.0023057068 0.0006977118 0.0010923523

```
f1 <- 4.778641
w = 6.484e-06
a1 = 2.024e-01
a2 = 3.244e-02
a3 = 5.880e-03
a4 = 1.044e-02
a5 = 4.728e-02
a6 = 1.141e-01
a7 = 2.398e-01
a8 = 1.100e-01
h1 <- w+a1*(0.0011234968)^2+a2*(0.0036899292)^2+a3*(0.0008623689)^2+a4*(0.0011438917)^2+a5*(-0.0044864987)^2+a6*(-0.0023057068)
^2+a7*(0.0006977118)^2+a8*(0.0010923523)^2
f1 + c(-1.96, 1.96) * sqrt(h1)
```

```
## [1] 4.772759 4.784523
```

Compare this to the interval based on the ARIMA only model from Problem 2.

According to the data I have collected, the last observation, which is one step ahead of the data set used in this project, is 4.780047 which is included in both the forecast intervals of the ARIMA model and the ARCH model.

This ARCH forecast interval is [4.772759, 4.784523] while the forecast interval based on the ARIMA only model is [4.769842, 4.787441]. It seems that the interval based on ARCH(8) model is a more narrow than that based on ARIMA only model. This is because the variance in the ARIMA only model is constant while the variance is the ARCH model is not constant and the variance at this time point is relatively small.

Problem 6

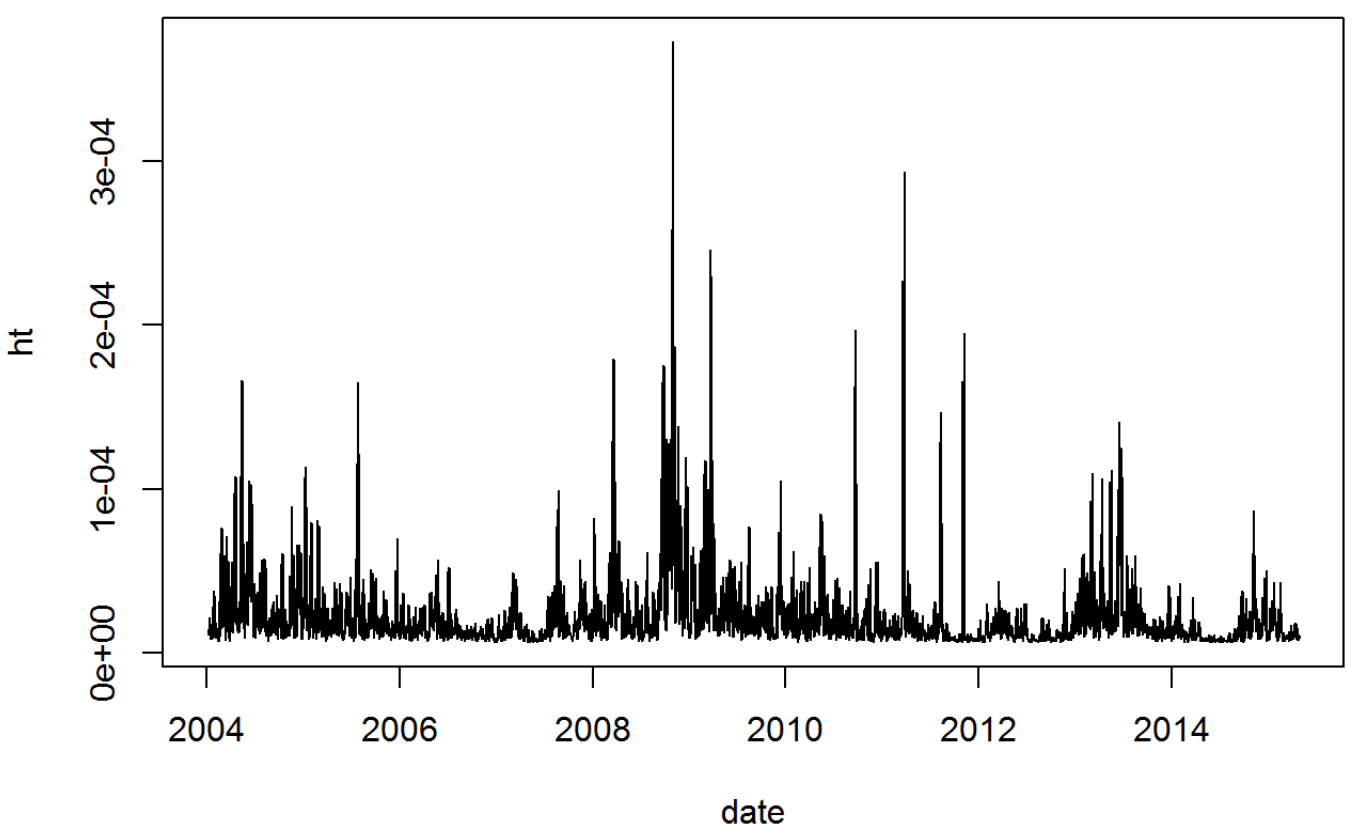
Here are the conditional variances, with the last 10 values printed out:

```
ht <- fit.var$fit[,1]^2
tail(ht, n=10)
```

```
##  [1] 1.114117e-05 1.521007e-05 1.374845e-05 8.318124e-06 7.726839e-06
##  [6] 1.110458e-05 9.056556e-06 8.779145e-06 1.053365e-05 9.539902e-06
```

Here is a plot of the conditional variances:

```
plot(date,ht,type="l")
```



Use this plot to locate bursts of high volatility.

Based on this plot of the conditional variances, we can see that the most significant burst of high volatility which is around 0.0003 happened in the second half of 2008. There are also some obvious bursts of high volatility in the beginning of 2009, the first half of 2011, the second half of 2010, the end of 2011, the first half of 2008, etc.

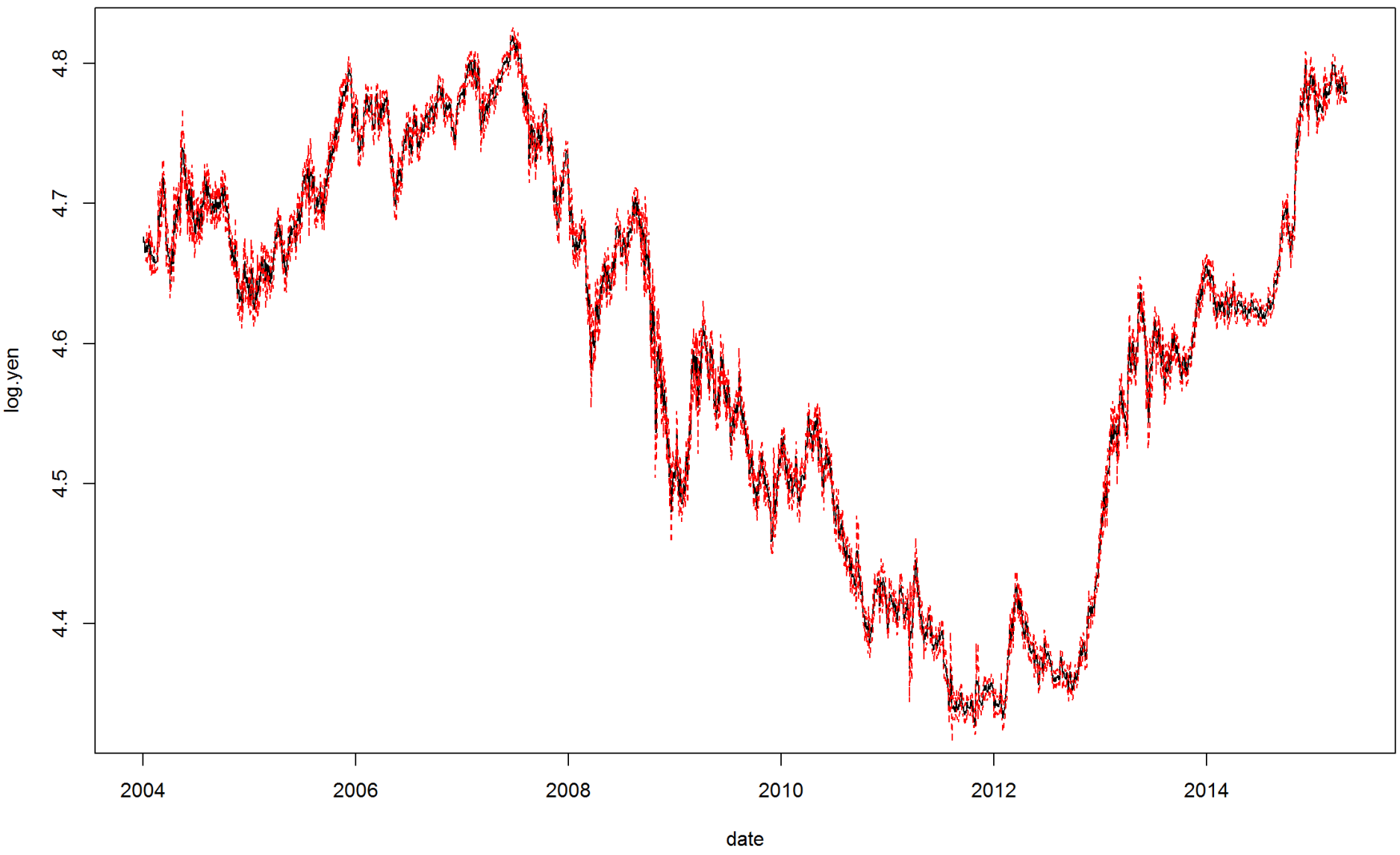
Do these highly volatile periods agree with those found from examination of the time series plot of the log exchange rates themselves?

Yes, we could find that the values in the plot of the log exchange rates look highly volatile during these periods we identified from the plot of the conditional variances.

Problem 7

Here is a time series plot which simultaneously shows the log exchange rates, together with the ARIMA-ARCH one-step-ahead 95% forecast intervals based on information available in the previous day:

```
plot(date, log.yen, type="l")
lines(date, f + 1.96 * sqrt(ht), lty=2, col=2)
lines(date, f - 1.96 * sqrt(ht), lty=2, col=2)
```



Using this plot, comment on the accuracy and practical usefulness of the forecast intervals.

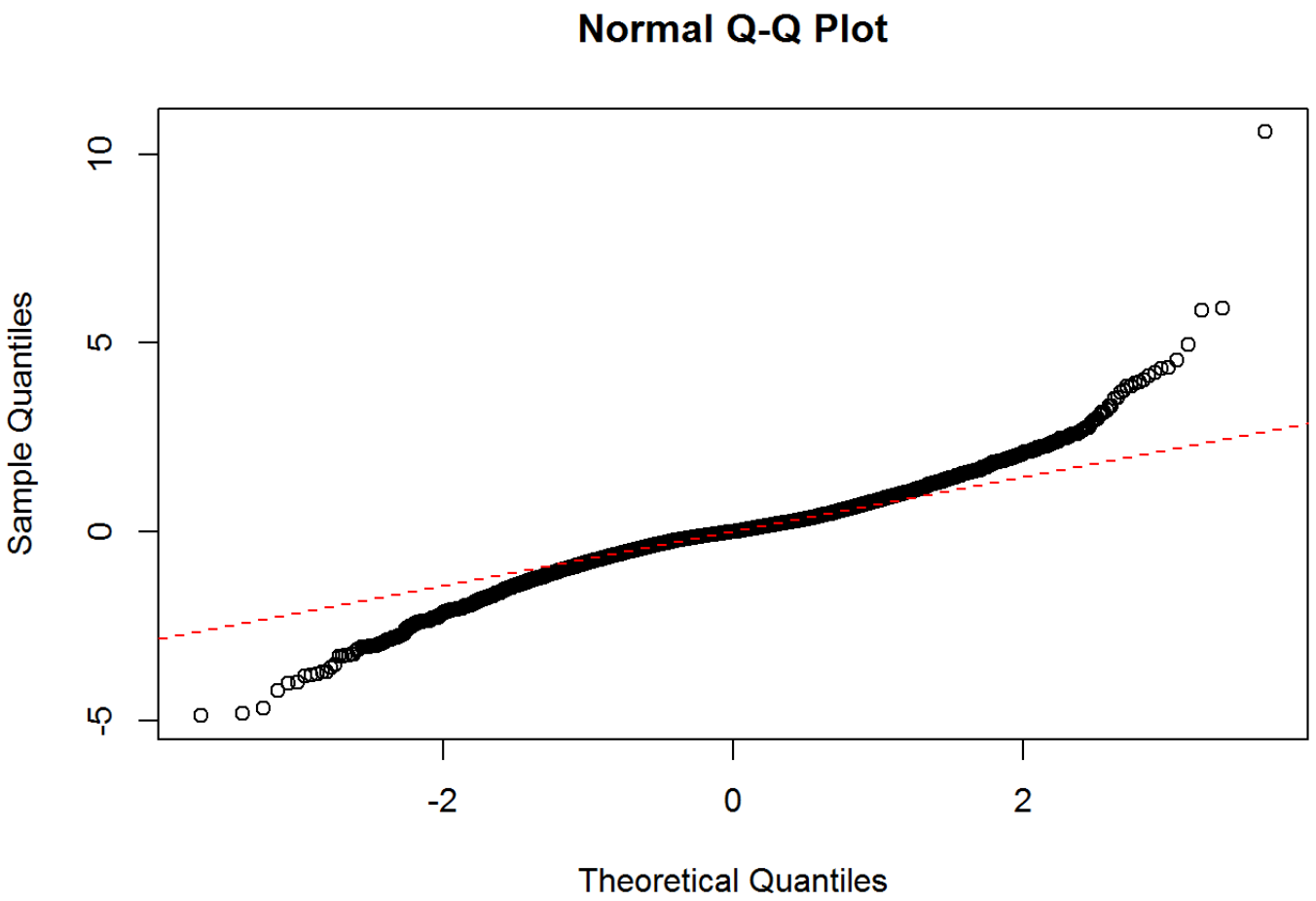
From this plot, we can see that the forecast intervals seem quite accurate since the actual data is almost all within the forecast intervals. Through using the forecast intervals, we can have better estimates of the intervals for the future data and have a better understanding for the time-varying volatility. We can have clear vision of the periods in which the volatility is high or low.

The advantage of ARCH models is the ability to describe the time-varying stochastic conditional volatility, which can be used to improve the reliability of interval forecasts and help us in understanding the process and making better forecast for time series data in the real world. Time-varying volatility has important implications for financial risk management, asset allocation, asset pricing, etc.

Problem 8

Here is a normal probability plot of the ARCH residuals.

```
resid.arch <- resid/sqrt(ht)
qqnorm(resid.arch, col=1)
qqline(resid.arch, col=2, lty=2)
```



Does the model seem to have adequately described the leptokurtosis (“long-tailedness”) in the data?

It seems that the model has not adequately described the leptokurtosis in the data. Because the normal probability plot shows that the residuals are not quite normally distributed.

Problem 9

Here is a count of how many prediction interval failures there were:

```
sum(abs(resid.arch) > 1.96, na.rm=TRUE)
```

```
## [1] 252
```

The number of prediction intervals is:

```
sum(!is.na(resid.arch))
```

```
## [1] 4130
```

**What percentage of the time did the intervals fail?

According to the calculations above, the number of the forecast intervals faild is 252 and the number of the intervals is 4130. Thus, the percentage of the time that the intervals failed is 6.1%.