36-618 HW5

Daniel Nason

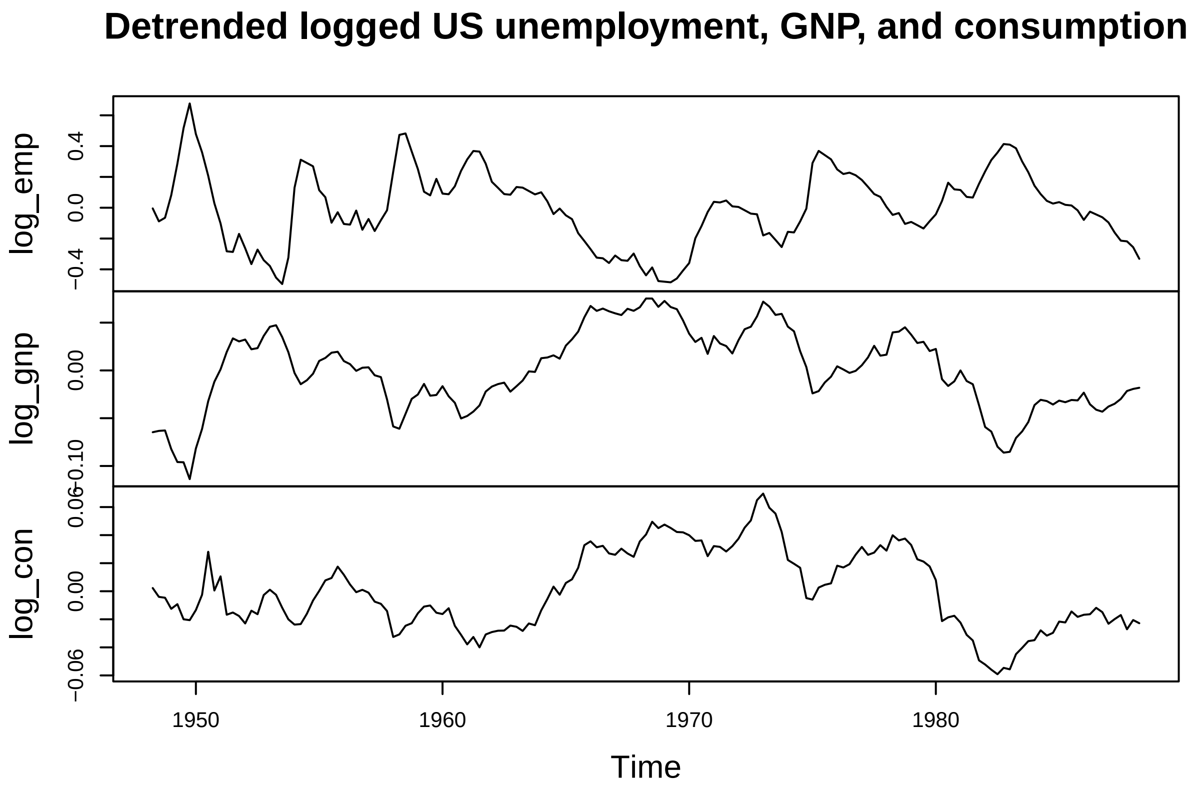
4/4/2022

```
setwd("C:/Users/Owner/CMU/Spring/36-618/HW/HW5")
library(forecast)
library(astsa)
library(vars)
library(fGarch)
library(tidyverse)
data <- ts(econ5, start = c(1948,2), frequency = 4)
```

Question 1

a)

```
# detrending the data
log_emp_lm <- lm(log(data[,1]) ~ time(data), na.action = NULL)
log_gnp_lm <- lm(log(data[,2]) ~ time(data), na.action = NULL)
log_con_lm <- lm(log(data[,3]) ~ time(data), na.action = NULL)
log_emp <- resid(log_emp_lm)
log_gnp <- resid(log_gnp_lm)
log_con <- resid(log_con_lm)
log_data <- cbind(log_emp, log_gnp, log_con)
plot(log_data, main = 'Detrended logged US unemployment, GNP, and consumption')
```



The time series seem to all be related in that there is autocorrelation present among the detrended time series and there is a cyclical pattern that is affecting all of the trends. Specifically, the unemployment and GNP are like reflections of each other in that they trend in opposite directions over time. Consumption is not a mirror image of either of these series, but it trends more closely with GNP and therefore tends to trend in the opposite direction of unemployment.

These results align with standard economic theory, since consumption is a major additive component of a country's GNP and is driven directly by consumer spending behavior. Similarly, people tend to spend more money based on their income, so less unemployment means more people have jobs and so are spending more money, which drives consumption and therefore GNP. The reverse is also true: more unemployment implies less income which results in less consumer spending and therefore a reduction in GNP.

b)

```
VARselect(log_data, type = 'none')
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      3      2      3
##
## $criteria
##                    1             2             3             4             5
## AIC(n) -2.455574e+01 -2.490292e+01 -2.497922e+01 -2.495073e+01 -2.489341e+01
## HQ(n)  -2.448268e+01 -2.475681e+01 -2.476004e+01 -2.465849e+01 -2.452811e+01
## SC(n)  -2.437590e+01 -2.454325e+01 -2.443971e+01 -2.423138e+01 -2.399422e+01
## FPE(n)  2.165637e-11  1.530566e-11  1.418558e-11  1.460404e-11  1.548057e-11
##                    6             7             8             9            10
## AIC(n) -2.487411e+01 -2.482618e+01 -2.483303e+01 -2.479439e+01 -2.475113e+01
## HQ(n)  -2.443576e+01 -2.431476e+01 -2.424855e+01 -2.413685e+01 -2.402054e+01
## SC(n)  -2.379509e+01 -2.356731e+01 -2.339432e+01 -2.317585e+01 -2.295276e+01
## FPE(n)  1.580495e-11  1.661468e-11  1.654610e-11  1.725827e-11  1.810075e-11
```

```
q1b <- VAR(log_data, p = 2, type = 'none')
summary(q1b)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: log_emp, log_gnp, log_con
## Deterministic variables: none
## Sample size: 159
## Log Likelihood: 1305.087
## Roots of the characteristic polynomial:
## 0.9675 0.8852 0.7217 0.7217 0.147 0.01896
## Call:
## VAR(y = log_data, p = 2, type = "none")
##
##
## Estimation results for equation log_emp:
## ========================================
## log_emp = log_emp.l1 + log_gnp.l1 + log_con.l1 + log_emp.l2 + log_gnp.l2 + log_con.l2
##
##            Estimate Std. Error t value Pr(>|t|)
## log_emp.l1  1.10447    0.08261  13.370  < 2e-16 ***
## log_gnp.l1 -2.89012    0.76651  -3.771 0.000232 ***
## log_con.l1 -1.83833    0.83988  -2.189 0.030127 *
## log_emp.l2 -0.25833    0.08394  -3.078 0.002473 **
## log_gnp.l2  2.22724    0.76092   2.927 0.003945 **
## log_con.l2  1.99323    0.83181   2.396 0.017771 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
##
## Residual standard error: 0.07164 on 153 degrees of freedom
## Multiple R-Squared: 0.9158,  Adjusted R-squared: 0.9125
## F-statistic: 277.3 on 6 and 153 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation log_gnp:
## ========================================
## log_gnp = log_emp.l1 + log_gnp.l1 + log_con.l1 + log_emp.l2 + log_gnp.l2 + log_con.l2
##
##             Estimate Std. Error t value Pr(>|t|)
## log_emp.l1 -0.005525   0.011203  -0.493   0.6226
## log_gnp.l1  1.144621   0.103950  11.011   <2e-16 ***
## log_con.l1  0.266403   0.113900   2.339   0.0206 *
## log_emp.l2  0.019929   0.011383   1.751   0.0820 .
## log_gnp.l2 -0.168204   0.103192  -1.630   0.1052
## log_con.l2 -0.206874   0.112806  -1.834   0.0686 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
##
## Residual standard error: 0.009716 on 153 degrees of freedom
## Multiple R-Squared: 0.9478,  Adjusted R-squared: 0.9458
## F-statistic: 463.3 on 6 and 153 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation log_con:
## ========================================
## log_con = log_emp.l1 + log_gnp.l1 + log_con.l1 + log_emp.l2 + log_gnp.l2 + log_con.l2
##
##             Estimate Std. Error t value Pr(>|t|)
## log_emp.l1  0.010502   0.008976   1.170   0.2438
## log_gnp.l1  0.176646   0.083287   2.121   0.0355 *
## log_con.l1  0.877578   0.091260   9.616   <2e-16 ***
## log_emp.l2  0.002150   0.009120   0.236   0.8140
## log_gnp.l2 -0.114766   0.082680  -1.388   0.1671
## log_con.l2  0.065436   0.090383   0.724   0.4702
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
##
## Residual standard error: 0.007784 on 153 degrees of freedom
## Multiple R-Squared: 0.9282,  Adjusted R-squared: 0.9254
## F-statistic: 329.7 on 6 and 153 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##            log_emp    log_gnp    log_con
## log_emp  0.0051326 -3.548e-04 -1.112e-04
## log_gnp -0.0003548  9.435e-05  3.615e-05
## log_con -0.0001112  3.615e-05  6.055e-05
##
## Correlation matrix of residuals:
##         log_emp log_gnp log_con
## log_emp  1.0000 -0.5099 -0.1994
## log_gnp -0.5099  1.0000  0.4783
## log_con -0.1994  0.4783  1.0000
```

Since there are 3 univariate time series and we’re fitting a VAR(2) model that has removed trend and intercept, there will be three models that each have 6 parameters.

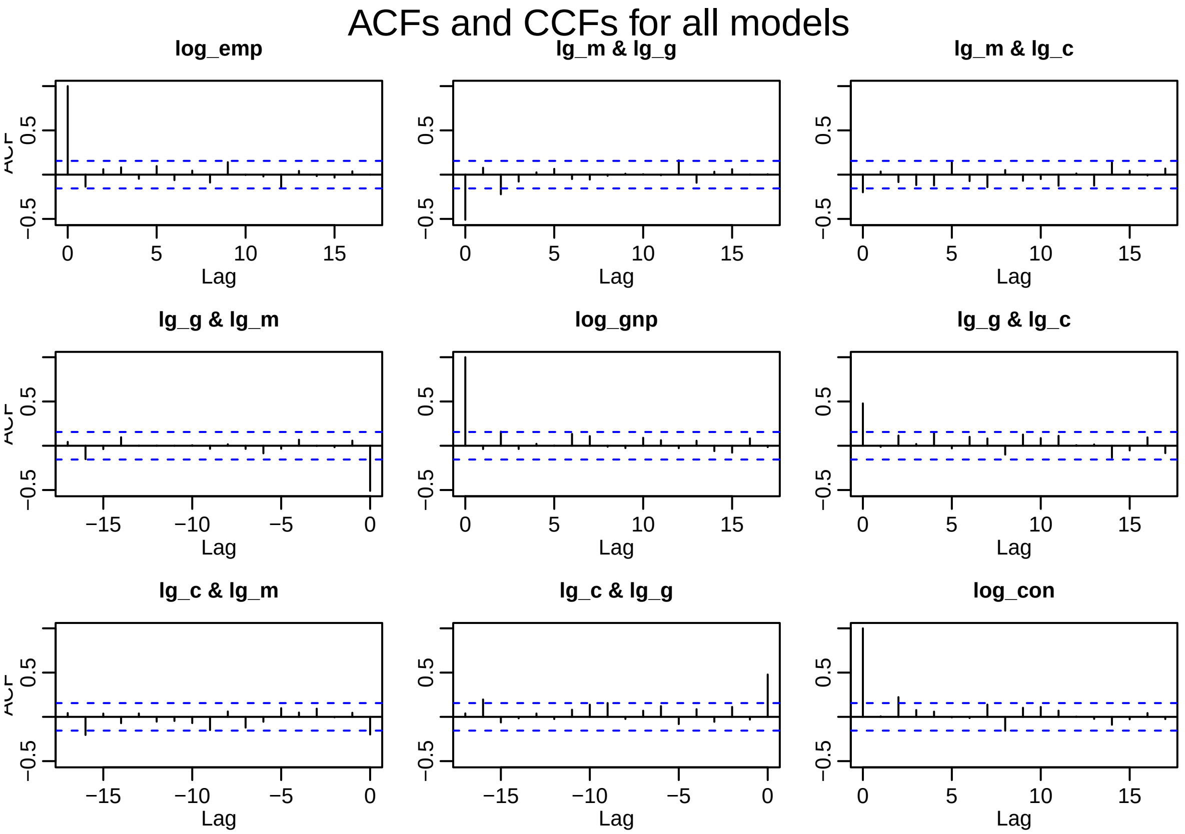
For the unemployment model, we see that for a single lag the coefficient of unemployment is positive while the coefficient of GNP and consumption is negative. Similarly, the signs of the coefficients for the second order lagged terms are also different for unemployment compared to GNP and consumption; however, in this case the signs are flipped in that the coefficient estimate is positive for unemployment and negative for the other two. This alternating coefficient signs suggests that there is negative autocorrelation between unemployment and GNP as well as negative autocorrelation between unemployment and consumption, which reflects the relationships observed in part a). All of the coefficients are also statistically significant at at least the 5% level. The adjusted r_t^2 also suggests the model is a good fit since it is greater than 90%.

For the GNP model, we see similar trends in the coefficient estimates from the unemployment model in that for a single lag the coefficient of unemployment is negative while the coefficient of GNP and consumption is positive. Similarly, the signs of the coefficients for the second order lagged terms are also different for unemployment compared to GNP and consumption; however, in this case the signs are flipped in that the coefficient estimate is negative for unemployment and positive for the other two. The alternating coefficient signs suggests that there is negative autocorrelation between unemployment and GNP as well as negative autocorrelation between unemployment and consumption, which reflects the relationships observed in part a). However, in this model only the first order lag coefficients for GNP and consumption are statistically significant at the 5% level. The adjusted r_t^2 also suggests the model is a good fit since it is greater than 90%.

For the consumption model, the coefficient estimates are all not statistically significant at the 5% level except for the first order lag for GNP and consumption. Also, the only negative coefficient estimate for model is the second order lag of GNP, although this is not statistically significant at the 5% level. This is different from the other two models but consistent with the findings in part a); that is, consumption seems to be less strongly correlated with the other two variables, which are essentially reflections of one another. The adjusted r_t^2 also suggests the model is a good fit since it is greater than 90%.

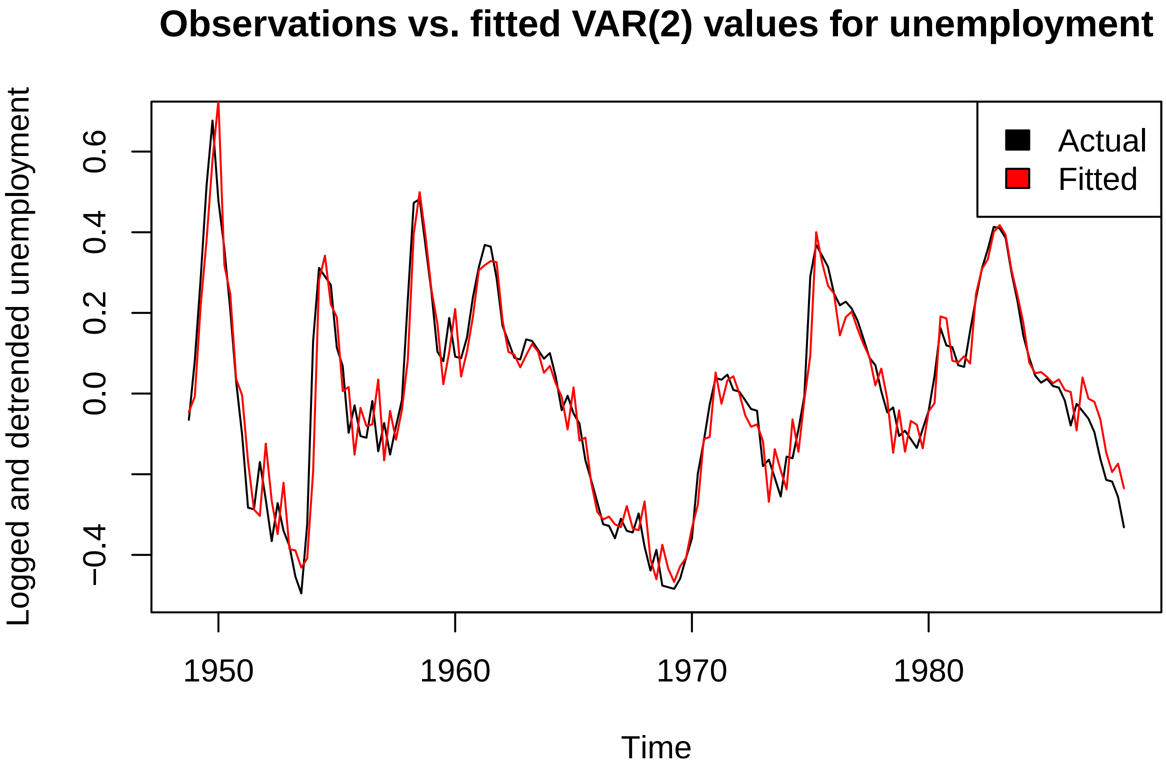
c)

```
acf(resid(q1b))
mtext('ACFs and CCFs for all models',
      side = 3, cex = 1.2, line = -1,
      outer = T)
```



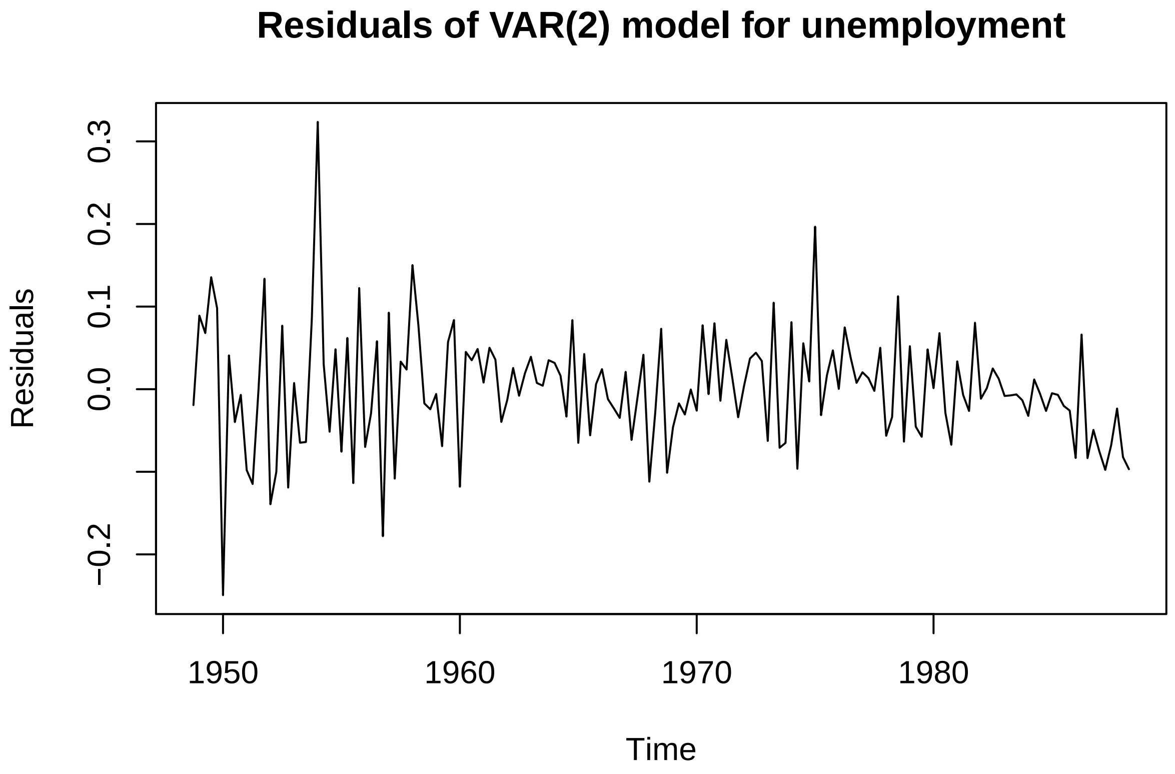
Unemployment model

```
plot(ts(log_emp[3:length(log_emp)], start = c(1948,4), frequency = 4),
     main = "Observations vs. fitted VAR(2) values for unemployment",
     ylab = "Logged and detrended unemployment")
lines(ts(log_emp[3:length(log_emp)], start = c(1948,4), frequency = 4) -
          q1b$varresult$log_emp$residuals, col="red")
legend("topright",legend = c("Actual", "Fitted"), fill = c("black", "red"))
```

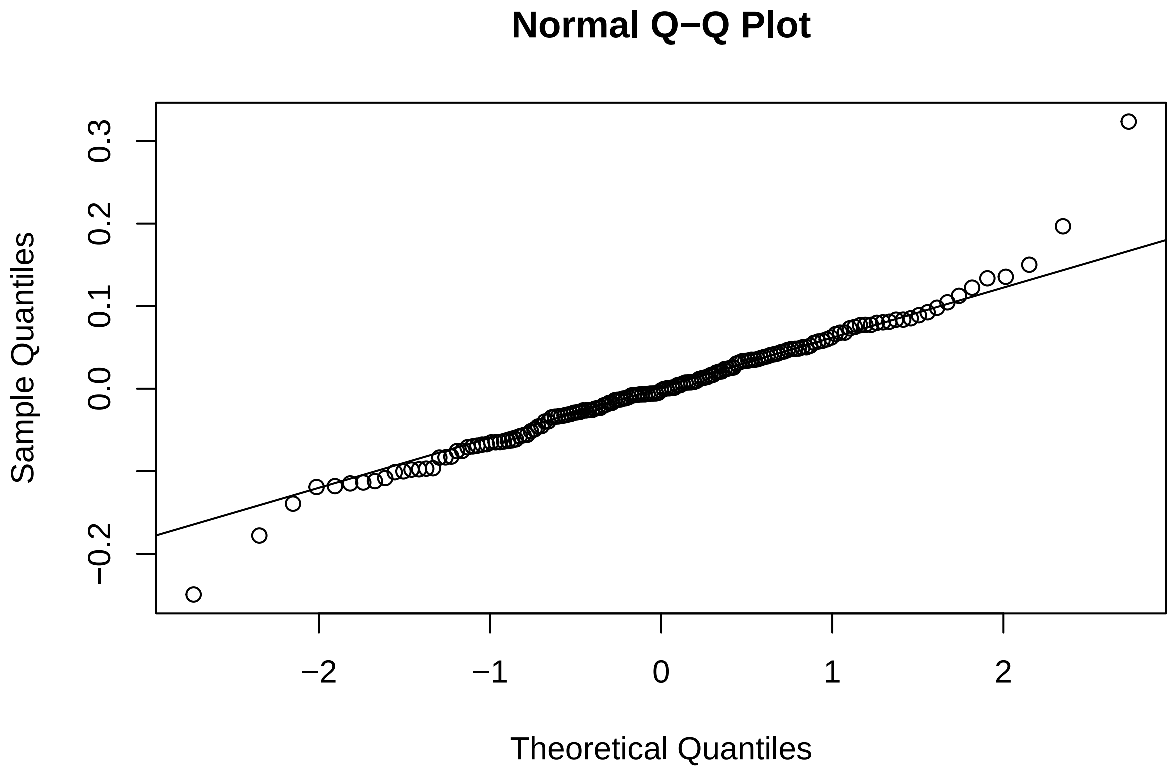


Comparing the observations to the fitted values, we see that the fitted values fit the data relatively well, although the these seem to be slightly shifted to the right for the majority of the data compared to the actual data. However, overall this fit suggests the model is a good fit for the data.

```
plot(ts(q1b$varresult$log_emp$residuals, frequency = 4, start = c(1948,4)),
     main = 'Residuals of VAR(2) model for unemployment',
     ylab = 'Residuals')
```



```
qqnorm(q1b$varresult$log_emp$residuals)
qqline(q1b$varresult$log_emp$residuals)
```

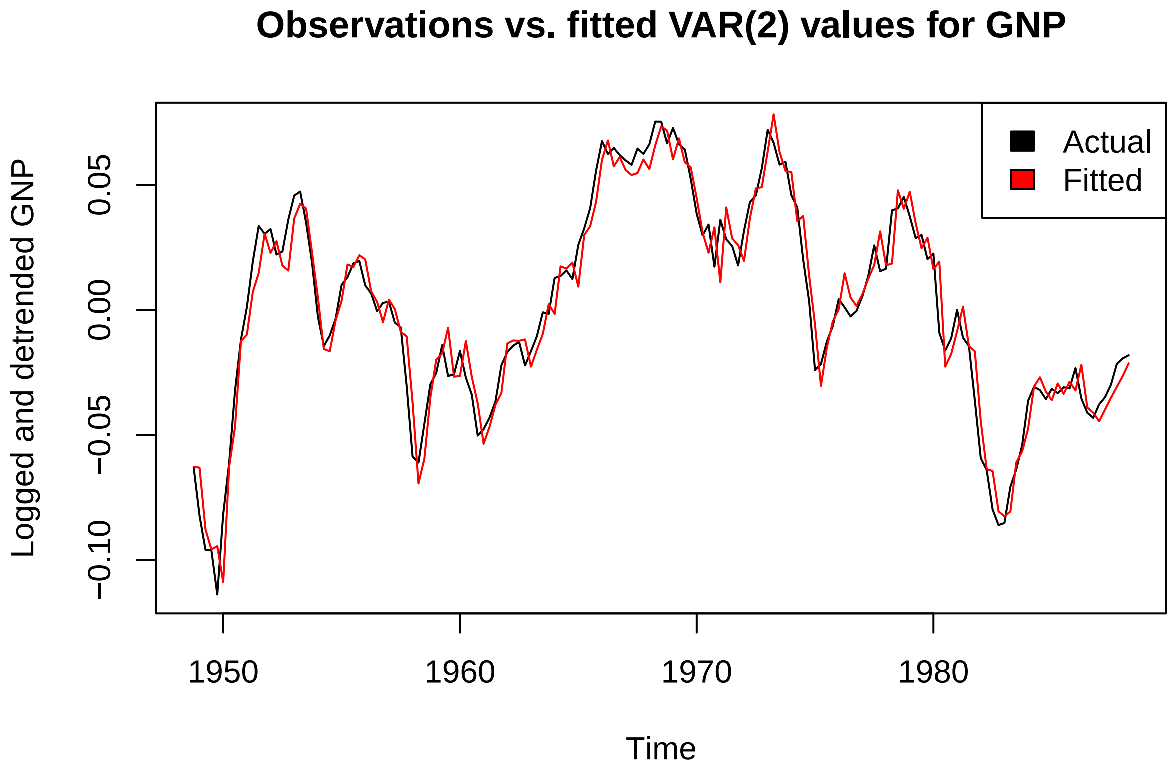


The residual values of model are centered at 0 and have relatively constant variance. The variance do not exhibit behaviors of autocorrelation since sequential terms do not usually have similar values based on the fluctuations. The ACF and CCF plots shown above illustrated that for unemployment all the lagged values beyond lag 0 of the residuals are approximately within the 95% error bars. The Q-Q plot illustrates that the residuals follow a normal distribution since the theoretical and sample quantiles are linearly related for the majority of the data; however, there are some minor deviations from normality at the tails. These suggest that the residuals are roughly Gaussian vector white noise.

Given that the fitted values fit the sample observations reasonably well and the residuals display Gaussian white noise behavior, this implies that the VAR(2) model is appropriate and a good fit for the unemployment data.

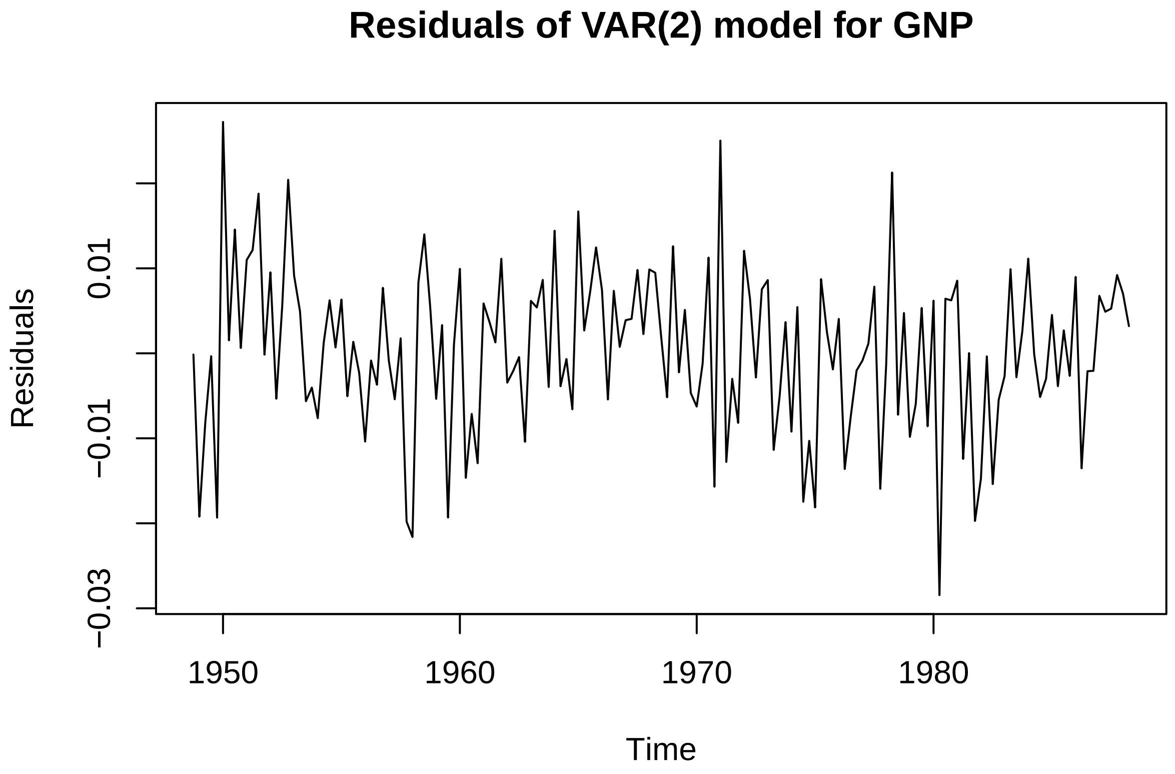
GNP model

```
plot(ts(log_gnp[3:length(log_gnp)], start = c(1948,4), frequency = 4),
     main = "Observations vs. fitted VAR(2) values for GNP",
     ylab = "Logged and detrended GNP")
lines(ts(log_gnp[3:length(log_gnp)], start = c(1948,4), frequency = 4) -
         q1b$varresult$log_gnp$residuals, col="red")
legend("topright",legend = c("Actual", "Fitted"), fill = c("black", "red"))
```

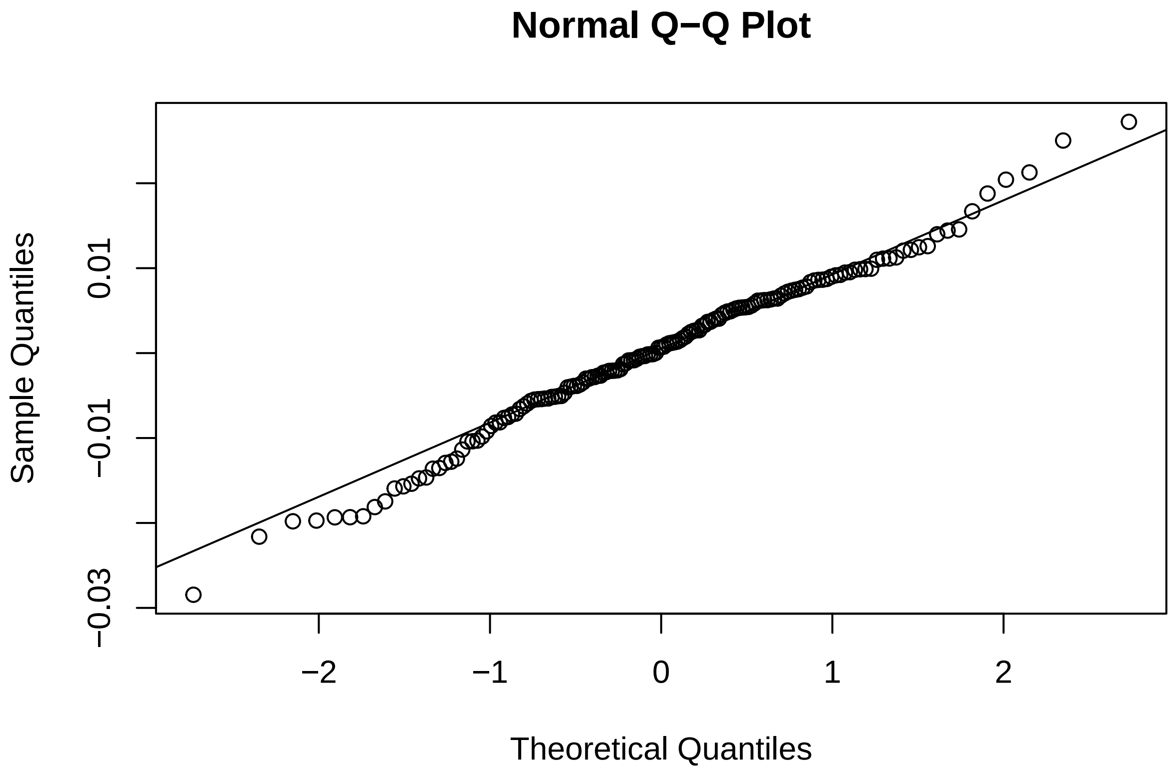


Comparing the observations to the fitted values, we see that the fitted values fit the data relatively well, although the these seem to be slightly shifted to the right for the majority of the data compared to the actual data. However, overall this fit suggests the model is a good fit for the data.

```
plot(ts(q1b$varresult$log_gnp$residuals, frequency = 4, start = c(1948,4)),
     main = 'Residuals of VAR(2) model for GNP', ylab = 'Residuals')
```



```
qqnorm(q1b$varresult$log_gnp$residuals)
qqline(q1b$varresult$log_gnp$residuals)
```

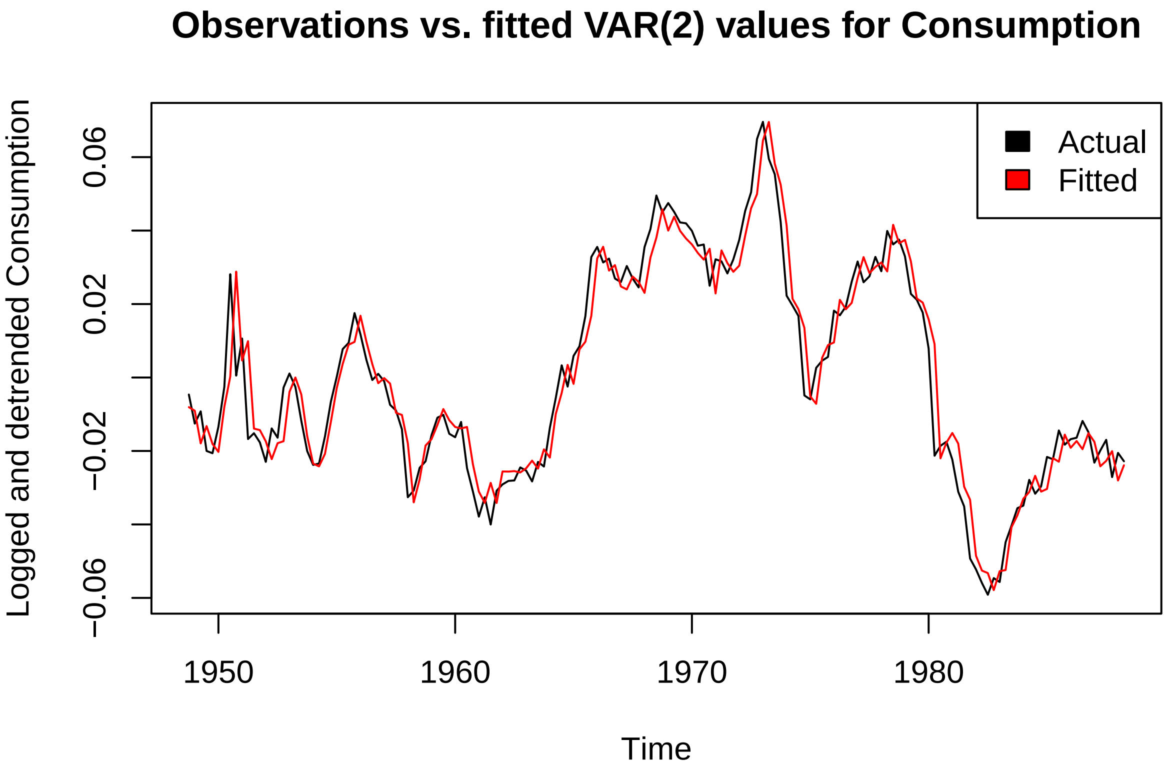


The residual values of model are centered at 0 and have relatively constant variance. The variance do not exhibit behaviors of autocorrelation since sequential terms do not usually have similar values based on the fluctuations. The ACF and CCF plots shown above illustrated that for GNP all the lagged values beyond lag 0 of the residuals are approximately within the 95% error bars. The Q-Q plot illustrates that the residuals follow a normal distribution since the theoretical and sample quantiles are linearly related for the majority of the data; however, there are some minor deviations from normality at the tails. These suggest that the residuals are roughly Gaussian white noise.

Given that the fitted values fit the sample observations reasonably well and the residuals display Gaussian vector white noise behavior, this implies that the VAR(2) model is appropriate and a good fit for the GNP data.

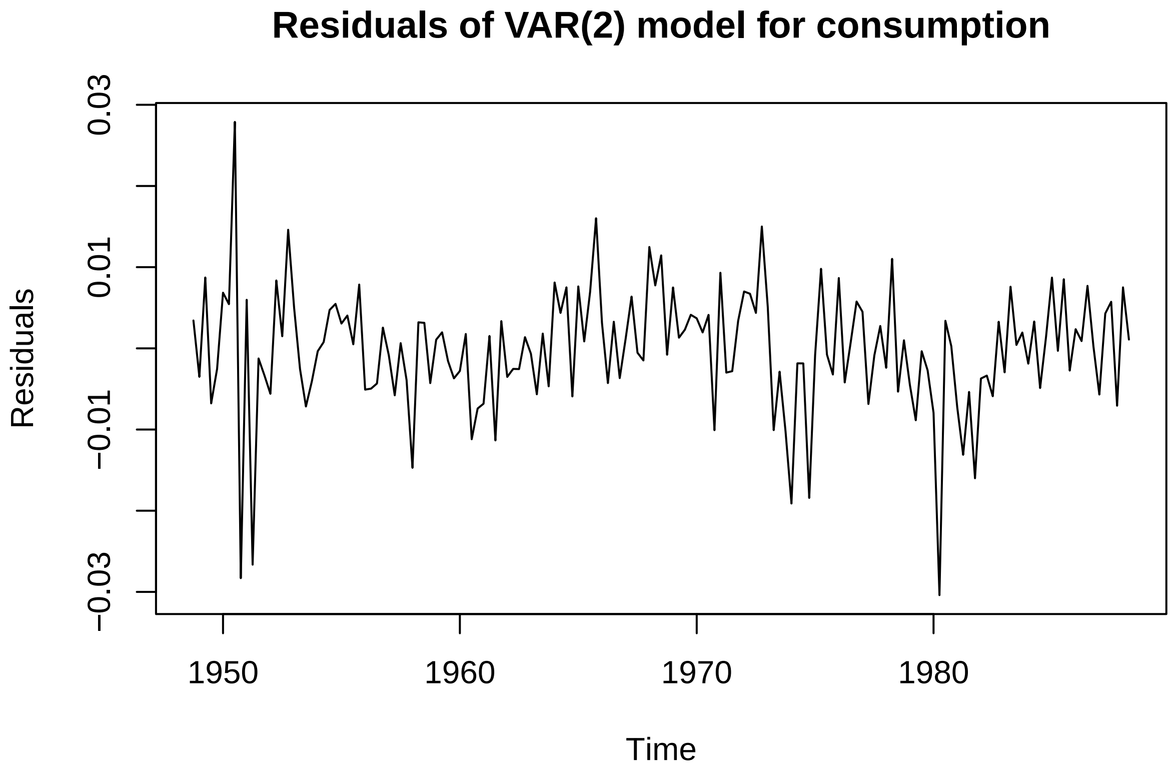
Consumption model

```
plot(ts(log_con[3:length(log_con)], start = c(1948,4), frequency = 4),
     main = "Observations vs. fitted VAR(2) values for Consumption",
     ylab = "Logged and detrended Consumption")
lines(ts(log_con[3:length(log_con)], start = c(1948,4), frequency = 4) -
         q1b$varresult$log_con$residuals, col="red")
legend("topright",legend = c("Actual", "Fitted"), fill = c("black", "red"))
```

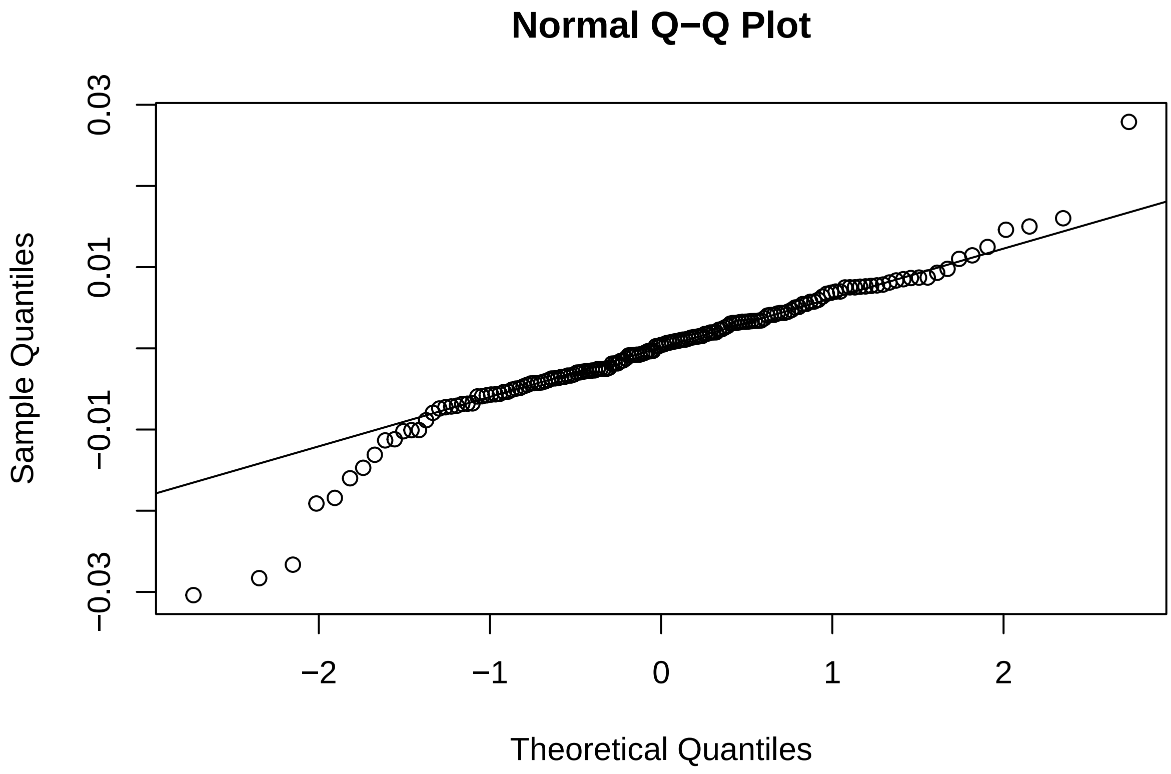


Comparing the observations to the fitted values, we see that the fitted values fit the data relatively well, although the these seem to be slightly shifted to the right for the majority of the data compared to the actual data. However, overall this fit suggests the model is a good fit for the data.

```
plot(ts(q1b$varresult$log_con$residuals, frequency = 4, start = c(1948,4)),
     main = 'Residuals of VAR(2) model for consumption', ylab = 'Residuals')
```



```
qqnorm(q1b$varresult$log_con$residuals)
qqline(q1b$varresult$log_con$residuals)
```



The residual values of model are centered at 0 and have relatively constant variance. The variance do not exhibit behaviors of autocorrelation since sequential terms do not usually have similar values based on the fluctuations. The ACF and CCF plots shown above illustrated that for consumption all the lagged values beyond lag 0 of the residuals are approximately within the 95% error bars. The Q-Q plot illustrates that the residuals follow a normal distribution since the theoretical and sample quantiles are linearly related for the majority of the data; however, there are some minor deviations from normality at the tails. These suggest that the residuals are roughly Gaussian vector white noise.

Given that the fitted values fit the sample observations reasonably well and the residuals display Gaussian vector white noise behavior, this implies that the VAR(2) model is appropriate and a good fit for the consumption data.

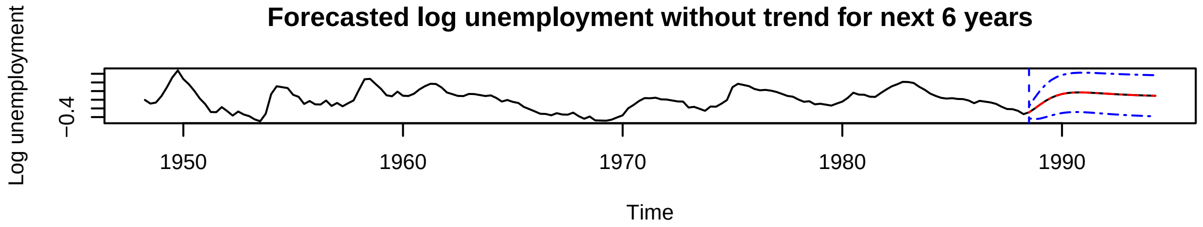
Overall this suggests that the VAR(2) model is an appropriate fit for a vector of each of the 3 univariate time series for unemployment, GNP, and consumption.

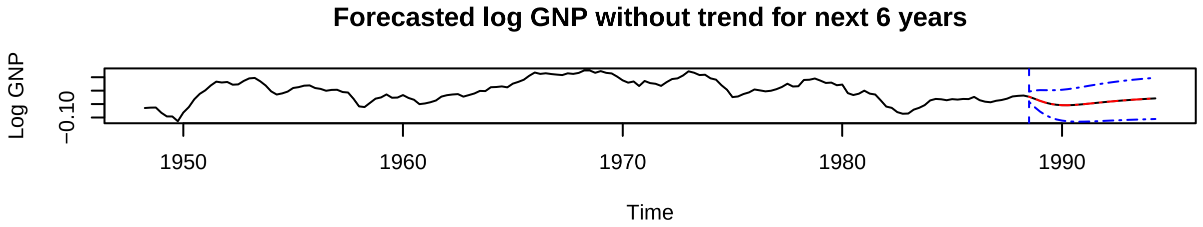
d)

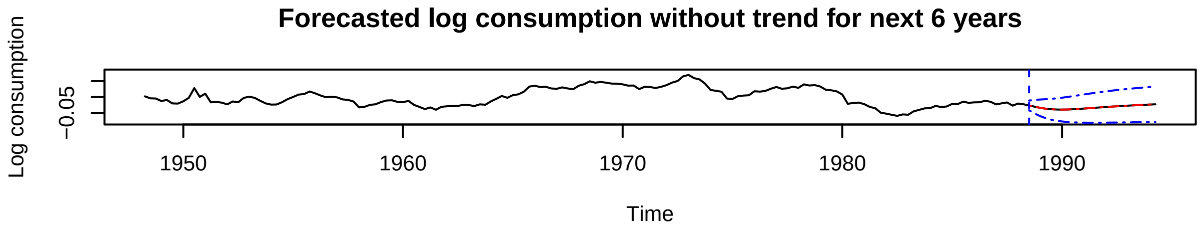
forecast without trend

```
q1di <- predict(q1b, n.ahead = 4*6)
log_emp_fcst <- ts(q1di$fcst$log_emp[,1], start = c(1988,3), frequency = 4)
log_emp_total <- ts(c(log_emp, log_emp_fcst), start = c(1948,2), frequency = 4)
log_emp_fcst_lower <- ts(q1di$fcst$log_emp[,2], start = c(1988,3), frequency = 4)
log_emp_fcst_upper <- ts(q1di$fcst$log_emp[,3], start = c(1988,3), frequency = 4)

log_gnp_fcst <- ts(q1di$fcst$log_gnp[,1], start = c(1988,3), frequency = 4)
```

```r
log_gnp_total <- ts(c(log_gnp, log_gnp_fcst), start = c(1948,2), frequency = 4)
log_gnp_fcst_lower <- ts(q1di$fcst$log_gnp[,2], start = c(1988,3), frequency = 4)
log_gnp_fcst_upper <- ts(q1di$fcst$log_gnp[,3], start = c(1988,3), frequency = 4)

log_con_fcst <- ts(q1di$fcst$log_con[,1], start = c(1988,3), frequency = 4)
log_con_total <- ts(c(log_con, log_con_fcst), start = c(1948,2), frequency = 4)
log_con_fcst_lower <- ts(q1di$fcst$log_con[,2], start = c(1988,3), frequency = 4)
log_con_fcst_upper <- ts(q1di$fcst$log_con[,3], start = c(1988,3), frequency = 4)

par(mfrow = c(3,1))
plot(log_emp_total,
     main = 'Forecasted log unemployment without trend for next 6 years',
     ylab = 'Log unemployment')
lines(log_emp_fcst, col = 'red', lty = 6)
lines(log_emp_fcst_lower, col = 'blue', lty = 6)
lines(log_emp_fcst_upper, col = 'blue', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')

plot(log_gnp_total,
     main = 'Forecasted log GNP without trend for next 6 years',
     ylab = 'Log GNP')
lines(log_gnp_fcst, col = 'red', lty = 6)
lines(log_gnp_fcst_lower, col = 'blue', lty = 6)
lines(log_gnp_fcst_upper, col = 'blue', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')

plot(log_con_total,
     main = 'Forecasted log consumption without trend for next 6 years',
     ylab = 'Log consumption',
     ylim = c(-.08,.08))
lines(log_con_fcst, col = 'red', lty = 6)
lines(log_con_fcst_lower, col = 'blue', lty = 6)
lines(log_con_fcst_upper, col = 'blue', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')
```







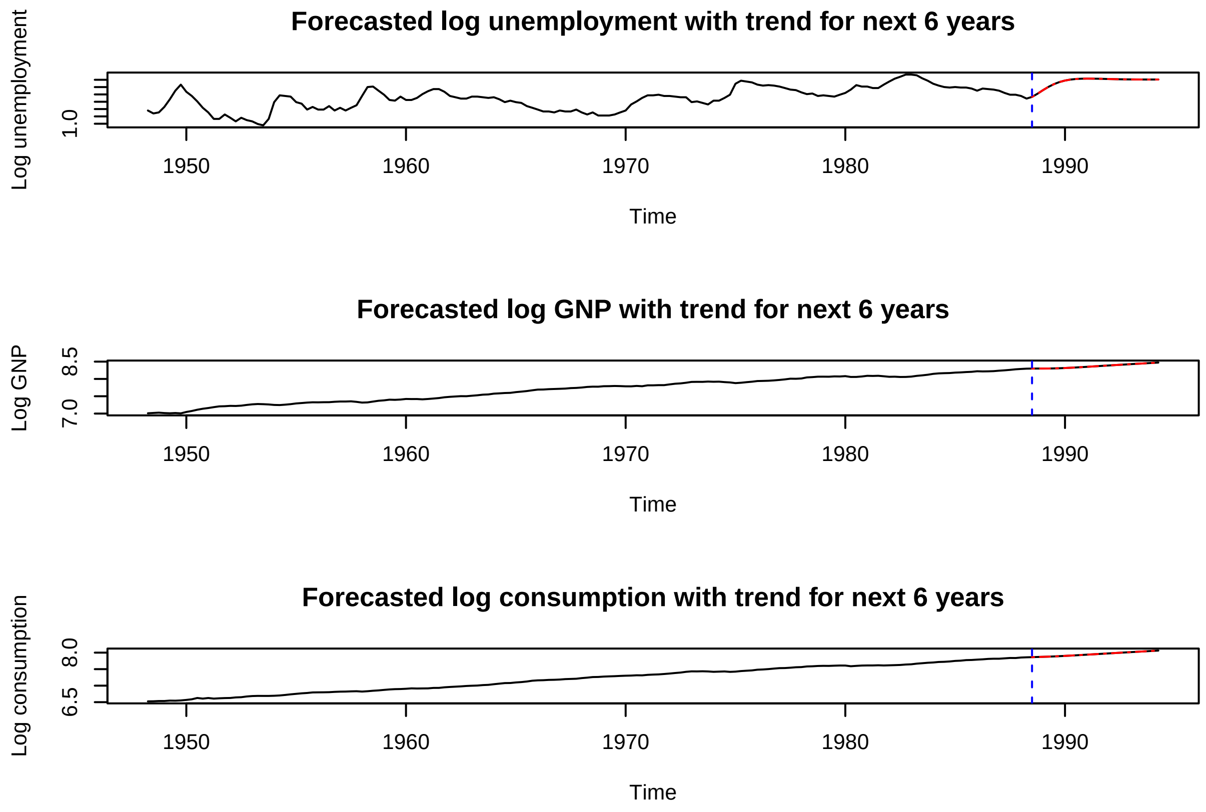
```
par(mfrow = c(1,1))
#plot(q1di)
# plot(cbind(unemp = log_emp_fcst, gnp = log_gnp_fcst, cons = log_con_fcst),
#      main = 'Forecasted unemployment, GNP, and consumption for next 6 years')
# plot(q1di,
#      xlim = c(length(resid(q1di$model))/3-dim(q1di$fcst$log_emp)[1],
#               length(resid(q1di$model))/3+dim(q1di$fcst$log_emp)[1]))
# plot(q1di)
```

Forecasting without a trend, we see that the model predicts that GNP and consumption will reach their minimums between 1990 and 1991 while unemployment will reach its maximum around 1991. After these extreme are reached, however, GNP and consumption are projected to increase through the start of 1994 and unemployment is projected to decrease (although at a slower rate than GNP and consumption increase) through the start of 1994.

forecast with trend

```
forecast_time <- seq(time(data)[length(time(data[,1]))]+0.25,
                     time(data)[length(time(data[,1]))]+6,
                     by = 0.25)

log_emp_trend_fcst <- coef(log_emp_lm)[1] + coef(log_emp_lm)[2]*forecast_time + log_emp_fcst
log_gnp_trend_fcst <- coef(log_gnp_lm)[1] + coef(log_gnp_lm)[2]*forecast_time + log_gnp_fcst
log_con_trend_fcst <- coef(log_con_lm)[1] + coef(log_con_lm)[2]*forecast_time + log_con_fcst
```

```r
log_emp_comb <- ts(c(as.vector(log(data[,1])),log_emp_trend_fcst), start = c(1948,2), frequency = 4)
log_gnp_comb <- ts(c(as.vector(log(data[,2])),log_gnp_trend_fcst), start = c(1948,2), frequency = 4)
log_con_comb <- ts(c(as.vector(log(data[,3])),log_con_trend_fcst), start = c(1948,2), frequency = 4)


par(mfrow = c(3,1))
plot(log_emp_comb, #xlim = c(1948, 1995),
     main = 'Forecasted log unemployment with trend for next 6 years',
     ylab = 'Log unemployment')
lines(forecast_time, log_emp_trend_fcst, col = 'red', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')

plot(log_gnp_comb, #xlim = c(1948, 1995), ylim = c(7, 9),
     main = 'Forecasted log GNP with trend for next 6 years',
     ylab = 'Log GNP')
lines(forecast_time, log_gnp_trend_fcst, col = 'red', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')

plot(log_con_comb, #xlim = c(1948, 1995), ylim = c(6, 9),
     main = 'Forecasted log consumption with trend for next 6 years',
     ylab = 'Log consumption')
lines(forecast_time, log_con_trend_fcst, col = 'red', lty = 6)
abline(v = max(time(data)) + 1/max(cycle(data)), lty = 2, col = 'blue')
```



```
# plot(cbind(unemp = log_emp_trend_fcst, gnp =  log_gnp_trend_fcst, cons =  log_con_trend_fcst),
#      main = 'Forecasted unemployment, GNP, and consumption (with trend) for next 6 years')
par(mfrow = c(1,1))
```

Forecasting with a trend, we see that an increasing trend occurs for unemployment in that it peaks around 1991 and then slowly decreases and eventually remains constant by 1994. This is relatively similar to the projection without a trend. However, the forecast with trend exhibits slightly different behavior for GNP and consumption relative to the forecast without trend. With the trend, wee see that both GNP and consumption are predicted to increase only from 1988 through the start of 1994.

Question 2

a)

We are given that

$$\begin{aligned} r_t &= \sigma_t \epsilon_t \\ \sigma_t^2 &= \alpha_0 + \sum_{j=1}^{p} \alpha_j r_{t-j}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2 \end{aligned}$$

and that $v_t = \sigma_t^2(\epsilon_t^2 - 1)$.

Using these equations we see that $r_t^2 = (\sigma_t \epsilon_t)^2$ and therefore

$$r_t^2 - \sigma_t^2 = (\sigma_t \epsilon_t)^2 - \sigma_t^2 = \sigma_t^2(\epsilon_t^2 - 1) = v_t$$

and so $\sigma_t^2 = r_t^2 - v_t$. We can use this to plug into the second equation which yields

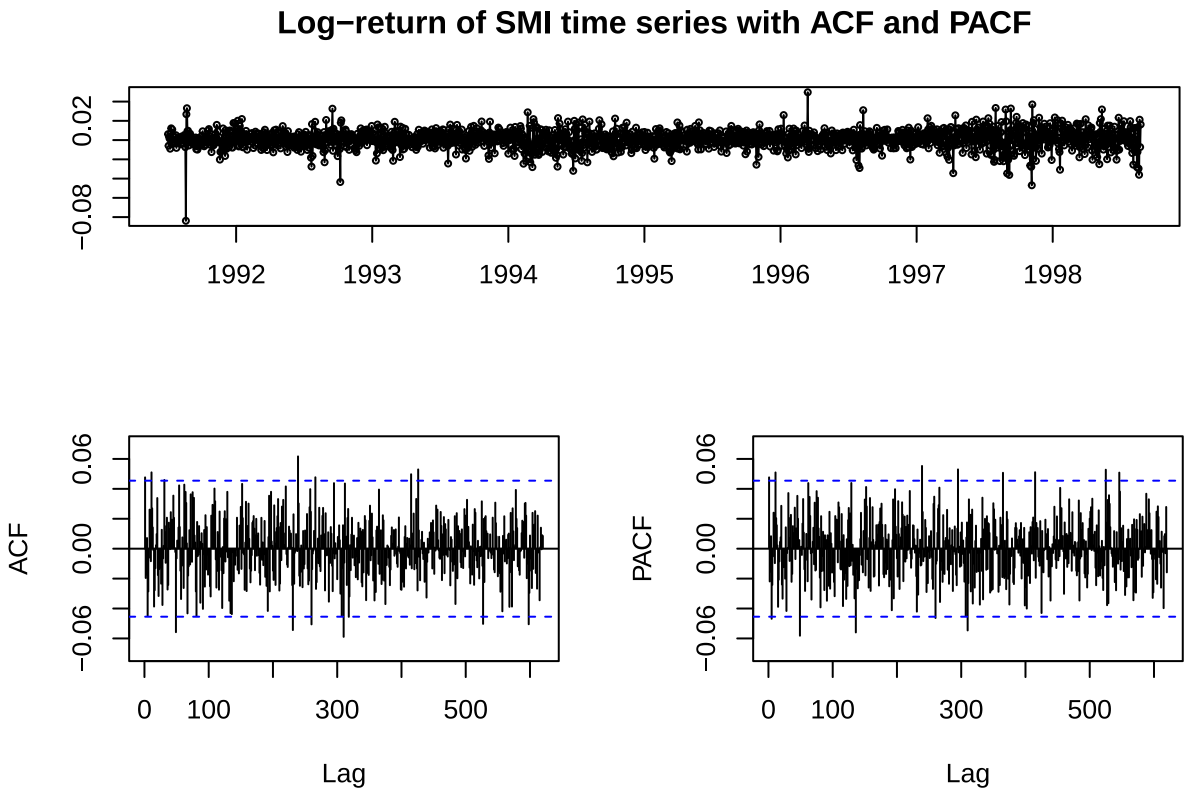
$$\begin{aligned} \sigma_t^2 &= \alpha_0 + \sum_{j=1}^{p} \alpha_j r_{t-j}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2 \\ r_t^2 - v_t &= \alpha_0 + \sum_{j=1}^{p} \alpha_j r_{t-j}^2 + \sum_{j=1}^{q} \beta_j (r_{t-j}^2 - v_{t-j}) \\ r_t^2 &= \alpha_0 + \sum_{j=1}^{p} \alpha_j r_{t-j}^2 + \sum_{j=1}^{q} \beta_j r_{t-j}^2 + v_t - \sum_{j=1}^{q} v_{t-j} \end{aligned}$$

and since we are given that $\alpha_j = 0$ for $j > p$ and $\beta_j = 0$ for $j > q$, this can then be written as

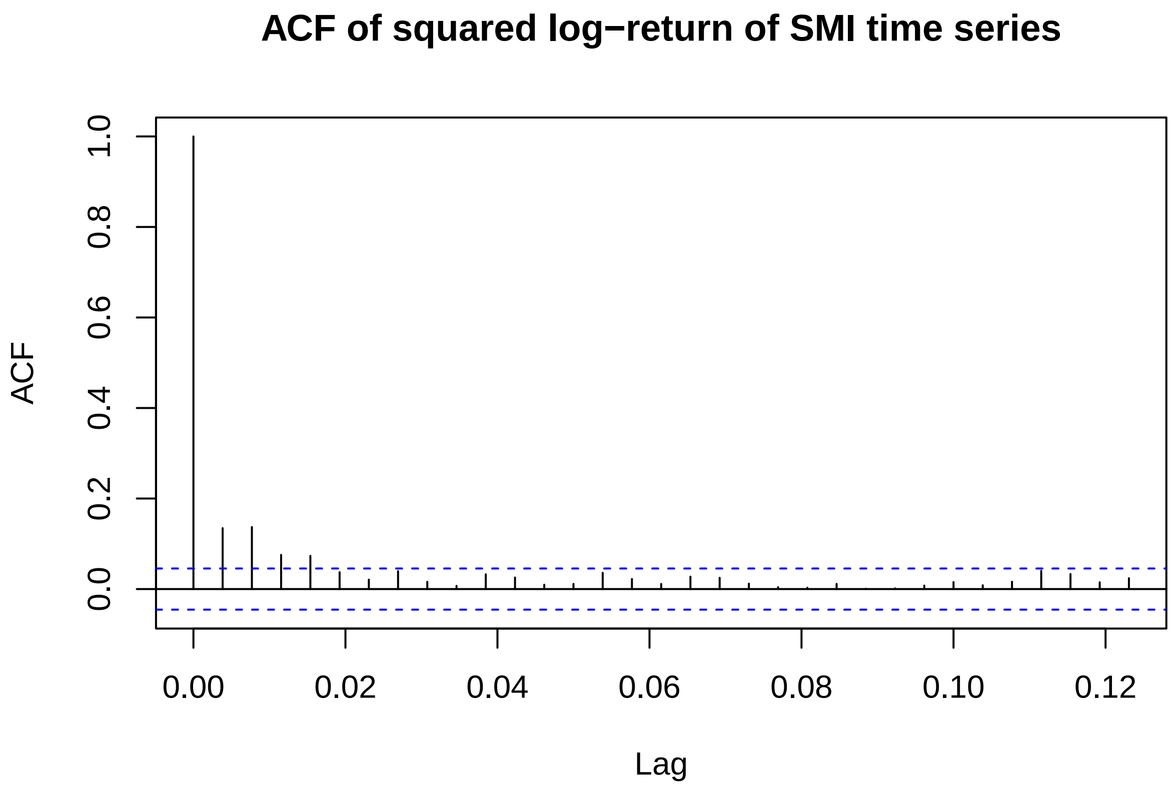
$$r_t^2 = \alpha_0 + \sum_{j=1}^{max(p,q)} (\alpha_j + \beta_j) r_{t-j}^2 + v_t - \sum_{j=1}^{q} v_{t-j}$$

b)

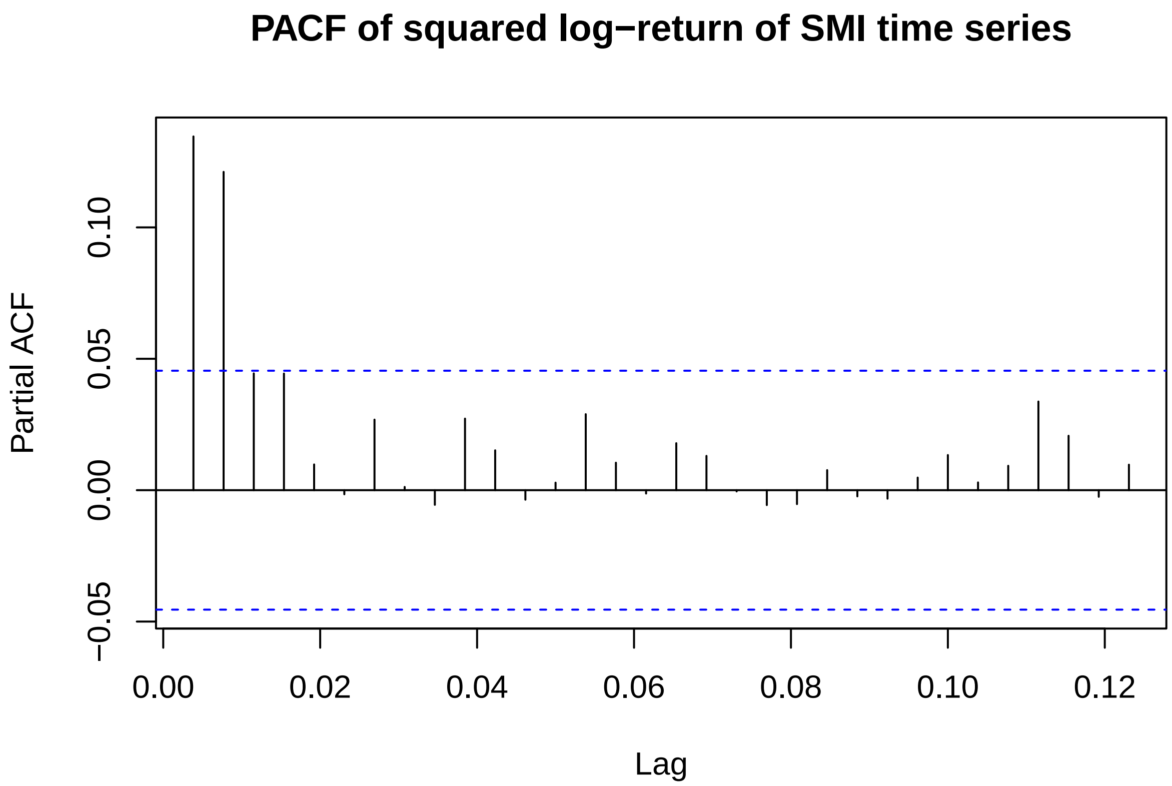
```
rt <- diff(log(EuStockMarkets[,'SMI']))
rt2 <- rt^2
tsdisplay(rt, main = 'Log-return of SMI time series with ACF and PACF')
```



```
acf(rt2, main = 'ACF of squared log-return of SMI time series')
```



```
pacf(rt2, main = 'PACF of squared log-return of SMI time series')
```



From the plots for r_t we see that it is approximately white noise since the data are roughly centered at zero and are not temporally correlated over time since sequential observations do not have similar signs. However, there is volatility in this data with spikes randomly occurring throughout the time series such as in 1991 or 1996. Additionally, the ACF and PACF plots show that the majority of the ACF and PACF values at different lags are within the 95% error bars.

Examining the ACF and PACF plot for r_t^2, we see that in both plots that the respective values for the first two lags are noticeably outside of the 95% percent error bars. These results and a) suggest that a GARCH(p,q) model could be reasonable for r_t since r_t^2 displays ARMA(p,q) structure.

c)

```
q2c_fit <- auto.arima(rt2, max.p = 5, max.q = 5, max.order = 10, stationary = T, seasonal = F, trace = 
```

```
##
##  ARIMA(0,0,0)                    with zero mean     : -25556.8
##  ARIMA(0,0,0)                    with non-zero mean : -25789.62
##  ARIMA(0,0,1)                    with zero mean     : -25631.94
##  ARIMA(0,0,1)                    with non-zero mean : -25814.87
##  ARIMA(0,0,2)                    with zero mean     : -25690.64
##  ARIMA(0,0,2)                    with non-zero mean : -25839.07
##  ARIMA(0,0,3)                    with zero mean     : -25711.25
##  ARIMA(0,0,3)                    with non-zero mean : -25842.85
##  ARIMA(0,0,4)                    with zero mean     : -25731.88
##  ARIMA(0,0,4)                    with non-zero mean : -25847.99
##  ARIMA(0,0,5)             with zero mean     : -25739.86
##  ARIMA(0,0,5)             with non-zero mean : -25847.32
##  ARIMA(1,0,0)             with zero mean     : -25662.4
##  ARIMA(1,0,0)             with non-zero mean : -25821.6
##  ARIMA(1,0,1)             with zero mean     : Inf
##  ARIMA(1,0,1)             with non-zero mean : -25852.73
##  ARIMA(1,0,2)             with zero mean     : Inf
##  ARIMA(1,0,2)             with non-zero mean : -25851.99
##  ARIMA(1,0,3)             with zero mean     : Inf
##  ARIMA(1,0,3)             with non-zero mean : -25850.84
##  ARIMA(1,0,4) with zero mean     : Inf
##  ARIMA(1,0,4)             with non-zero mean : Inf
##  ARIMA(1,0,5) with zero mean     : Inf
##  ARIMA(1,0,5)             with non-zero mean : Inf
##  ARIMA(2,0,0)             with zero mean     : -25731.66
##  ARIMA(2,0,0)             with non-zero mean : -25847.05
##  ARIMA(2,0,1)             with zero mean     : Inf
##  ARIMA(2,0,1)             with non-zero mean : -25852.11
##  ARIMA(2,0,2) with zero mean     : Inf
##  ARIMA(2,0,2)             with non-zero mean : -25850.86
##  ARIMA(2,0,3) with zero mean     : Inf
##  ARIMA(2,0,3)             with non-zero mean : -25848.95
##  ARIMA(2,0,4)             with zero mean     : Inf
##  ARIMA(2,0,4)             with non-zero mean : -25846.97
##  ARIMA(2,0,5) with zero mean     : Inf
##  ARIMA(2,0,5)             with non-zero mean : Inf
##  ARIMA(3,0,0)             with zero mean     : -25749.36
##  ARIMA(3,0,0)             with non-zero mean : -25848.72
##  ARIMA(3,0,1) with zero mean     : Inf
##  ARIMA(3,0,1)             with non-zero mean : -25851.04
##  ARIMA(3,0,2) with zero mean     : Inf
##  ARIMA(3,0,2)             with non-zero mean : -25848.93
##  ARIMA(3,0,3) with zero mean     : Inf
##  ARIMA(3,0,3)             with non-zero mean : Inf
##  ARIMA(3,0,4) with zero mean     : Inf
##  ARIMA(3,0,4)             with non-zero mean : Inf
##  ARIMA(3,0,5) with zero mean     : Inf
##  ARIMA(3,0,5)             with non-zero mean : Inf
##  ARIMA(4,0,0)             with zero mean     : -25764.45
##  ARIMA(4,0,0)             with non-zero mean : -25850.5
##  ARIMA(4,0,1) with zero mean     : Inf
##  ARIMA(4,0,1)             with non-zero mean : -25848.61
##  ARIMA(4,0,2) with zero mean     : Inf
##  ARIMA(4,0,2)             with non-zero mean : -25846.84
##  ARIMA(4,0,3) with zero mean     : Inf
##  ARIMA(4,0,3)             with non-zero mean : -25846.68
##  ARIMA(4,0,4) with zero mean     : Inf
##  ARIMA(4,0,4)             with non-zero mean : Inf
##  ARIMA(4,0,5) with zero mean     : Inf
##  ARIMA(4,0,5)             with non-zero mean : Inf
##  ARIMA(5,0,0)             with zero mean     : -25768.44
##  ARIMA(5,0,0)             with non-zero mean : -25848.7
##  ARIMA(5,0,1)             with zero mean     : Inf
##  ARIMA(5,0,1)             with non-zero mean : Inf
##  ARIMA(5,0,2) with zero mean     : Inf
##  ARIMA(5,0,2)             with non-zero mean : Inf
##  ARIMA(5,0,3) with zero mean     : Inf
##  ARIMA(5,0,3)             with non-zero mean : Inf
##  ARIMA(5,0,4) with zero mean     : Inf
##  ARIMA(5,0,4)             with non-zero mean : Inf
##  ARIMA(5,0,5) with zero mean     : Inf
##  ARIMA(5,0,5)             with non-zero mean : Inf
##
##
##
##  Best model: ARIMA(1,0,1)             with non-zero mean
```

The auto.arima function identifies the ARMA(1,1) model is appropriate for r_t^2, so using part a) the model specified should be GARCH(1,1).

d)

```
q2d_fit <- garchFit(~garch(1,1), data = rt)
```

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 1)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 1
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          1859
##  Recursion Init:            mci
##  Series Scale:              0.009250036
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                        U           V     params includes
##     mu     -0.88421240   0.8842124 0.08842124     TRUE
##     omega   0.00000100 100.0000000 0.10000000     TRUE
##     alpha1  0.00000001   1.0000000 0.10000000     TRUE
##     gamma1 -0.99999999   1.0000000 0.10000000    FALSE
##     beta1   0.00000001   1.0000000 0.80000000     TRUE
##     delta   0.00000000   2.0000000 2.00000000    FALSE
##     skew    0.10000000  10.0000000 1.00000000    FALSE
##     shape   1.00000000  10.0000000 4.00000000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1  beta1
##      1      2      3      5
##  Persistence:                  0.9
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     2563.8960: 0.0884212 0.100000 0.100000 0.800000
##   1:     2563.7817: 0.0884274 0.102030 0.100554 0.800450
##   2:     2563.7017: 0.0884382 0.102508 0.0995685 0.798601
##   3:     2563.5553: 0.0884656 0.106451 0.0998386 0.796924
##   4:     2563.2440: 0.0885422 0.109596 0.101291 0.789090
##   5:     2563.0349: 0.0886379 0.112705 0.107085 0.783639
##   6:     2562.6395: 0.0891084 0.119923 0.110694 0.769389
##   7:     2562.6036: 0.0893279 0.124918 0.109218 0.769634
##   8:     2562.4892: 0.0897770 0.124451 0.109566 0.766947
##   9:     2562.4375: 0.0901486 0.124514 0.113255 0.765505
##  10:     2562.3317: 0.0911001 0.126582 0.112595 0.761909
##  11:     2562.2798: 0.0920401 0.130668 0.112267 0.760008
##  12:     2562.1858: 0.0930360 0.130041 0.113325 0.757760
##  13:     2562.0829: 0.0950326 0.130249 0.117814 0.755851
##  14:     2561.9676: 0.0970429 0.132159 0.117082 0.752291
##  15:     2561.8908: 0.0990435 0.136032 0.117451 0.749709
##  16:     2561.5746: 0.108886 0.147378 0.130727 0.725729
##  17:     2561.5657: 0.111012 0.145982 0.129157 0.728515
##  18:     2561.5645: 0.111127 0.147867 0.128480 0.727136
##  19:     2561.5627: 0.111361 0.147834 0.128902 0.726501
##  20:     2561.5615: 0.111598 0.147994 0.129394 0.726144
##  21:     2561.5606: 0.112190 0.148533 0.130223 0.724909
##  22:     2561.5606: 0.112196 0.148586 0.130235 0.724852
##  23:     2561.5606: 0.112196 0.148583 0.130236 0.724853
##
## Final Estimate of the Negative LLH:
##  LLH:  -6144.374    norm LLH:  -3.305204
##           mu        omega       alpha1        beta1
## 1.037813e-03 1.271327e-05 1.302356e-01 7.248533e-01
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1         beta1
## mu      -25072715.44 -1.228153e+08  1.241490e+04     -14988.18
## omega  -122815252.51 -2.390290e+12 -1.143234e+08 -172899661.59
## alpha1      12414.90 -1.143234e+08 -9.614993e+03     -10017.80
## beta1      -14988.18 -1.728997e+08 -1.001780e+04     -13760.53
## attr(,"time")
## Time difference of 0.02325988 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
```

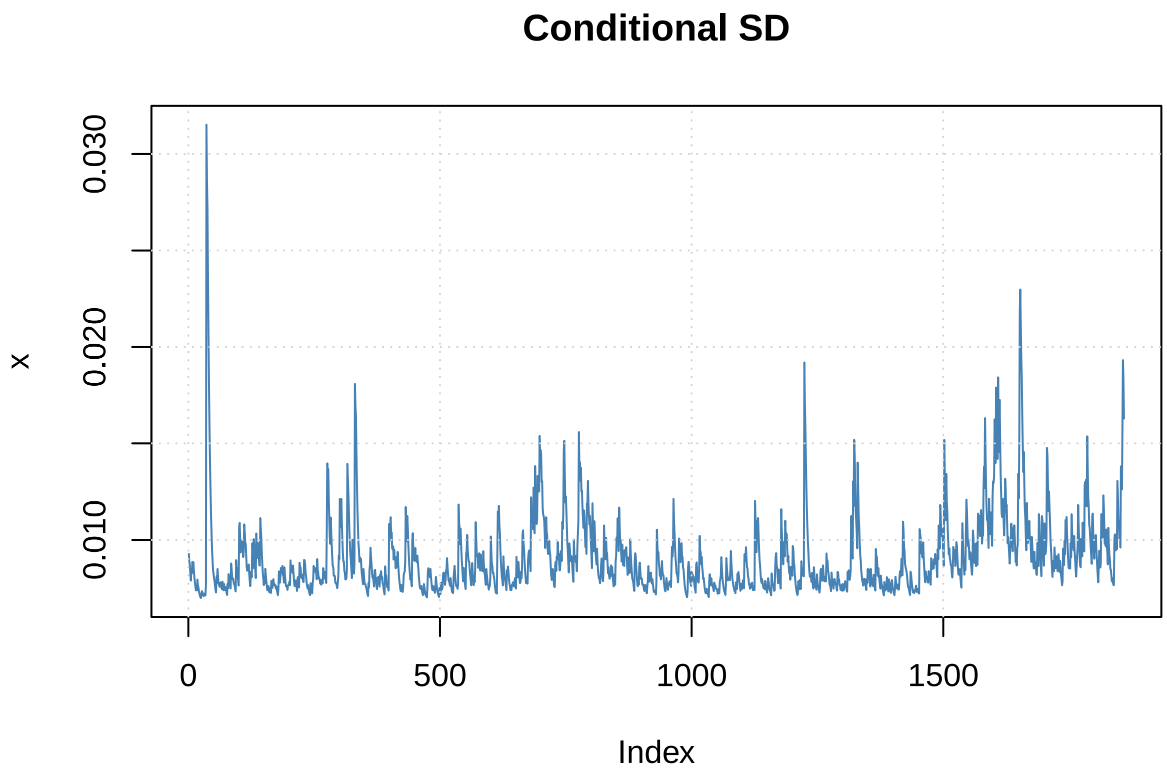
```
##  Time difference of 0.09941101 secs
```

```
summary(q2d_fit)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = rt)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x000000001c7be078>
##  [data = rt]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 1.0378e-03  1.2713e-05  1.3024e-01  7.2485e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     1.038e-03   2.016e-04    5.149 2.62e-07 ***
## omega  1.271e-05   2.507e-06    5.070 3.97e-07 ***
## alpha1 1.302e-01   2.439e-02    5.340 9.32e-08 ***
## beta1  7.249e-01   4.431e-02   16.358  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Log Likelihood:
##  6144.374    normalized:  3.305204
##
## Description:
##  Tue Apr 12 21:35:15 2022 by user: Owner
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  6927.395  0
##  Shapiro-Wilk Test  R    W      0.9518397 0
##  Ljung-Box Test     R    Q(10)  15.50268  0.1147815
##  Ljung-Box Test     R    Q(15)  20.89969  0.1400651
##  Ljung-Box Test     R    Q(20)  22.69771  0.3038804
##  Ljung-Box Test     R^2  Q(10)  0.7598238 0.9999519
##  Ljung-Box Test     R^2  Q(15)  1.300684  0.9999984
##  Ljung-Box Test     R^2  Q(20)  1.451523  1
##  LM Arch Test       R    TR^2   1.054866  0.9999809
##
```

```
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -6.606104 -6.594210 -6.606114 -6.601721
```

```
# round(q2d_fit@fit$matcoef, 6)
plot(q2d_fit, which = 2)
```



From the summary we see that the parameter estimates are all statistically significantly different from 0 at the 5% level or lower. The mu term represents the mean of the model, omega is the intercept term of the volatility of r_t (defined as σ_t^2 in part a), alpha1 represents the first lag of the squared log-return and beta1 is the first lag of the volatility of r_t.

The plot of the conditional standard deviation clearly shows that the variance is not constant over time and that there are spikes that randomly occur in the standard deviation and align with the spikes in the log-return of the original time series, which further illustrates that the GARCH model is appropriate since it allows for conditional heteroscedasticity over the data points.

Question 3

How

In the technology industry, designed experiments are used to determine how changing a minor feature can influence an outcome measure of interest. A/B testing is a popular example of this, and its conducted by changing a small feature in a technology application (i.e. website) and then randomly assigning users to either the A group (control) or B group (treatment) and measuring differences in usage. This can also be extended to more univariable tests (i.e. A/B/C/D testing where A is control and the rest are treatments) or multivariable testing. Measures are referred to as overall evaluation criterion (OEC) and need to be developed by the experiment designer in order to measure the success of the experiment. This often relies on domain expertise, so the designers of the experiment typically need to work with the relevant stakeholders and experts when preparing an experiment.

In addition to establishing these measures, preliminary data checks should also be considered to make sure the experiment gets rigorous results. A/A testing is an example of this, where the users are randomly assigned both to the control, with the expectation being that the OEC measures should agree about 95% of the time. Other considerations include outliers or errors and technical issues when collecting data from the experiment as well as other relevant covariates. Collecting this additional data can allow the experiments to disaggregate the data and analyze whether the results hold across these specific subdemographics. Other experimental considerations are to avoid reusing the same users in treatment/control groups by reshuffling them between treatment and control in order to prevent carryover effects of users being assigned to the same group repeated times. Checks are also run to validate that the percentage of users in the experiment match up with the initial design of the experiment.

The article specifically mentions three different types of models for experimentation that are employed by businesses. The first is the centralized model, where a team serves the entire company. This has the benefit of focusing on long-term projects for better tools and algorithms. However, the drawbacks are that different units of the business have different priorities, and the team has less domain knowledge and may not be able to convince management of their results. The second is the decentralized model where experimenters are distributed through the company, thus allowing them to gain domain expertise. However, the experiments may have a difficult time advancing because they cannot develop skills, or their experiments are not relevant. The final model mentioned is the center of excellence model, which combines the first two models mentioned. This lowers the time and resources needed for experimentation and spreads best practices throughout the organization. The drawback is that there is not ownership clarity and responsibility and therefore it is difficult to appropriately allocate resources. The context of the business usually dictates which model is most appropriate.

Simple designs are preferred in order to find the relationship between a treatment and control group, but it is important to note that correlation doesn't necessary imply causation in the context of A/B testing. This is because there is typically insufficient controls available to account for other factors that may cause differences between treatment and control groups.

Why

Designed experiments are often employed in the technology industry because they are typically low cost experiments to randomly assign users for a controlled experiment. The results allow the companies to quickly assess the quality of new ideas such as models, strategies, products, services, and marketing campaigns. This allows more scientific decision-making to optimize business performance rather than being based on domain knowledge intuition alone. Similarly, these experiments can be used to help guide investments in future features by quantifying the impact of these decisions and measuring the tradeoffs in rolling out potential new features. Technology firms can often quickly collect large amounts of data about these small changes that provide value to the business. The experiments are usually very scalable for larger companies assuming they have sufficient infrastructure to collect and analyze the data and can be run concurrrently to save time and money.