

Time Series - Homework 4

Matt Isaac

October 29, 2018

Exercise 1. Brockwell and Davis 3.6

Show that the two MA(1) processes

$$X_t = Z_t + \theta Z_{t-1}, \quad \{Z_t\} \sim WN(0, \sigma^2)$$

$$Y_t = \tilde{Z}_t + \frac{1}{\theta} \tilde{Z}_{t-1}, \quad \{\tilde{Z}_t\} \sim WN(0, \theta^2 \sigma^2)$$

where $0 < |\theta| < 1$, have the same autocovariance functions.

First, for X_t , we have:

$$\gamma_X(0) = \text{cov}(X_t, X_t)$$

$$\gamma_X(0) = \text{Var}(X_t)$$

$$\gamma_X(0) = \text{Var}(Z_t + \theta Z_{t-1})$$

Then, since Z_t and Z_{t-1} are uncorrelated, we have:

$$\gamma_X(0) = \text{Var}(Z_t) + \text{Var}(\theta Z_{t-1})$$

$$\gamma_X(0) = \text{Var}(Z_t) + \theta^2 \text{Var}(Z_{t-1})$$

$$\gamma_X(0) = \sigma^2 + \theta^2 \sigma^2$$

$$\gamma_X(0) = (1 + \theta^2) \sigma^2$$

Next, $\gamma_X(1)$:

$$\gamma_X(1) = \text{cov}(X_t, X_{t-1})$$

$$\gamma_X(1) = E[X_t X_{t-1}]$$

$$\gamma_X(1) = E[(Z_t + \theta Z_{t-1})(Z_{t-1} + \theta Z_{t-2})]$$

Since there is only one “overlap” between the two terms in the expected value, there is only one covariance that is non-zero.

$$\gamma_X(1) = E[\theta Z_{t-1}^2]$$

$$\gamma_X(1) = \theta \sigma^2$$

Next, $\gamma(2)$:

$$\gamma_X(2) = \text{cov}(X_t, X_{t-2})$$

$$\gamma_X(2) = E[(Z_t + \theta Z_{t-1})(Z_{t-2} + \theta Z_{t-3})]$$

We see that there are no “overlaps” between the terms in the expected values. So,

$$\gamma_X(h) = 0, \quad h \geq 2$$

Next, we will compute the ACVF for Y_t :

First, $\gamma_Y(0)$

$$\gamma_Y(0) = \text{cov}(Y_t, Y_t)$$

$$\gamma_Y(0) = \text{Var}(Y_t)$$

$$\gamma_Y(0) = \text{Var}(\tilde{Z}_t + \frac{1}{\theta}\tilde{Z}_{t-1})$$

Then, since \tilde{Z}_t and \tilde{Z}_{t-1} are uncorrelated, we have:

$$\gamma_Y(0) = \text{Var}(\tilde{Z}_t) + \text{Var}(\frac{1}{\theta}\tilde{Z}_{t-1})$$

$$\gamma_Y(0) = \text{Var}(\tilde{Z}_t) + \frac{1}{\theta^2}\text{Var}(\tilde{Z}_{t-1})$$

$$\gamma_Y(0) = \theta^2\sigma^2 + \frac{1}{\theta^2}\theta^2\sigma^2$$

$$\gamma_Y(0) = (1 + \theta^2)\sigma^2$$

Next, $\gamma_Y(1)$:

$$\gamma_Y(1) = \text{cov}(Y_t, Y_{t-1})$$

$$\gamma_Y(1) = E[Y_t Y_{t-1}]$$

$$\gamma_Y(1) = E[(\tilde{Z}_t + \frac{1}{\theta}\tilde{Z}_{t-1})(\tilde{Z}_{t-1} + \frac{1}{\theta}\tilde{Z}_{t-2})]$$

Since there is only one “overlap” between the two terms in the expected value, there is only one covariance that is non-zero.

$$\gamma_Y(1) = E[\frac{1}{\theta}\tilde{Z}_{t-1}^2]$$

$$\gamma_Y(1) = \frac{1}{\theta}\theta^2\sigma^2$$

$$\gamma_Y(1) = \theta\sigma^2$$

Next, $\gamma(2)$:

$$\gamma_Y(2) = \text{cov}(Y_t, Y_{t-2})$$

$$\gamma_Y(2) = E[(\tilde{Z}_t + \frac{1}{\theta}\tilde{Z}_{t-1})(\tilde{Z}_{t-2} + \frac{1}{\theta}\tilde{Z}_{t-3})]$$

We see that there are no “overlaps” between the terms in the expected values. So,

$$\gamma_Y(h) = 0, \quad h \geq 2$$

In conclusion,

$$\gamma_X(h) = \gamma_Y(h) \begin{cases} (1 + \theta^2)\sigma^2 & h = 0 \\ \theta\sigma^2 & h = 1 \\ 0 & h \geq 2 \end{cases}$$

Exercise 2. Consider the monthly log returns of CRSP equal-weighted index from January 1962 to December 1999 *mlrew.xlsx* containing 456 observations. Build a MA(q) model for this data set. Use the first 36 years for model fitting and the last 2 years for forecasting.

a. Decide on the value of q and justify your choice.

```
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.4.4
```

```
mlr <- read.csv("mlr_ew.csv", header = FALSE) # read in data
```

```
mlr <- ts(mlr, start = 1, frequency = 12)
```

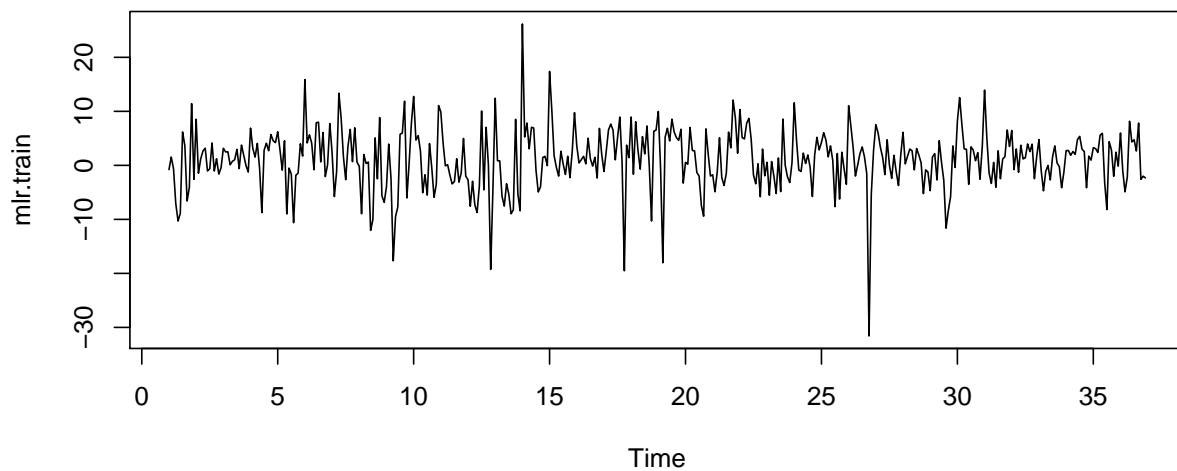
```
mlr.train <- mlr[1:(12*36)]
```

```
mlr.train <- ts(mlr.train, start = 1, frequency = 12)
```

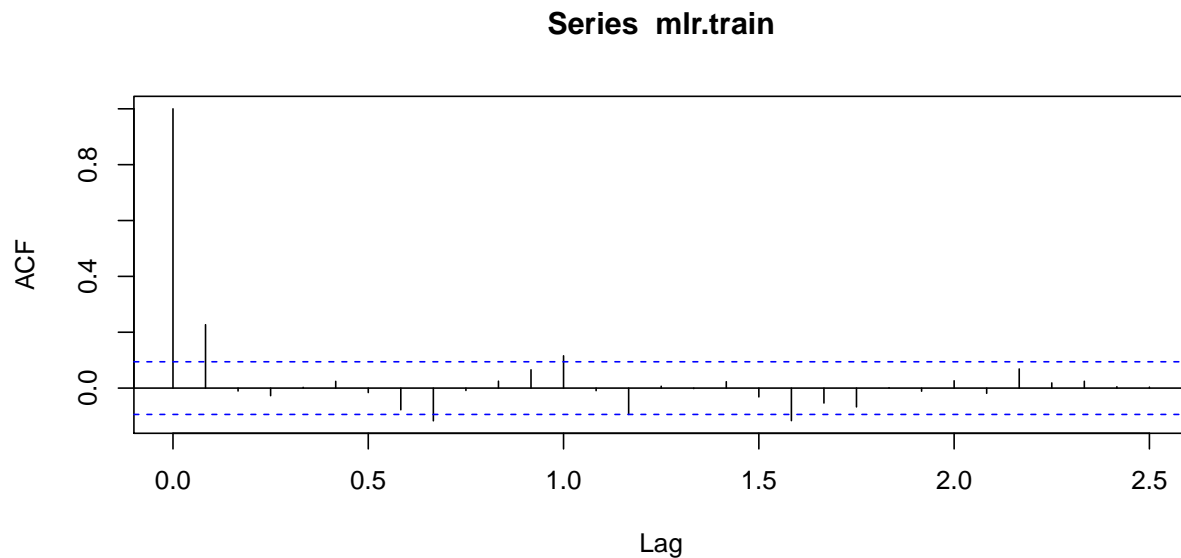
```
mlr.test <- mlr[((12*36)+1):length(mlr)]
```

```
mlr.test <- ts(mlr.test, start = 37, frequency = 12)
```

```
plot(mlr.train) # appears stationary
```



```
acf(mlr.train, lag.max = 30) # Appears stationary. Plot ACF to determine value of q
```



The ACF plot above shows that the last significant lag is at lag 19. Thus, we will fit a MA(19) model.

b. Find the ML estimates for the parameters. Perform appropriate tests on whether the estimated MA coefficients are significant.

First, from looking back at the ACF, we can see that the coefficients will likely be significant at lags 1, 8, 12, and 19.

The coefficients are estimated as follows (using the CSS-ML method):

```
ma.19 <- arima(mlr.train - mean(mlr.train),
               order = c(0,0,19),
               method = "CSS-ML",
               include.mean = FALSE)

round(ma.19$coef, 4)
```

##	ma1	ma2	ma3	ma4	ma5	ma6	ma7	ma8	ma9
##	0.2405	-0.0110	-0.0164	0.0160	0.0281	-0.0359	-0.0599	-0.1144	0.0209
##	ma10	ma11	ma12	ma13	ma14	ma15	ma16	ma17	ma18
##	0.0345	0.0299	0.1193	0.0028	-0.1133	0.0111	0.0020	0.0102	-0.0136
##	ma19								
##	-0.1272								

We can also use the `coeftest()` function to obtain further evidence of significant/insignificant coefficients.

```
library(lmtest)
coeftest(ma.19)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ma1    0.2404527  0.0483000  4.9783 6.414e-07 ***
## ma2   -0.0110104  0.0500170 -0.2201  0.82577
## ma3   -0.0163599  0.0503770 -0.3247  0.74537
## ma4    0.0159586  0.0498876  0.3199  0.74905
## ma5    0.0280573  0.0499940  0.5612  0.57465
## ma6   -0.0359468  0.0507297 -0.7086  0.47858
## ma7   -0.0598766  0.0504816 -1.1861  0.23558
## ma8   -0.1144222  0.0513293 -2.2292  0.02580 *
## ma9    0.0209171  0.0513407  0.4074  0.68370
## ma10   0.0345243  0.0537932  0.6418  0.52100
## ma11   0.0298789  0.0520311  0.5743  0.56580
## ma12   0.1192702  0.0539678  2.2100  0.02710 *
## ma13   0.0027939  0.0529487  0.0528  0.95792
## ma14  -0.1132517  0.0477874 -2.3699  0.01779 *
## ma15   0.0111104  0.0515864  0.2154  0.82948
## ma16   0.0019833  0.0571592  0.0347  0.97232
## ma17   0.0102424  0.0522974  0.1958  0.84473
## ma18  -0.0135803  0.0509088 -0.2668  0.78966
## ma19  -0.1271648  0.0540770 -2.3515  0.01870 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We reach similar conclusions as we did by looking at the ACF plot. It appears that the 1st, 8th, 12th, and 19th coefficients may be significant. The 14th coefficient may also be significant. I will assess the fit of the

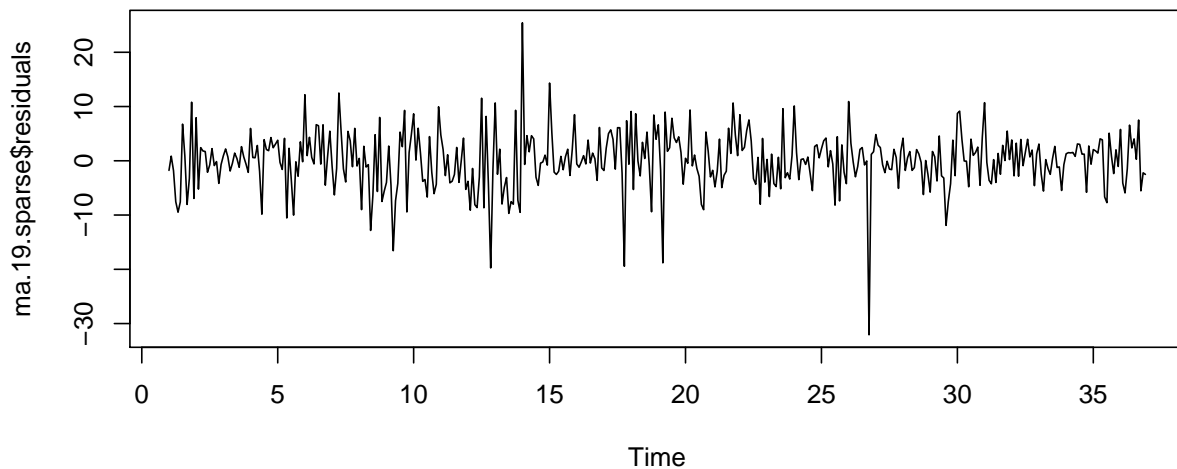
model without the 14th coefficient, and add it in if the model does not adequately fit the data.

c. Based on the result of the previous question, fit a sparse MA model to the data.

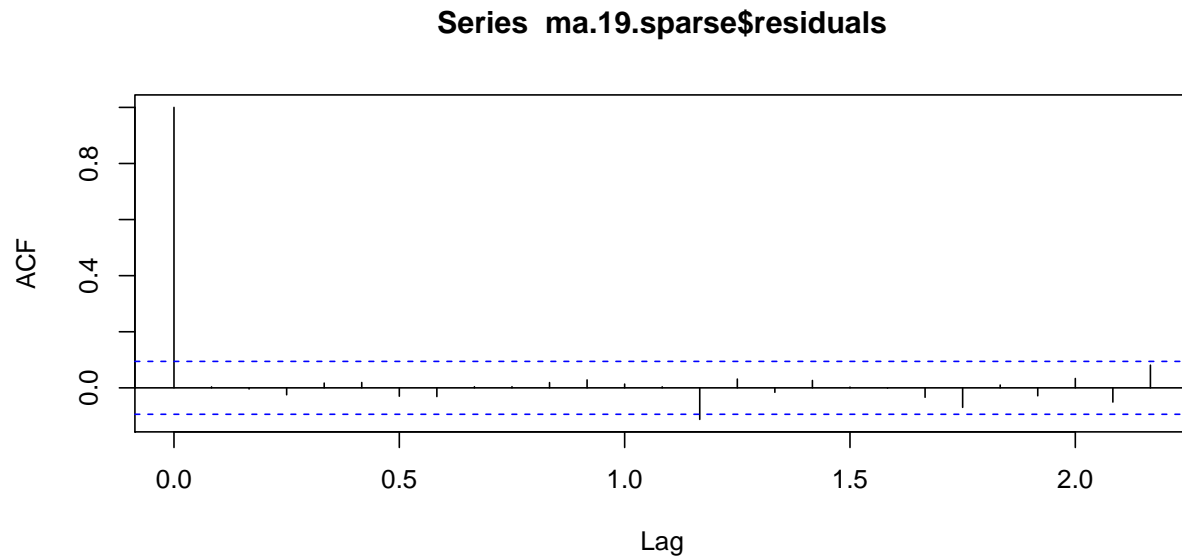
```
ma.19.sparse <- arima(mlr.train - mean(mlr.train),  
  order = c(0,0,19),  
  method = "CSS-ML",  
  fixed = c(NA, rep(0,6), NA,  
    rep(0,3), NA,  
    rep(0,6), NA),  
  include.mean = FALSE)
```

d. What do you conclude from the residuals?

```
plot(ma.19.sparse$residuals)
```



```
# acf(ma.19$residuals)  
acf(ma.19.sparse$residuals)
```



The residuals resemble white noise in both plots above. The ACF plot particularly shows good evidence that the residuals are white noise, as there is only one slightly significant ACF at lag 14. This is strong evidence that our sparse model adequately fits these data.

e. Make forecasts up to 2 years ahead using the method of realized innovations.

```
getInnovations <- function(timeSeries, coeff.index, ma.model){
  innov <- double(length(timeSeries))
  innov[1] <- timeSeries[1]
  for(t in 2:length(timeSeries)){
    Xt <- mlr[t]
    summation <- 0
    for(j in coeff.index){
      if(t - j <= 0){
        innov_tmj <- 0
      } else {
        innov_tmj <- innov[t - j]
      }
      summation <- summation + (ma.model$coef[j] * innov_tmj)
    }
    innov[t] <- Xt - summation
  }
  return(innov)
}

ci <- c(1,8,12,19)
innov <- getInnovations(timeSeries = mlr.train,
                        coeff.index = ci,
                        ma.model = ma.19.sparse)
innov <- ts(innov, start = 1, frequency = 12)

hStepAhead <- function(h, ma.model, innov, timeseries){
  n <- length(timeseries)
```

```

q <- length(ma.model$coef)
if(h > q){
  return(0)
} else {
  summation <- 0
  for(j in h:q){
    summation <- summation + (ma.model$coef[j] * innov[n + h - j])
  }
  return(summation)
}
}

innov.preds <- double(24)
for(i in 1:length(innov.preds)){
  innov.preds[i] <- hStepAhead(h = i, ma.model = ma.19.sparse, innov = innov,
                                timeseries = mlr.train)
}
preds <- ts(innov.preds + mean(mlr.train), start = 37, frequency = 12)

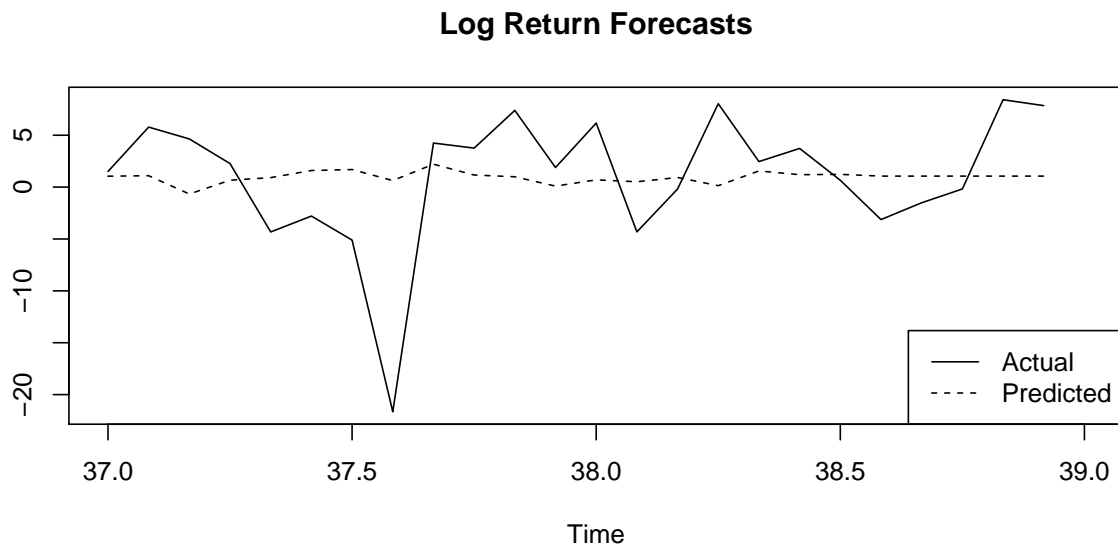
```

f. Display the forecasts v.s. the actual values in the same figure.

```

{ts.plot(mlr.test, preds, gpars = list(lty = c("solid", "dashed"),
                                          xlim = c(37,39),
                                          main = "Log Return Forecasts"))
legend("bottomright", lty = c("solid", "dashed"),
      legend = c("Actual", "Predicted"))}

```



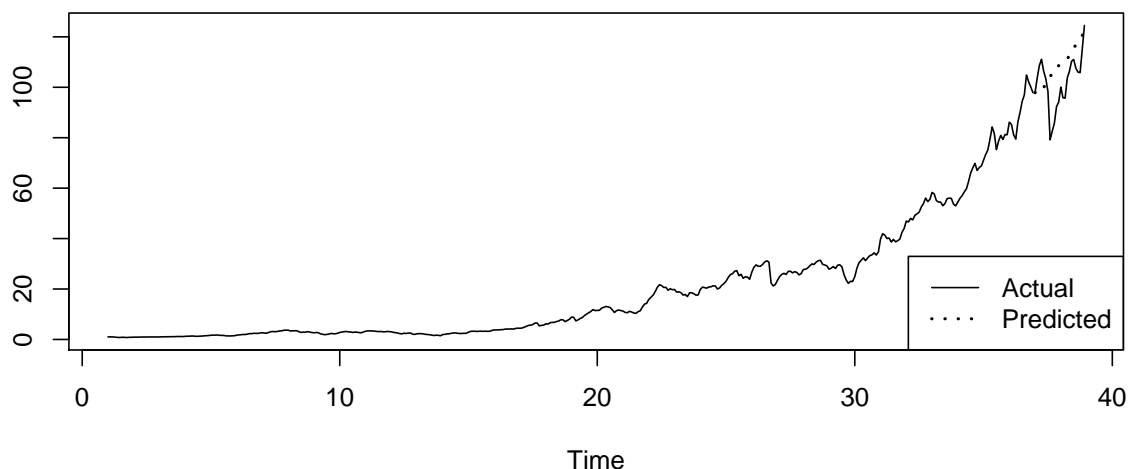
g. Transform the log return data back to the original price data. Also transform the 2 years forecasts of log returns to those of prices. Plot the entire price series and the last 2 years forecasts in the same figure.

```
toOrigPrice <- function(logret, start){
  op <- double(length(logret))
  op[1] <- start
  for(i in 2:length(op)){
    op[i] <- op[i - 1] * exp(logret[i]/100)
  }
  return(op)
}
op.train <- ts(toOrigPrice(mlr.train, start = 1), start = 1, frequency = 12)
anchor <- op.train[length(op.train)]

op.test <- ts(toOrigPrice(mlr.test, start = anchor), start = 37, frequency = 12)
op.forc <- ts(toOrigPrice(preds, start = anchor), start = 37, frequency = 12)

{ts.plot(op.train, op.test, op.forc,
  gpars = list(lty = c("solid", "solid", "dotted"),
    main = "Original Price Forecasts", lwd = c(1,1,2))
legend("bottomright", lty = c("solid", "dotted"),
  legend = c("Actual", "Predicted"), lwd = c(1, 2))}
```

Original Price Forecasts

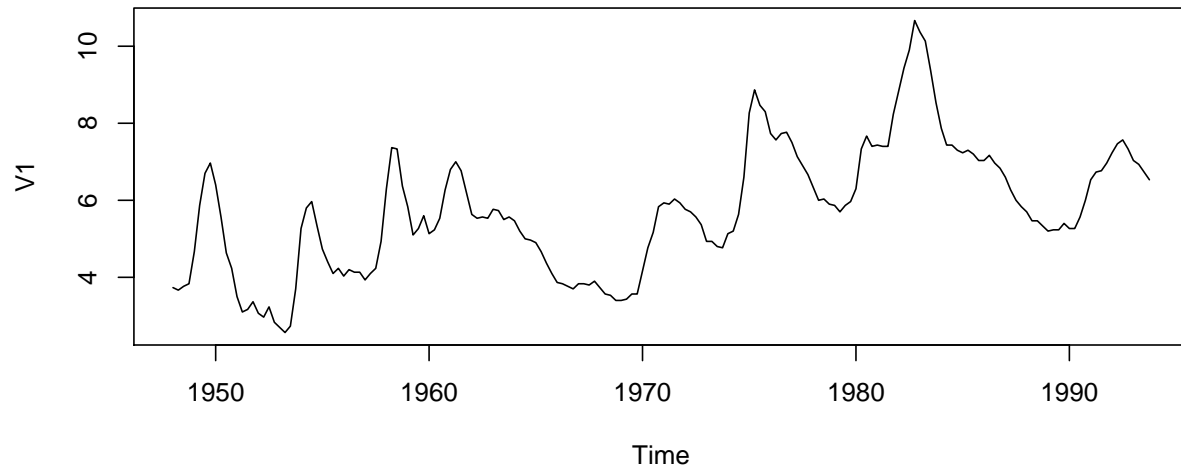


Exercise 3. Consider the seasonally adjusted quarterly U.S. unemployment rate from 1948 to 1993.

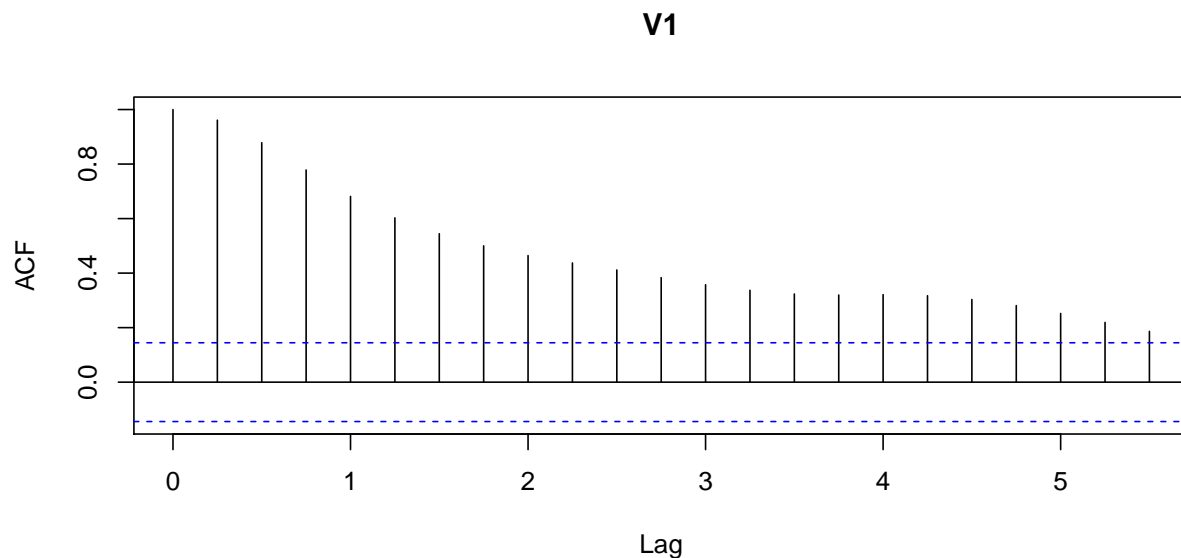
a. Build an adequate time series model for the data and check the validity of the fitted model.

At first glance, these data appear to have a linear trend. However, differencing didn't seem to improve our ACF. Because the ACF is extremely slowly decreasing, an ARIMA or ARMA model seems like a good candidate. The `auto.arima()` function selected an ARMA(2,1) model.


```
uer <- read.table("q-unemrate.txt")
uer <- ts(uer, start = 1948, frequency = 4)
plot(uer)
```



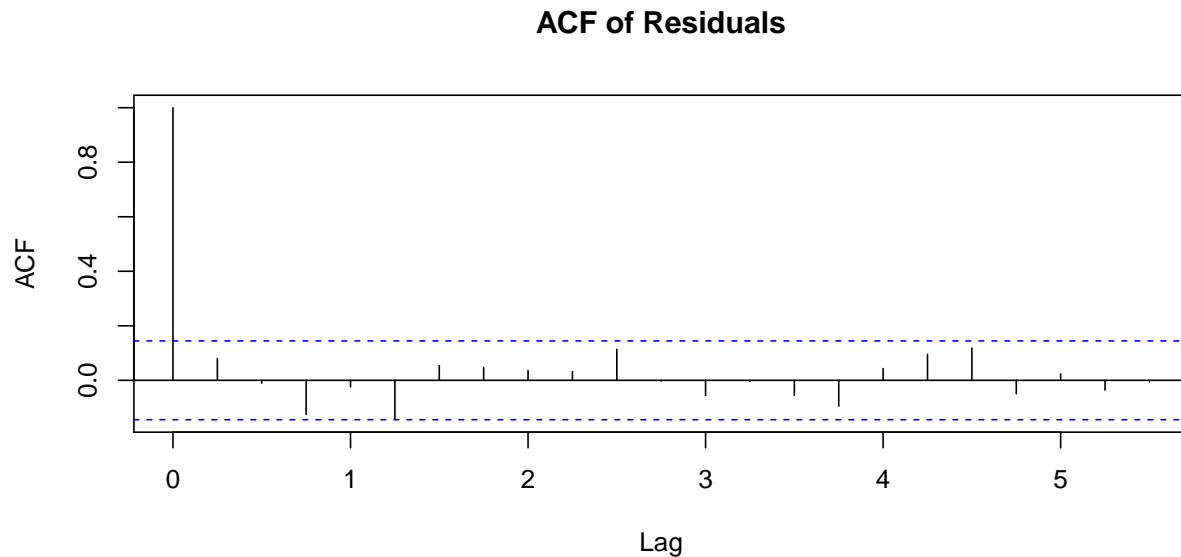
```
demean.uer <- uer - mean(uer)
acf(demean.uer)
```



```
model <- auto.arima(demean.uer, d = 0, D = 0, max.p=10, max.q=10,
  ic = 'aic', stepwise = FALSE, trace = FALSE)
```

The ACF of the residuals is quite good, and is a strong indicator that the residuals resemble white noise, which in turn indicates that the model is sufficiently describing the data. A Box-Ljung test was also performed. With a p-value of 0.2757, we have even stronger evidence that the residuals follow white noise and that the model adequately describes the data.

```
acf(model$residuals, main = "ACF of Residuals")
```



```
Box.test(model$residuals, type = "Ljung")
```

```
##
## Box-Ljung test
##
## data: model$residuals
## X-squared = 1.1906, df = 1, p-value = 0.2752
```

b. Use the fitted model to predict the quarterly unemployment rate for 1994 and 1995.

The code below obtains the predictions and the 95% confidence interval.

```
uer.predict <- predict(model, n.ahead = 8, se.fit = TRUE)
uer.predict$pred <- ts(uer.predict$pred + mean(uer), start = 1994,
                      frequency = 4)
pred_lower = uer.predict$pred - 1.96 * uer.predict$se
pred_upper = uer.predict$pred + 1.96 * uer.predict$se
```

c. Plot the predictions (with 95% confidence interval) and the observed series in the same figure.

See plot below:

```
{ts.plot(uer, uer.predict$pred, pred_lower, pred_upper,
        gpars = list(lty = c("solid", "dotted", "dashed", "dashed"),
                    main = "Unemployment Rate Forecast"))
legend(legend = c("History", "Forecast", "95% CI"), x = "topleft",
      lty = c("solid", "dotted", "dashed"))}
```

Unemployment Rate Forecast

