QMSS GR5016 - Lab3 - Times Series

Mathieu Sauterey - UNI: mjs2364

November 28, 2018

1. Create a multivariate time series; perform any interpolations.

For this study we will explore the response variable $hrs1$ which is the number of hours worked last week for each respondent. We will include the explanatory variables *degree* and *income* later when we expand our model from simple to multivariate linear regression.

For now we simply create the time series initially ranging from 1972 to 2012. As described in the code below, we interpolate all variables for the missing odd years and then exclude year 1972 from the sample as $hrs1$ has no data for this year.

```
# Reads GSS Panel Data
GSS = read.csv(file.choose())

#load packages
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 3.4.3
```

```
#library(QMSS)
library(ggplot2)
library(plyr)
library(car)
```

```
## Warning: package 'car' was built under R version 3.4.3
```

```
library(fUnitRoots)
```

```
## Warning: package 'fUnitRoots' was built under R version 3.4.4
## Loading required package: timeDate
## Loading required package: timeSeries
## Loading required package: fBasics
##
## Rmetrics Package fBasics
## Analysing Markets and calculating Basic Statistics
## Copyright (C) 2005-2014 Rmetrics Association Zurich
## Educational Software for Financial Engineering and Computational Science
## Rmetrics is free software and comes with ABSOLUTELY NO WARRANTY.
## https://www.rmetrics.org --- Mail to: info@rmetrics.org
##
## Attaching package: 'fBasics'
## The following object is masked from 'package:car':
##
##     densityPlot
```

```
library(lmtest)
```

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

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following object is masked from 'package:timeSeries':
##
##     time<-
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Create a multivariate time series ##
vars <- c("year", "hrs1", "sex", "age", "partyid", "wrkstat", "happy", "degree", "realinc")
sub <- GSS[, vars]

sub <- mutate(sub,
              baplus = ifelse(degree >= 3, 1, 0),
              happiness = ifelse(happy == 1, 1, 0),
              income = realinc)


# get means by year
by.year <- aggregate(subset(sub, sel = -year), list(year = sub$year), mean, na.rm = T)

## interpolate for some missing years ##
# add the extra years
by.year[30:40, "year"] <- c(1979, 1981, 1992, 1995, seq(1997, 2009, 2))
by.year <- arrange(by.year, year)

# make a time series object by.year.ts and interpolate using na.approx
by.year.ts <- ts(by.year)
by.year.ts <- na.approx(by.year.ts)

# calculate pct strong republican, percent fulltime, percent under 50 with BA
by.year.ts <- as.data.frame(by.year.ts)
by.year.ts <- mutate(by.year.ts,
                     happy_pct = happiness*100,
                     ba_pct = baplus*100)

# only keep starting 1973 and convert back to time series object
by.year.ts <- subset(by.year.ts, year >= 1973)
#by.year.ts <- ts(by.year.ts)

head(by.year.ts)
```

```
##   year     hrs1      sex      age  partyid  wrkstat    happy    degree
## 2 1973 39.88250 1.533910 44.18200 2.693905 3.573803 1.772000 0.9509738
## 3 1974 39.82861 1.534367 44.59134 2.598220 3.585580 1.752027 0.9986514
## 4 1975 38.96728 1.550336 44.30774 2.501010 3.575168 1.802020 0.9523170
## 5 1976 39.65997 1.553702 45.28667 2.430769 3.625083 1.784523 0.9886135
```

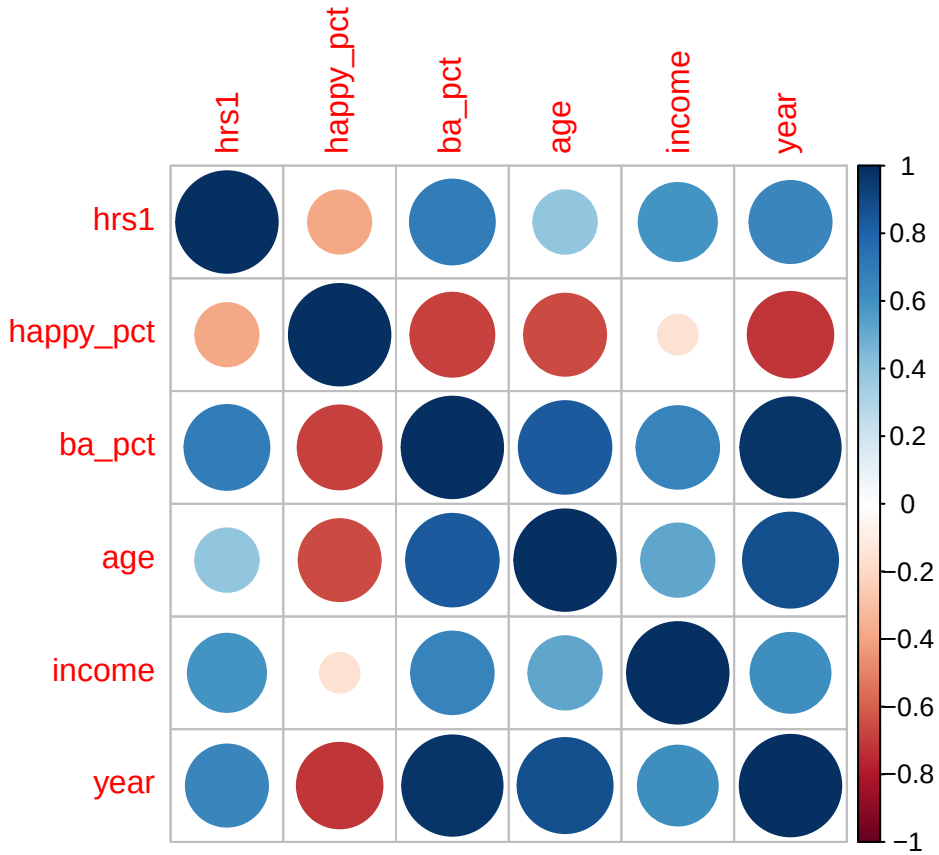
```
## 6 1977 40.53136 1.547059 44.66316 2.405797 3.228105 1.770792 0.9986877
## 7 1978 40.81170 1.580287 44.00984 2.590701 3.355744 1.752142 1.0392413
##    realinc    baplus happiness   income happy_pct   ba_pct
## 2 31362.33 0.1323036 0.3586667 31362.33  35.86667 13.23036
## 3 32124.53 0.1429535 0.3790541 32124.53  37.90541 14.29535
## 4 29403.92 0.1276024 0.3286195 29403.92  32.86195 12.76024
## 5 28273.75 0.1426658 0.3408939 28273.75  34.08939 14.26658
## 6 32640.56 0.1397638 0.3483955 32640.56  34.83955 13.97638
## 7 30178.04 0.1393067 0.3434410 30178.04  34.34410 13.93067
```

```
# correlations
cor.vars <- c("hrs1", "happy_pct", "ba_pct", "age", "income", "year")
cor.dat <- by.year.ts[, cor.vars]


# install.packages("corrplot")
library(corrplot)
```

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

```
## corrplot 0.84 loaded
```

```
corrplot(cor(cor.dat))
```



The correlation plot shows that the number of hours worked $hrs1$ is strongly positively correlated with the proportion of college educated people ba_pct, with income and with year. The correlation with age is also positive but of lower magnitude. We note that the proportion of people who consider themselves happy $happy_pct$ is negatively correlated with $hrs1$ but the magnitude of this correlation is also moderate.

2. Graph the relationships between X and Y. Explain how you think Y should relate to your key Xs.

```
library(reshape2)
```

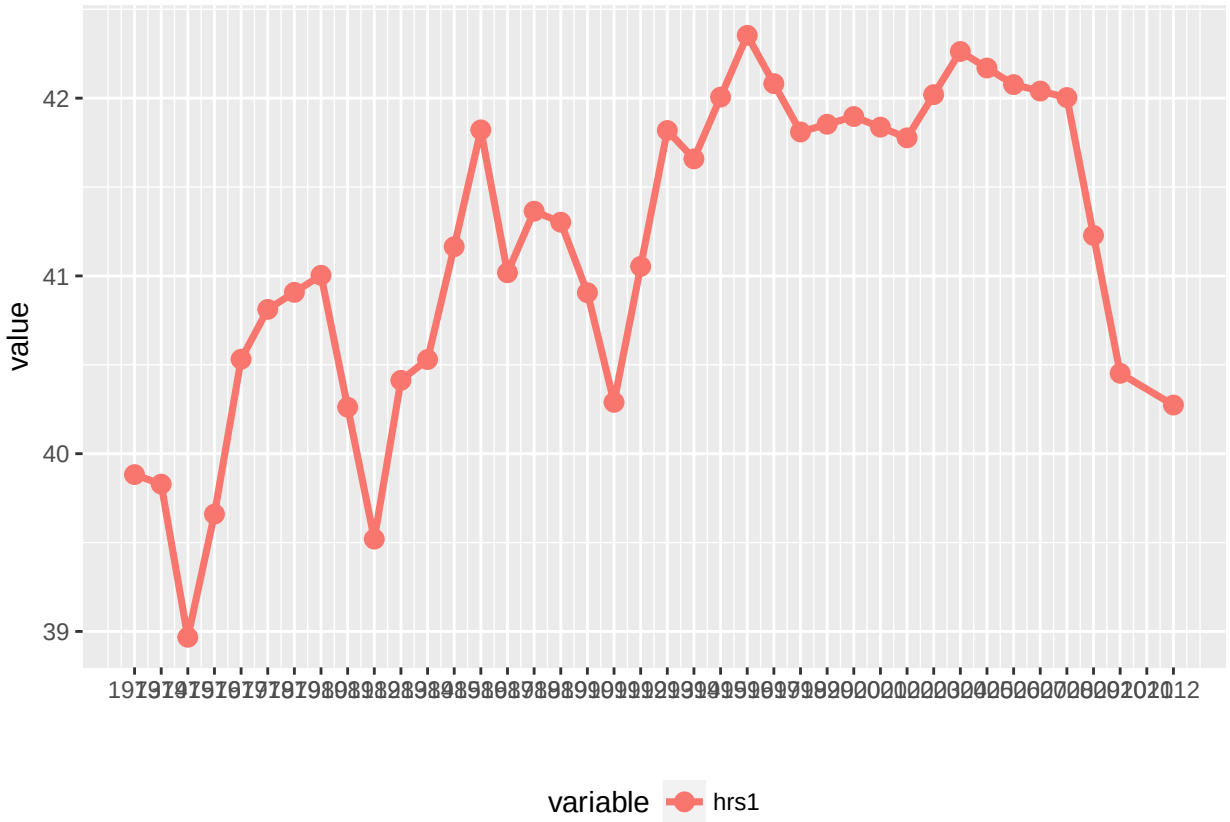
```
## Warning: package 'reshape2' was built under R version 3.4.3
```

```
meltMyTS <- function(mv.ts.object, time.var, keep.vars){
  # mv.ts.object = a multivariate ts object
  # keep.vars = character vector with names of variables to keep
  # time.var = character string naming the time variable
  require(reshape2)

  if(missing(keep.vars)) {
    melt.dat <- data.frame(mv.ts.object)
  }
  else {
    if (!(time.var %in% keep.vars)){
      keep.vars <- c(keep.vars, time.var)
    }
    melt.dat <- data.frame(mv.ts.object)[, keep.vars]
  }
  melt.dat <- melt(melt.dat, id.vars = time.var)
  colnames(melt.dat)[which(colnames(melt.dat) == time.var)] <- "time"
  return(melt.dat)
}

# Make a character vector naming the variables we might want to plot
keep.vars <- c("year", "hrs1", "happy_pct", "age", "ba_pct", "income")

# Use meltMyTS to transform the data to a 3-column dataset containing a column
# for time, a column for variable names, and a column of values corresponding to
# the variable names


plot.dat <- meltMyTS(mv.ts.object = by.year.ts, time.var = "year", keep.vars = keep.vars)
head(plot.dat)
```

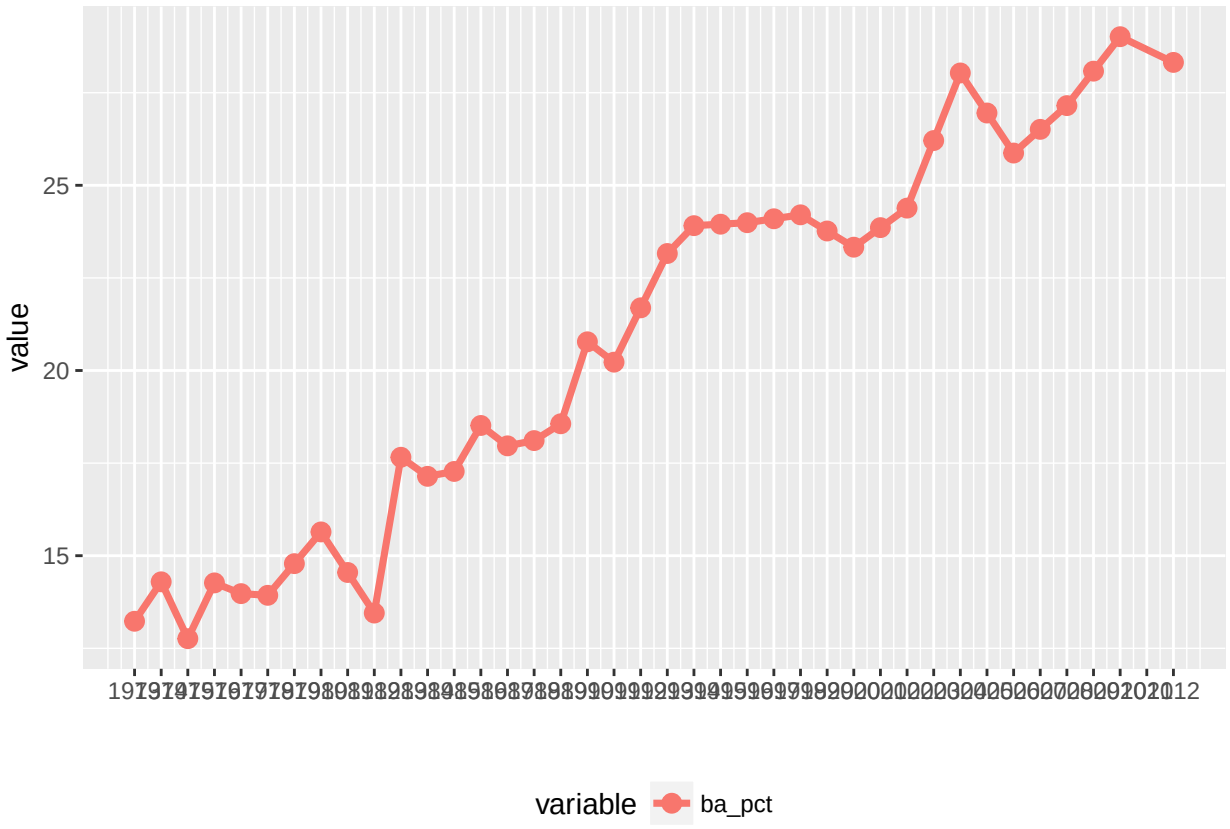
```
##   time variable    value
## 1 1973     hrs1 39.88250
## 2 1974     hrs1 39.82861
## 3 1975     hrs1 38.96728
## 4 1976     hrs1 39.65997
## 5 1977     hrs1 40.53136
## 6 1978     hrs1 40.81170
```

```
# Use ggMyTS to plot any of the variables or multiple variables together


ggMyTS <- function(df, varlist, line = TRUE, point = TRUE, pointsize = 3, linewidth = 1.25, ...){
  require(ggplot2)
  # varlist = character vector with names of variables to use
  if(missing(varlist)){
    gg <- ggplot(df, aes(time, value, colour = variable))
```

```
  }
  else{
    include <- with(df, variable %in% varlist)
    gg <- ggplot(df[include,], aes(time, value, colour = variable))
  }
  if(line == FALSE & point == FALSE) {
    stop("At least one of 'line' or 'point' must be TRUE")
  }
  else{
    if(line == TRUE) gg <- gg + geom_line(size = linewidth, aes(color = variable), ...)
    if(point == TRUE) gg <- gg + geom_point(size = pointsize, aes(color = variable), ...)
  }

  gg + xlab("") + theme(legend.position = "bottom") + scale_x_continuous(breaks = min(df$time):max(df$t
}

(g_hrs1 <- ggMyTS(df = plot.dat, varlist = c("hrs1")))
```

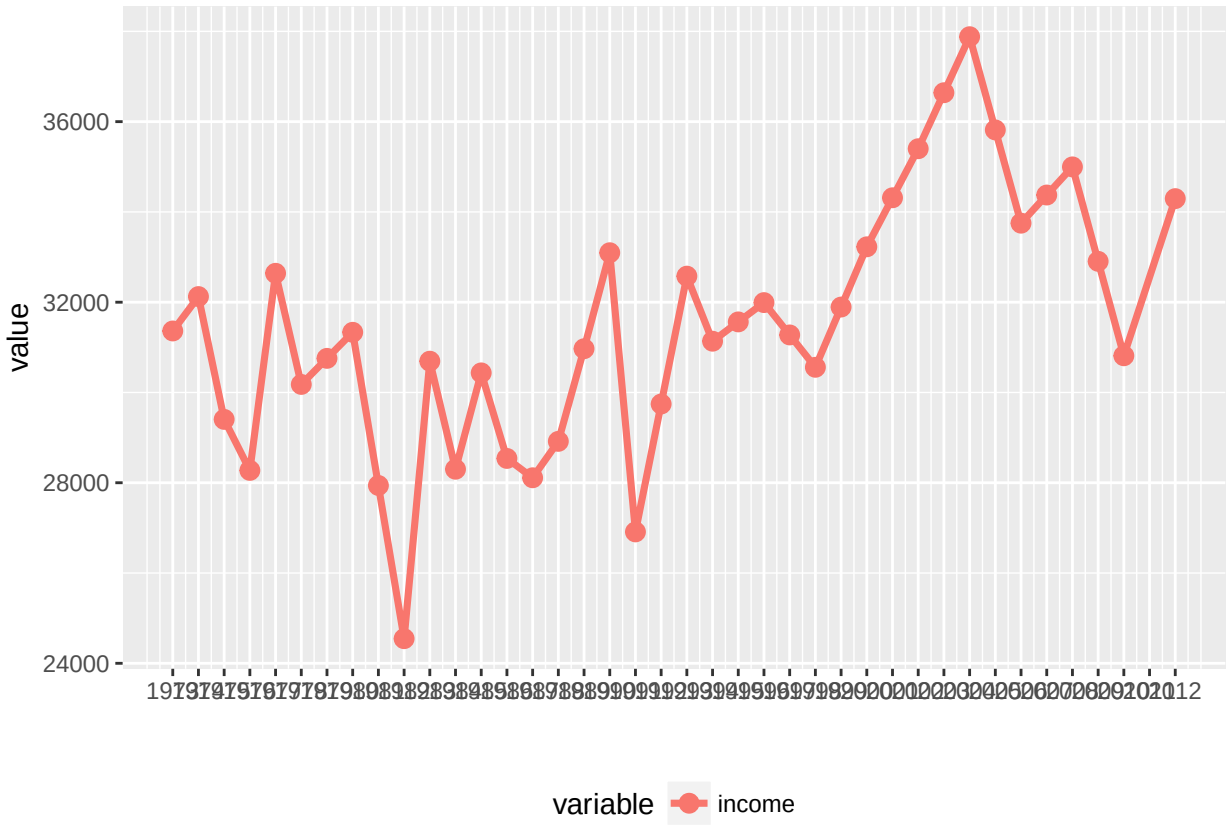



```
(g_degreelt50_pct <- ggMyTS(df = plot.dat, varlist = c("ba_pct")))
```

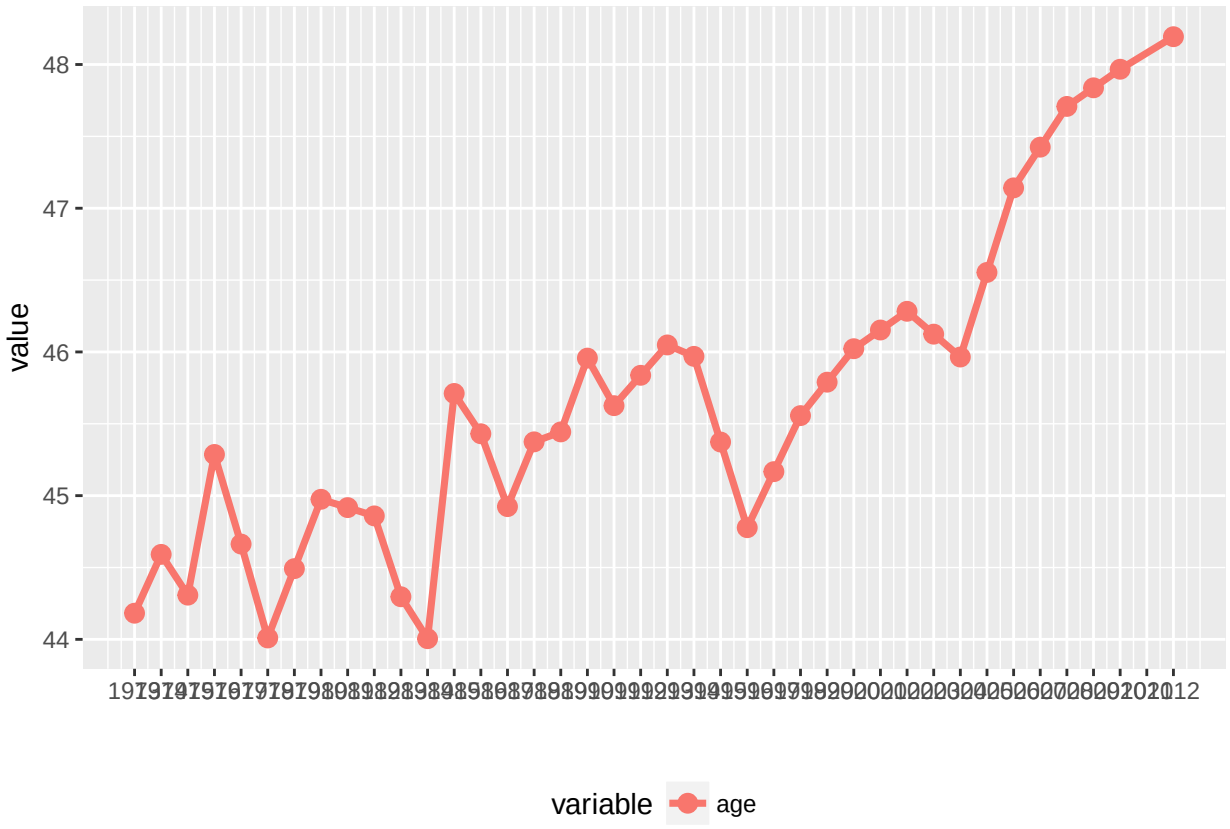



```
(g_degreelt50_pct <- ggMyTS(df = plot.dat, varlist = c("happy_pct")))
```


```
(g_degreelt50_pct <- ggMyTS(df = plot.dat, varlist = c("income")))
```




```
(g_age <- ggMyTS(df = plot.dat, varlist = c("age")))
```



The results plotted make sense as we expect average income to increase with more hours worked, and people who are more educated are also likely to work more hours. Likewise, when people work more hours we expect them to be less happy. The only surprising result is the positive correlation of $hrs1$ with age as I would have expected older people to work less hours.

3. Run a simple time series regression, with one X and no trend. Interpret it.

We regress the number of hours worked $hrs1$ on the proportion of respondents who are college educated. We then test for any heteroskedasticity and autocorrelation in the residuals.

```
# simplest regression
lm.hrs <- lm(hrs1 ~ ba_pct, data = by.year.ts)
summary(lm.hrs)
```

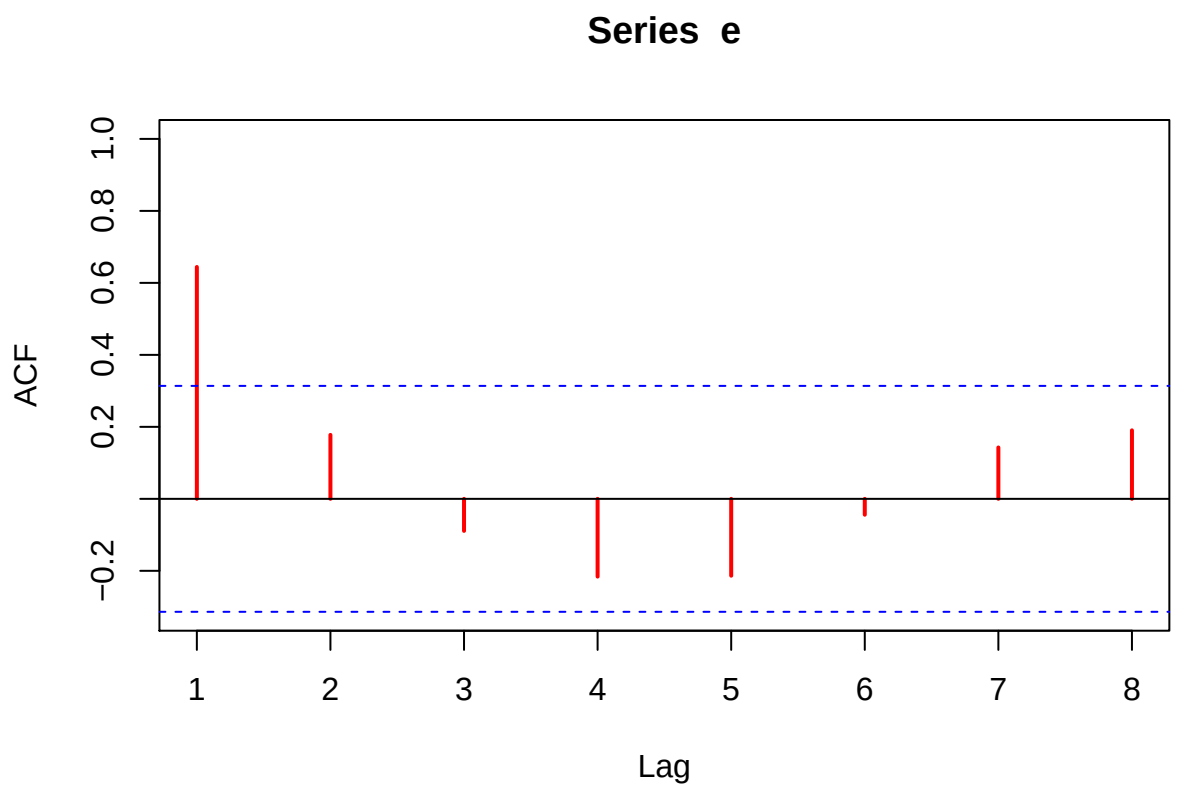
```
##
## Call:
## lm(formula = hrs1 ~ ba_pct, data = by.year.ts)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7665 -0.2945  0.2154  0.4346  0.9515
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 38.65972    0.43100  89.698  < 2e-16 ***
## ba_pct       0.11937    0.02007   5.947  7.4e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6376 on 37 degrees of freedom
## Multiple R-squared:  0.4887, Adjusted R-squared:  0.4749
## F-statistic: 35.37 on 1 and 37 DF,  p-value: 7.404e-07
```

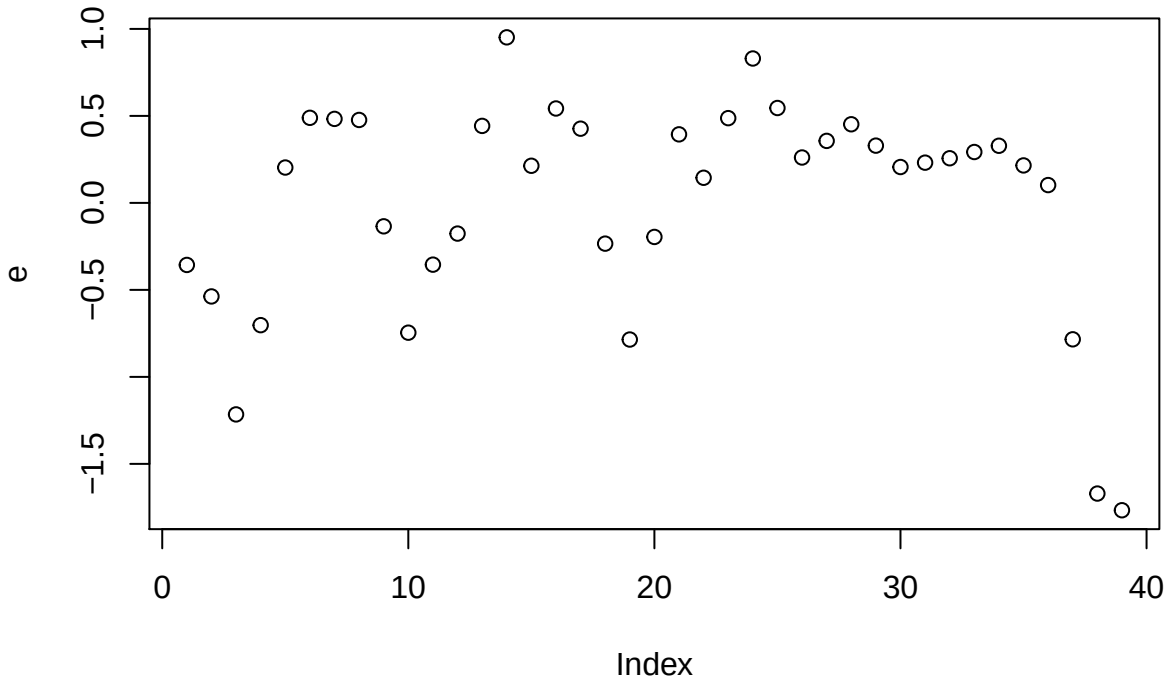
```
# test for heteroskedasticity
bptest(lm.hrs)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm.hrs
## BP = 1.5749, df = 1, p-value = 0.2095
```

```
# look for autocorrelation in errors
e <- lm.hrs$resid
acf(e, xlim = c(1,8), col = "red", lwd = 2) # can also customize acf output
```




```
plot(e) # plot residuals over time
```




```
dwtest(lm.hrs) # Durbin-Watson test
```

```
##
##  Durbin-Watson test
##
## data:  lm.hrs
## DW = 0.49557, p-value = 1.22e-09
## alternative hypothesis: true autocorrelation is greater than 0
```

```
bgtest(lm.hrs) # Breusch-Godfrey test
```

```
##
##  Breusch-Godfrey test for serial correlation of order up to 1
##
## data:  lm.hrs
## LM test = 20.777, df = 1, p-value = 5.16e-06
```

```
durbinWatsonTest(lm.hrs, max.lag=3) # Durbin-Watson with more lags
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.64424519     0.4955666   0.000
##    2      0.17774667     1.2237609   0.022
##    3     -0.08911836     1.6183515   0.314
##  Alternative hypothesis: rho[lag] != 0
```

The percent of people college educated ba_pct is statistically significant. Any increase by 1 in the percent of college educated people causes an increase of 0.11937 in the number of hours worked per week. The p-value of the Bresuh-Pagan test is > 0.05 so there is no heteroskedasticity apparent in this model. The ACF plot

shows a first-order and second-order autocorrelation of the residuals. This conclusion of autocorrelation is supported by the plot of residuals over time which shows a clear pattern, and by the Durbin-Watson test which is significant for lag 1 and 2. The Breusch-Godfrey test is essentially 0 and thus indicates strong serial correlation.

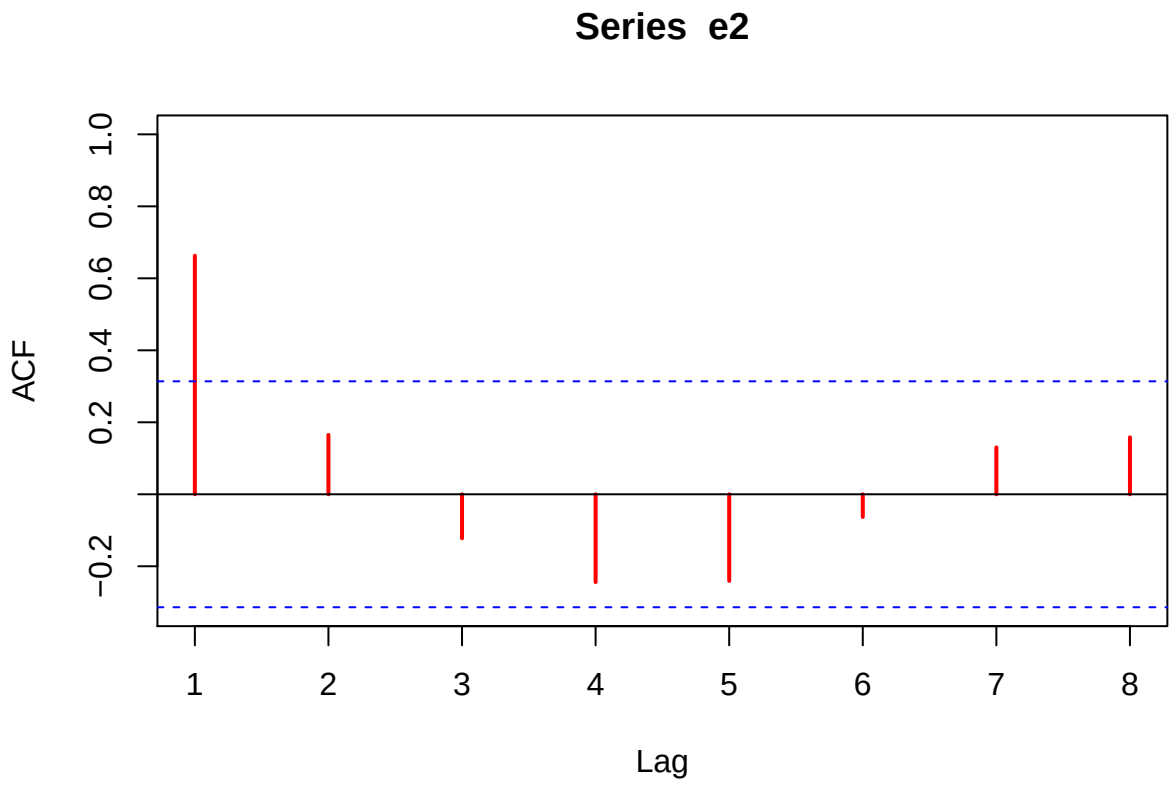
4. Run a time series regression with one X and trend. Interpret it. Perform autocorrelation diagnostics.

Explain what you found.

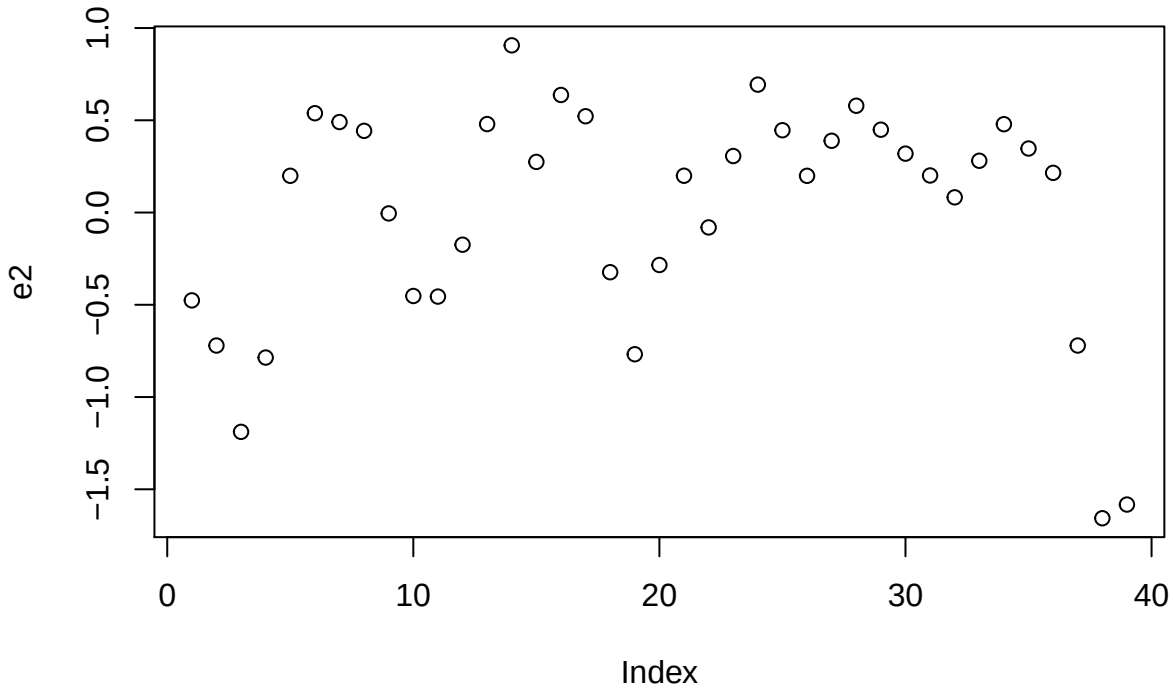
```
# include year trend
lm.hrs2 <- update(lm.hrs, ~ . + year)
summary(lm.hrs2)
```

```
##
## Call:
## lm(formula = hrs1 ~ ba_pct + year, data = by.year.ts)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6568 -0.3879  0.2008  0.4476  0.9062
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 133.10013   82.05501   1.622   0.1135
## ba_pct        0.22466    0.09363   2.399   0.0217 *
## year         -0.04851    0.04215  -1.151   0.2573
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6348 on 36 degrees of freedom
## Multiple R-squared:  0.5069, Adjusted R-squared:  0.4795
## F-statistic:  18.5 on 2 and 36 DF,  p-value: 2.973e-06
```

```
# look for autocorrelation
e2 <- lm.hrs2$resid
acf(e2, xlim = c(1,8), col = "red", lwd = 2)
```




```
plot(e2)
```




```
dwtest(lm.hrs2)
```

```
##
##  Durbin-Watson test
##
## data:  lm.hrs2
## DW = 0.48597, p-value = 4.212e-10
## alternative hypothesis: true autocorrelation is greater than 0
```

```
bgtest(lm.hrs2)
```

```
##
##  Breusch-Godfrey test for serial correlation of order up to 1
##
## data:  lm.hrs2
## LM test = 21.305, df = 1, p-value = 3.917e-06
```

```
durbinWatsonTest(lm.hrs2, max.lag=3)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.6628693     0.4859696   0.000
##    2       0.1649375     1.2567845   0.016
##    3      -0.1223727     1.6981809   0.410
##  Alternative hypothesis: rho[lag] != 0
```

After including the *year* variable we find that there is no significant time trend, and the percent of college educated people remains significant. Thus Education and Number of hours worked seem causally related and do not seem to share any underlying time trend. Net of time trend, any increase by 1 in the percent of college

educated people causes an increase of 0.22466 in the number of hours worked per week. The R-squared of this second model is pretty much the same as the first one. Moreover, inspecting the ACF plot shows weaker AR(1) and AR(2) autocorrelation, but there is still some autocorrelation of lag 1 and 2 in the model as proven by the small p-value of the Breusch-Godfrey and Durbin-Watson tests.

5. Consider running a time series regression with many Xs and trend. Interpret that. Check VIF.

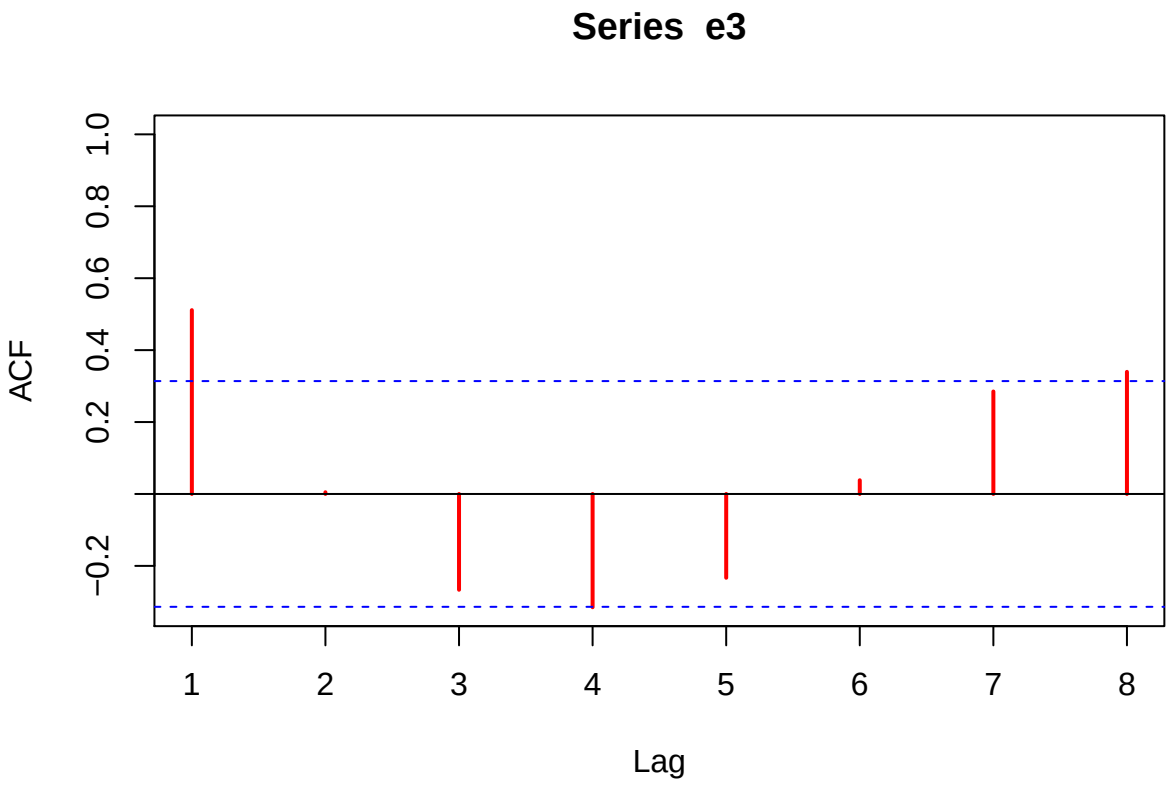
```
# add some more predictors
lm.hrs3 <- update(lm.hrs2, ~ . + age + income + happy_pct)
summary(lm.hrs3)
```

```
##
## Call:
## lm(formula = hrs1 ~ ba_pct + year + age + income + happy_pct,
##     data = by.year.ts)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.25936 -0.33141  0.08802  0.29850  1.24715
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.769e+00  8.591e+01  -0.090  0.92849
## ba_pct       1.119e-01  9.264e-02   1.208  0.23550
## year         3.526e-02  4.574e-02   0.771  0.44630
## age         -5.593e-01  1.791e-01  -3.123  0.00371 **
## income       7.455e-05  5.285e-05   1.411  0.16771
## happy_pct   -1.445e-02  6.835e-02  -0.211  0.83382
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5684 on 33 degrees of freedom
## Multiple R-squared:  0.6375, Adjusted R-squared:  0.5826
## F-statistic: 11.61 on 5 and 33 DF,  p-value: 1.606e-06
```

```
vif(lm.hrs3) # variance inflation factor
```

```
##    ba_pct      year       age    income happy_pct
## 26.795534 32.236792  4.636867  2.559266  2.850274
```

```
e3 <- lm.hrs3$resid
acf(e3, xlim = c(1,8), col = "red", lwd = 2)
```




```
durbinWatsonTest(lm.hrs3, max.lag=2)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1     0.511197192      0.812590   0.000
##    2     0.004789839      1.711411   0.222
##  Alternative hypothesis: rho[lag] != 0
```

We included the variables age, income and happiness (happy_pct) in the model which resulted in making all variables non-significant except age. Henceforth our previous hypothesis that education and number of hours worked per week are causally related is rejected by this model. Moreover, the correlation table and our previous visual inspection of the time series graphs showed that number of hours worked and age were positively correlated, but this model assigns a negative sign to the relationship which is highly surprising. The R-squared of the this model is 0.1 larger compared to the simple model and simple model with trend. The VIF is larger than 10 for the percent of college educated people (ba_pct) and year, thus we conclude that there is multicolinearity between these two variable. Finally, the ACF plot shows that there is only AR(1) left, no more AR(2), and this is confirmed by the large p-value for the lag 2 autocorrelation in the Durbin-Watson test.

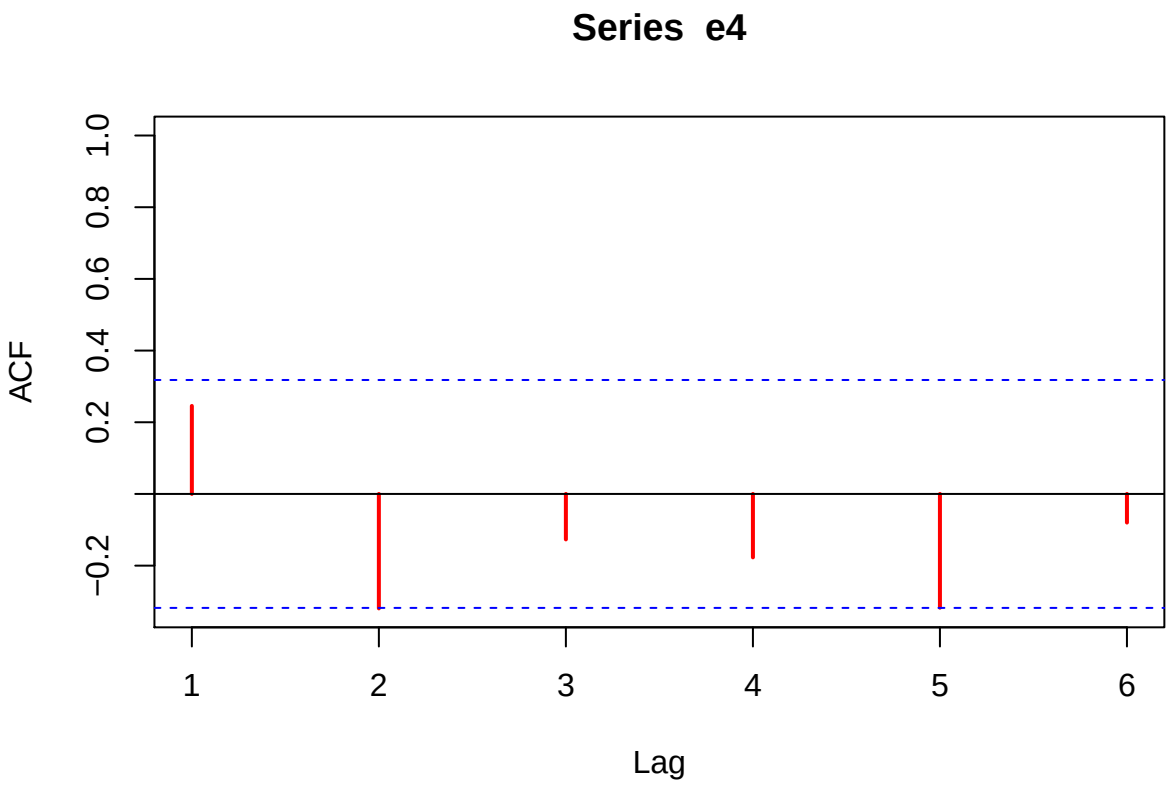
6. Run a first differenced time series regression. Interpret that.

```
firstD <- function(var, group, df){
  bad <- (missing(group) & !missing(df))
  if (bad) stop("if df is specified then group must also be specified")

  fD <- function(j){ c(NA, diff(j)) }
```

```
  var.is.alone <- missing(group) & missing(df)

  if (var.is.alone) {
    return(fD(var))
  }
  if (missing(df)){
    V <- var
    G <- group
  }
  else{
    V <- df[, deparse(substitute(var))]
    G <- df[, deparse(substitute(group))]
  }

  G <- list(G)
  D.var <- by(V, G, fD)
  unlist(D.var)
}

## Use the first differences
by.yearFD <- summarise(data.frame(by.year.ts),
                       hrs1 = firstD(hrs1), # using firstD functon from QMSS package
                       age = firstD(age),
                       ba_pct = firstD(ba_pct),
                       happy_pct = firstD(happy_pct),
                       income = firstD(income),
                       year = year)

lm.hrs4 <- update(lm.hrs2, data = by.yearFD)
summary(lm.hrs4)
```

```
##
## Call:
## lm(formula = hrs1 ~ ba_pct + year, data = by.yearFD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81233 -0.21900  0.02794  0.31481  0.86768
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.678291  12.782999   1.383  0.17544
## ba_pct       0.210864   0.063644   3.313  0.00215 **
## year        -0.008909   0.006415  -1.389  0.17368
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4354 on 35 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.2708, Adjusted R-squared:  0.2291
## F-statistic: 6.498 on 2 and 35 DF,  p-value: 0.003982
```

```
e4 <- lm.hrs4$resid
acf(e4, xlim = c(1,6), col = "red", lwd = 2)
```



We return to our previous simple model with trend. After taking the first difference, the year trend is still non-significant and the percent of college educated people is still significant. For each 1 percentage point change in people being college educated, average number of hours worked per week increases by 0.21086, at any point in time. As seen on the ACF plot, there is no autocorrelation left in the first difference model.

7. Check your variables for unit roots. Do some tests. Interpret them.

```
adfTest(by.year.ts[,"hrs1"], lags = 0, type="ct")
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 0
##   STATISTIC:
##     Dickey-Fuller: -1.4523
##   P VALUE:
##     0.7879
##
```

```
## Description:
##  Wed Nov 28 18:59:23 2018 by user: mathe
```

```
adfTest(by.year.ts[,"hrs1"], lags = 4, type="ct")
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 4
##   STATISTIC:
##     Dickey-Fuller: -0.8312
##   P VALUE:
##     0.9504
##
## Description:
##  Wed Nov 28 18:59:23 2018 by user: mathe
```

```
# Phillips-Perron test
PP.test(by.year.ts[,"hrs1"],lshort=TRUE)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  by.year.ts[, "hrs1"]
## Dickey-Fuller = -1.6063, Truncation lag parameter = 3, p-value =
## 0.7276
```

P-values with 4 lags, with a trend and a drift, are too high to be able to reject the null of Unit Root, therefore, we might have a unit roots here. This is confirmed by the p-value of the Phillips-Perron test that is too high to reject the null of No Unit Root.

8. Perform an Automatic ARIMA on the residuals from one of your earlier models. Tell me what it says.

```
library(forecast)
```

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

```
auto.arima(e2, trace=TRUE)
```

```
##
##  ARIMA(2,0,2) with non-zero mean : Inf
##  ARIMA(0,0,0) with non-zero mean : 76.44036
##  ARIMA(1,0,0) with non-zero mean : 49.08894
##  ARIMA(0,0,1) with non-zero mean : 44.29868
##  ARIMA(0,0,0) with zero mean     : 74.21513
##  ARIMA(1,0,1) with non-zero mean : 36.02747
##  ARIMA(1,0,2) with non-zero mean : 38.3654
##  ARIMA(1,0,1) with zero mean     : 33.67662
##  ARIMA(0,0,1) with zero mean     : 41.99497
##  ARIMA(2,0,1) with zero mean     : 35.61507
```

```
##  ARIMA(1,0,0) with zero mean     : 47.06995
##  ARIMA(1,0,2) with zero mean     : 35.8242
##  ARIMA(2,0,2) with zero mean     : Inf
##
##  Best model: ARIMA(1,0,1) with zero mean
```

```
## Series: e2
## ARIMA(1,0,1) with zero mean
##
## Coefficients:
##          ar1     ma1
##       0.5515  0.7877
## s.e.  0.1487  0.1139
##
## sigma^2 estimated as 0.117:  log likelihood=-13.5
## AIC=32.99   AICc=33.68   BIC=37.98
```

We perform ARIMA on the simple model with trend that had AR(1) and AR(2). Auto.arima suggests that the errors from the simple model with trend should have an AR(1) and MA(1) correction but no first differences.

9. Run an ARIMA that follows from Step 8. Interpret that, too.

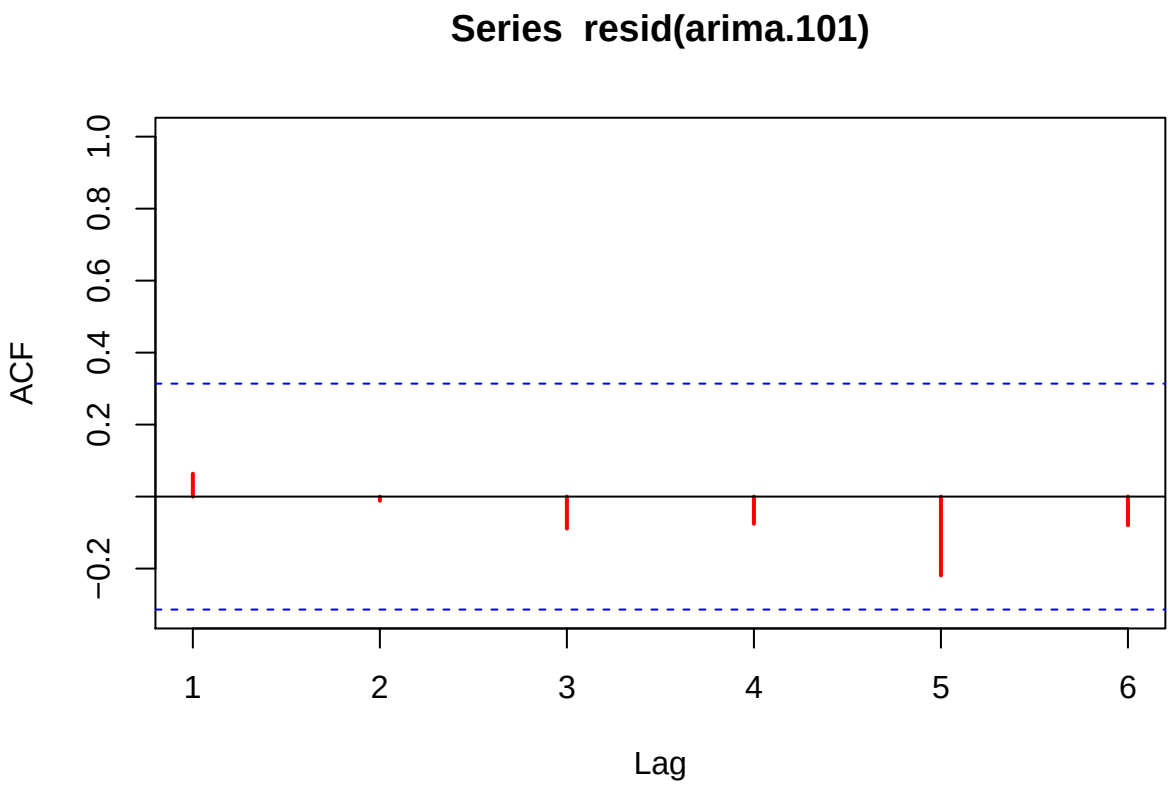
```
xvars.fat <- by.year.ts[,c("ba_pct", "year")]

# ARIMA(0,0,0) = OLS
arima.101 <- arima(by.year.ts[,"hrs1"], order = c(1,0,1), xreg = xvars.fat)
summary(arima.101)
```

```
##
## Call:
## arima(x = by.year.ts[, "hrs1"], order = c(1, 0, 1), xreg = xvars.fat)
##
## Coefficients:
##          ar1     ma1  intercept  ba_pct     year
##       0.5522  0.7896   160.3917  0.2457  -0.0625
## s.e.  0.1487  0.1168    48.7853  0.0435   0.0248
##
## sigma^2 estimated as 0.1096:  log likelihood = -13.26,  aic = 38.53
##
## Training set error measures:
##                       ME      RMSE       MAE        MPE      MAPE
## Training set 0.008097659 0.3311065 0.2548881 0.01301156 0.6225971
##                    MASE       ACF1
## Training set 0.6646559 0.06346706
```

```
Box.test(resid(arima.101), lag = 20, type = c("Ljung-Box"), fitdf = 0)
```

```
##
##  Box-Ljung test
##
## data:  resid(arima.101)
## X-squared = 13.528, df = 20, p-value = 0.8536
```

```
acf(resid(arima.101), xlim = c(1,6), col = "red", lwd = 2)
```



Net of the time trend, of the first lag and first lagged average of the number of hours worked (hrs1), each percent more of people with college degrees increases the number of weekly work hours by 0.2457. The very large p-value of the Box-Ljung test shows that there is no autocorrelation left at all. This is confirmed by the ACF plot that shows no autocorrelation at all.