Lesson 14

Christopher A. Swenson (chris@cswenson.com)

11/27/2021

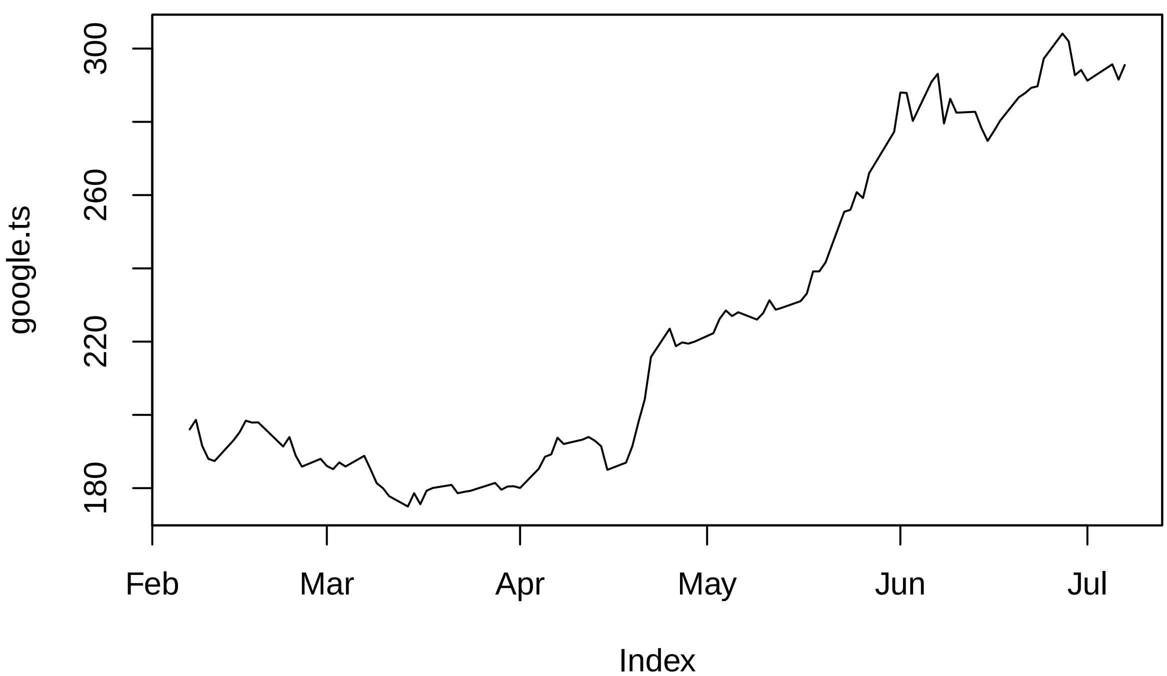
Google stock (autoregression model)

Use the read.zoo function in the zoo package to load the google_stock data in time series format. Create a time series plot of the data. Load the google_stock data in the usual way using read-table. Use the ts function to convert the price variable to a time series. Create a plot of partial autocorrelations of price. Calculate a lag-1 price variable (note that the lag argument for the function is −1, not +1). Create a scatterplot of price vs lag1price. Use the ts.intersect function to create a dataframe containing price and lag1price. Fit a simple linear regression model of price vs lag1price (a first-order autoregression model).

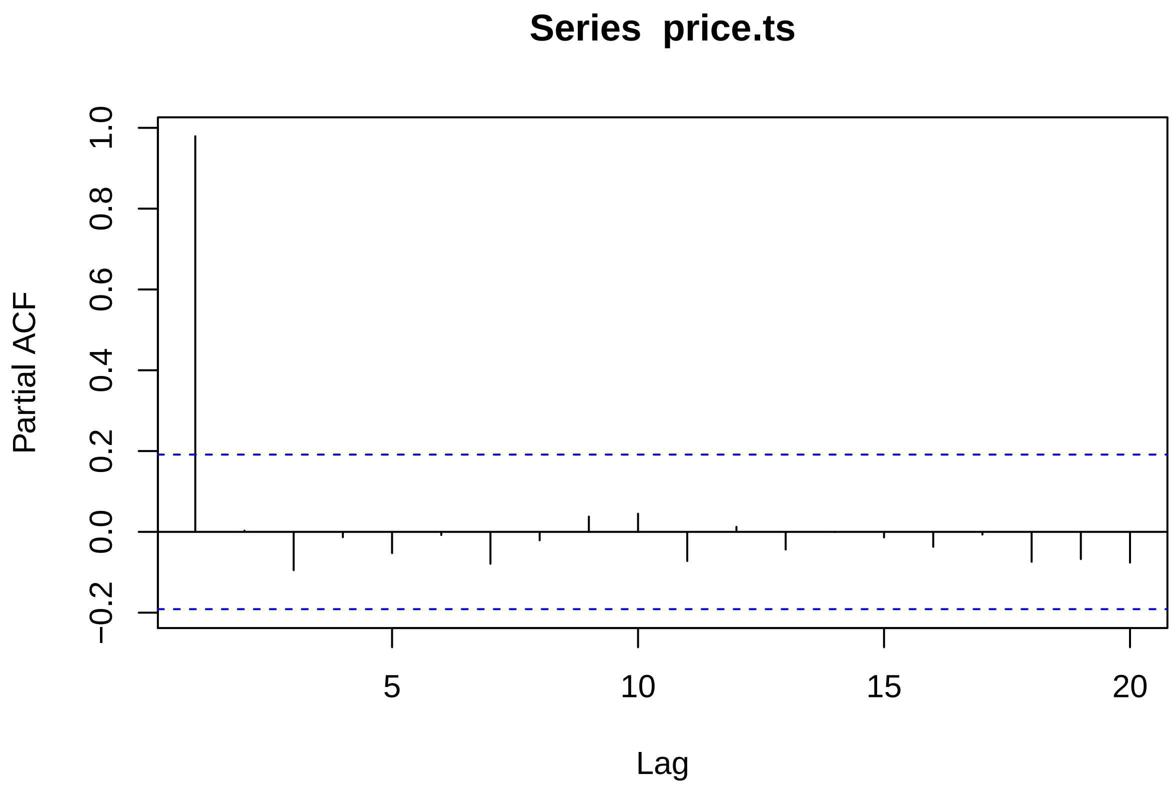
```
library(zoo)
```

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

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

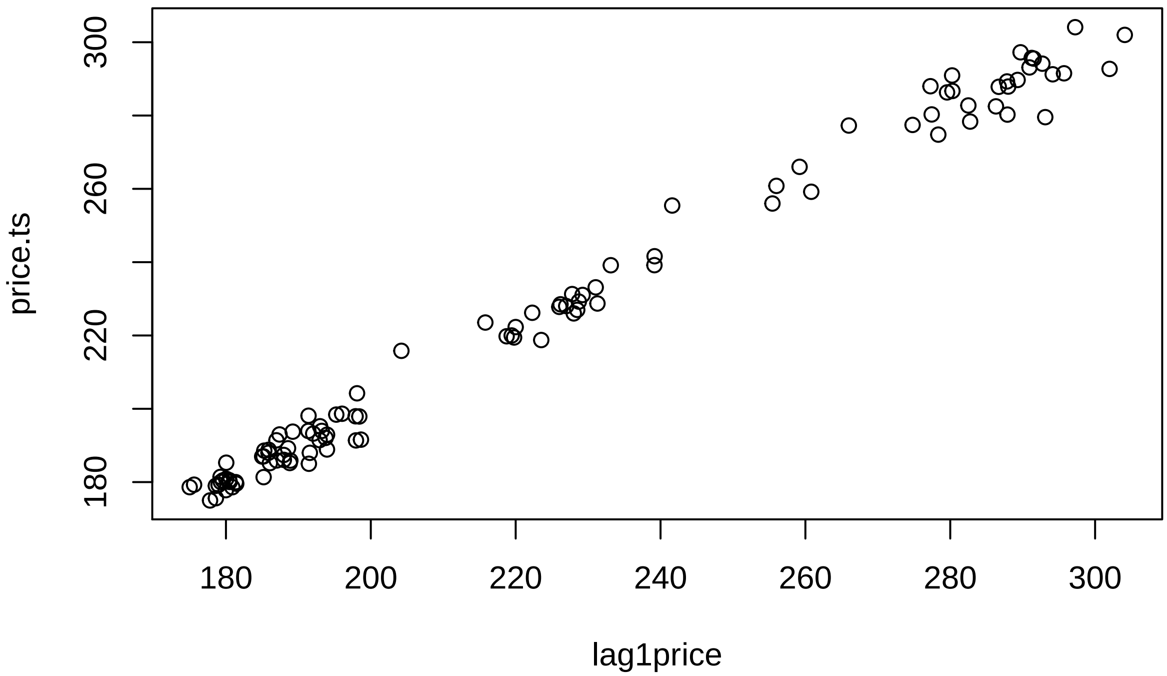
```
google.ts <- read.zoo("./Data/google_stock.txt", format="%m/%d/%Y",
                      header=T)
plot(google.ts)
```



```
google <- read.table("./Data/google_stock.txt", header=T)
attach(google)

price.ts <- ts(price)
pacf(price.ts)
```




```
lag1price <- lag(price.ts, -1)
plot(price.ts ~ lag1price, xy.labels=F)
```




```
lagdata <- ts.intersect(price.ts, lag1price, dframe=T)
summary(lm(price.ts ~ lag1price, data=lagdata))
```

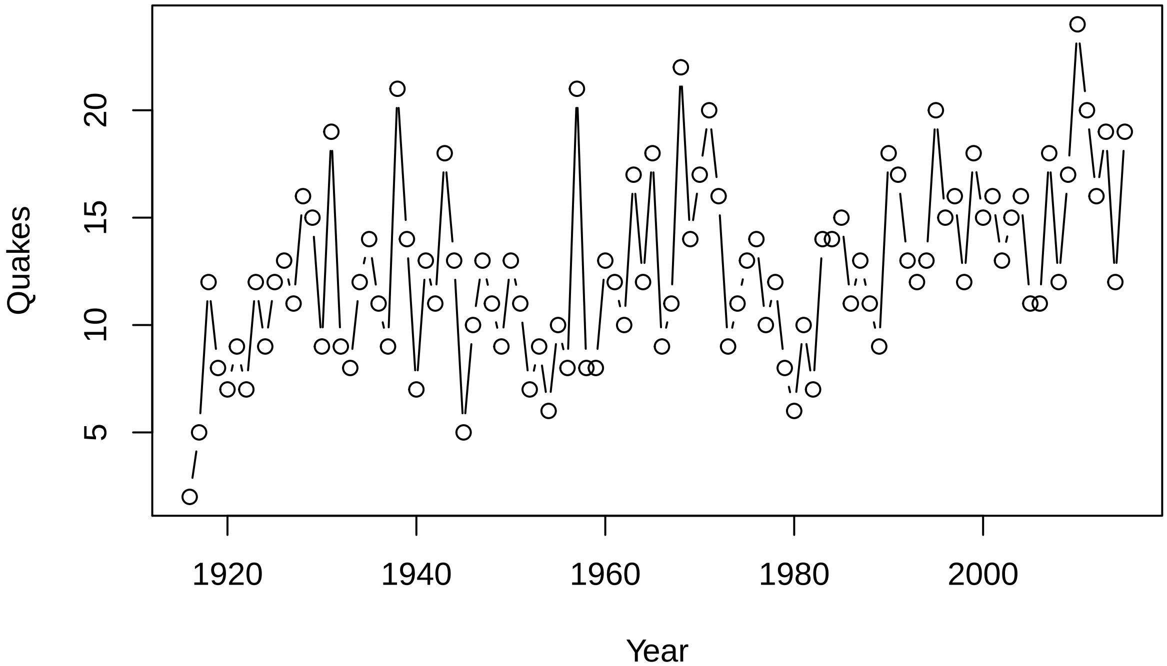
```
##
## Call:
## lm(formula = price.ts ~ lag1price, data = lagdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.9224  -2.5863   0.0049   2.4220  12.7800
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.35845    2.35289  -0.152    0.879
## lag1price    1.00587    0.01032  97.476   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.463 on 102 degrees of freedom
## Multiple R-squared:  0.9894, Adjusted R-squared:  0.9893
## F-statistic:  9502 on 1 and 102 DF,  p-value: < 2.2e-16
```

```
detach(google)
```

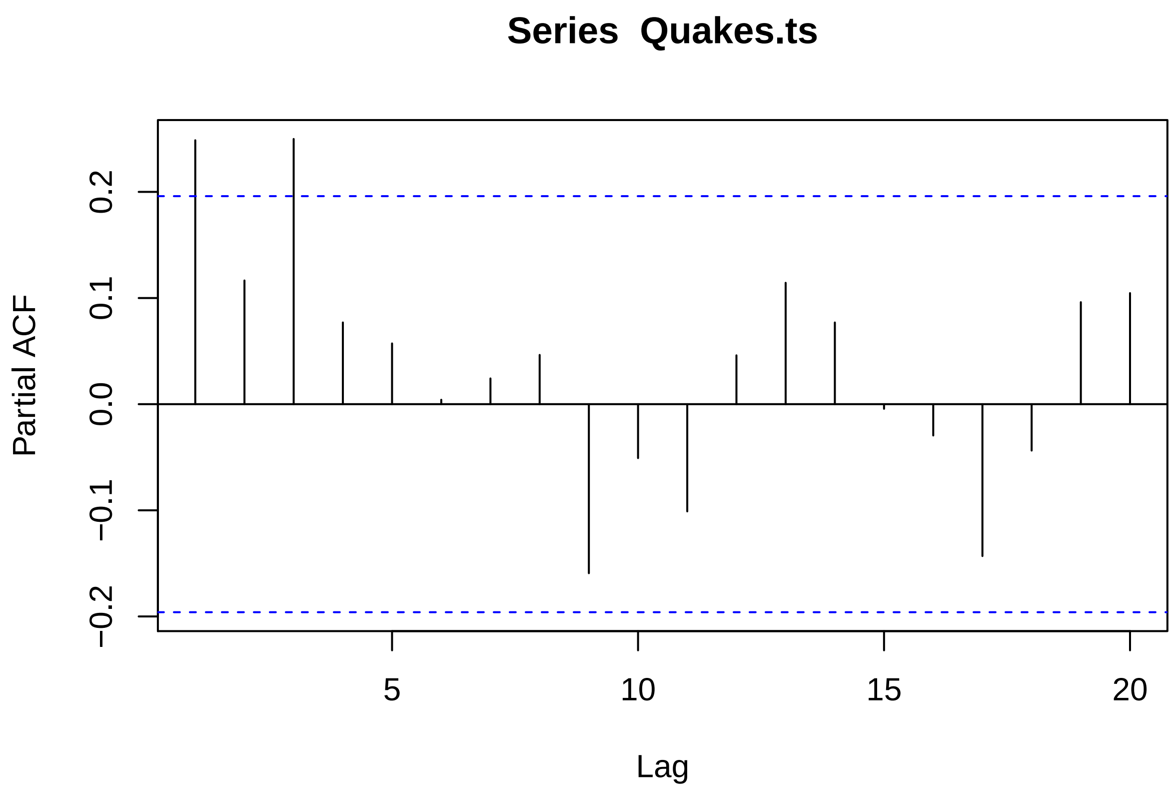
Earthquakes (autoregression model)

Load the earthquakes data. Create a time series plot of the data. Use the ts function to convert the Quakes variable to a time series. Create a plot of partial autocorrelations of Quakes. Calculate lag-1, lag-2, and lag-3 Quakes variables. Use the ts.intersect function to create a dataframe containing Quakes and the three lag variables. Fit a multiple linear regression model of Quakes versus the three lag variables (a third-order autoregression model).

```r
earthquakes <- read.table("./Data/earthquakes.txt", header=T)
attach(earthquakes)

plot(Year, Quakes, type="b")
```




```r
Quakes.ts <- ts(Quakes)
pacf(Quakes.ts)
```



```r
lag1Quakes <- lag(Quakes.ts, -1)
lag2Quakes <- lag(Quakes.ts, -2)
lag3Quakes <- lag(Quakes.ts, -3)

lagdata <- ts.intersect(Quakes.ts, lag1Quakes, lag2Quakes, lag3Quakes, dframe=T)
summary(lm(Quakes.ts ~ lag1Quakes + lag2Quakes + lag3Quakes, data=lagdata))
```

```
##
## Call:
## lm(formula = Quakes.ts ~ lag1Quakes + lag2Quakes + lag3Quakes,
##     data = lagdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8289 -2.8749 -0.6899  2.1616 10.9087
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.44916    1.78646   3.610 0.000496 ***
## lag1Quakes   0.16424    0.10063   1.632 0.106049
## lag2Quakes   0.07125    0.10128   0.703 0.483517
## lag3Quakes   0.26928    0.09783   2.753 0.007110 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.851 on 93 degrees of freedom
## Multiple R-squared:  0.1388, Adjusted R-squared:  0.111
## F-statistic: 4.997 on 3 and 93 DF,  p-value: 0.002942
```

```r
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept)  6.44916    1.78646   3.610 0.000496 ***
# lag1Quakes   0.16424    0.10063   1.632 0.106049
# lag2Quakes   0.07125    0.10128   0.703 0.483517
# lag3Quakes   0.26928    0.09783   2.753 0.007110 **
# ---
# Residual standard error: 3.851 on 93 degrees of freedom
# Multiple R-squared:  0.1388,  Adjusted R-squared:  0.111
# F-statistic: 4.997 on 3 and 93 DF,  p-value: 0.002942

detach(earthquakes)
```

Blaisdell company (regression with autoregressive errors)

Load the blaisdell data. Fit a simple linear regression model of comsales vs indsales. Use the dwt function in the car package to conduct the Durbin-Watson test on the residuals. Conduct the Ljung-Box test on the residuals. Perform the Cochrane-Orcutt procedure to transform the variables. Forecast comsales for period 21 when indsales are projected to be $175.3 million. Perform the Hildreth-Lu procedure to transform the variables. Perform the first differences procedure to transform the variables.

```r
blaisdell <- read.table("./Data/blaisdell.txt", header=T)
attach(blaisdell)

model.1 <- lm(comsales ~ indsales)
summary(model.1)
```

```
##
## Call:
## lm(formula = comsales ~ indsales)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.149142 -0.054399 -0.000454  0.046425  0.163754
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.454750   0.214146  -6.793 2.31e-06 ***
## indsales     0.176283   0.001445 122.017  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08606 on 18 degrees of freedom
## Multiple R-squared:  0.9988, Adjusted R-squared:  0.9987
## F-statistic: 1.489e+04 on 1 and 18 DF,  p-value: < 2.2e-16
```

```
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept) -1.454750   0.214146  -6.793 2.31e-06 ***
# indsales     0.176283   0.001445 122.017  < 2e-16 ***

# Durbin-Watson Test
library(car)
```

```
## Loading required package: carData
```

```
dwt(model.1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.6260046     0.7347256       0
##  Alternative hypothesis: rho != 0
```

```
# lag Autocorrelation D-W Statistic p-value
#   1       0.6260046     0.7347256       0
# Alternative hypothesis: rho != 0

# Ljung-Box Q Test
Box.test(residuals(model.1), lag = 1, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  residuals(model.1)
## X-squared = 9.0752, df = 1, p-value = 0.002591
```

```
# Box-Ljung test
# data:  residuals(model.1)
# X-squared = 9.0752, df = 1, p-value = 0.002591

# Cochrane-Orcutt Procedure
res.ts <- ts(residuals(model.1))
lag1res <- lag(res.ts, -1)
lagdata1 <- ts.intersect(res.ts, lag1res)
acp <- coef(lm(res.ts ~ lag1res -1, data=lagdata1)) # 0.6311636
y.ts <- ts(comsales)
```

```r
x.ts <- ts(indsales)
lag1y <- lag(y.ts, -1)
lag1x <- lag(x.ts, -1)
y.co <- y.ts-acp*lag1y
x.co <- x.ts-acp*lag1x
model.2 <- lm(y.co ~ x.co)
summary(model.2)
```

```
##
## Call:
## lm(formula = y.co ~ x.co)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.097039 -0.056815  0.009902  0.034553  0.125048
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.394111   0.167230  -2.357   0.0307 *
## x.co         0.173758   0.002957  58.767   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06715 on 17 degrees of freedom
## Multiple R-squared:  0.9951, Adjusted R-squared:  0.9948
## F-statistic:  3454 on 1 and 17 DF,  p-value: < 2.2e-16
```

```r
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept) -0.394111   0.167230  -2.357   0.0307 *
# x.co         0.173758   0.002957  58.767   <2e-16 ***

dwt(model.2)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.1473569      1.650248    0.28
##  Alternative hypothesis: rho != 0
```

```r
# lag Autocorrelation D-W Statistic p-value
#   1       0.1473569      1.650248   0.306
# Alternative hypothesis: rho != 0

b0 <- coef(model.2)[1]/(1-acp) # -1.068524
sqrt(vcov(model.2)[1,1])/(1-acp) # se = 0.4533986
```

```
##   lag1res
## 0.4533986
```

```r
b1 <- coef(model.2)[2] # 0.1737583

fit.20 <- b0+b1*indsales[20] # 28.76577
res.20 <- comsales[20]-fit.20 # 0.01422919
fit.21 <- b0+b1*175.3 # 29.3913
forecast.21 <- fit.21+acp*res.20 # 29.40028

# Hildreth-Lu Procedure
sse <- vector()
```

```r
for(i in 1:90){
  y.hl = y.ts-(0.09+0.01*i)*lag1y
  x.hl = x.ts-(0.09+0.01*i)*lag1x
  sse[i] <- sum(residuals(lm(y.hl ~ x.hl))^2)
}
acp <- 0.09+0.01*which.min(sse) # 0.96
y.hl = y.ts-acp*lag1y
x.hl = x.ts-acp*lag1x
model.3 <- lm(y.hl ~ x.hl)
summary(model.3)
```

```
##
## Call:
## lm(formula = y.hl ~ x.hl)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.11494 -0.04399  0.01113  0.03968  0.13951
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.07117    0.05797   1.228    0.236
## x.hl         0.16045    0.00684  23.458 2.18e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06493 on 17 degrees of freedom
## Multiple R-squared:   0.97,  Adjusted R-squared:  0.9683
## F-statistic: 550.3 on 1 and 17 DF,  p-value: 2.177e-14
```

```r
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  0.07117    0.05797   1.228    0.236
# x.hl         0.16045    0.00684  23.458 2.18e-14 ***

dwt(model.3)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1        0.116145      1.725439   0.546
##  Alternative hypothesis: rho != 0
```

```r
# lag Autocorrelation D-W Statistic p-value
#   1        0.116145      1.725439   0.548
# Alternative hypothesis: rho != 0

coef(model.3)[1]/(1-acp) # 1.77933
```

```
## (Intercept)
##     1.77933
```

```r
sqrt(vcov(model.3)[1,1])/(1-acp) # 1.449373
```

```
## [1] 1.449373
```

```r
# First Differences Procedure

y.fd = y.ts-lag1y
x.fd = x.ts-lag1x
```

```
dwt(lm(y.fd ~ x.fd))
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.1160548      1.748834   0.624
##  Alternative hypothesis: rho != 0
```

```
# lag Autocorrelation D-W Statistic p-value
#   1       0.1160548      1.748834    0.62
# Alternative hypothesis: rho != 0

model.4 <- lm(y.fd ~ x.fd -1)
summary(model.4)
```

```
##
## Call:
## lm(formula = y.fd ~ x.fd - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.08958 -0.03231  0.02412  0.05344  0.15139
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## x.fd 0.168488   0.005096   33.06   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06939 on 18 degrees of freedom
## Multiple R-squared:  0.9838, Adjusted R-squared:  0.9829
## F-statistic:  1093 on 1 and 18 DF,  p-value: < 2.2e-16
```

```
#      Estimate Std. Error t value Pr(>|t|)
# x.fd 0.168488   0.005096   33.06   <2e-16 ***

mean(comsales)-coef(model.4)[1]*mean(indsales) # -0.3040052
```

```
##       x.fd
## -0.3040052
```

```
detach(blaisdell)
```

Metal fabricator and vendor employees (regression with autoregressive errors)

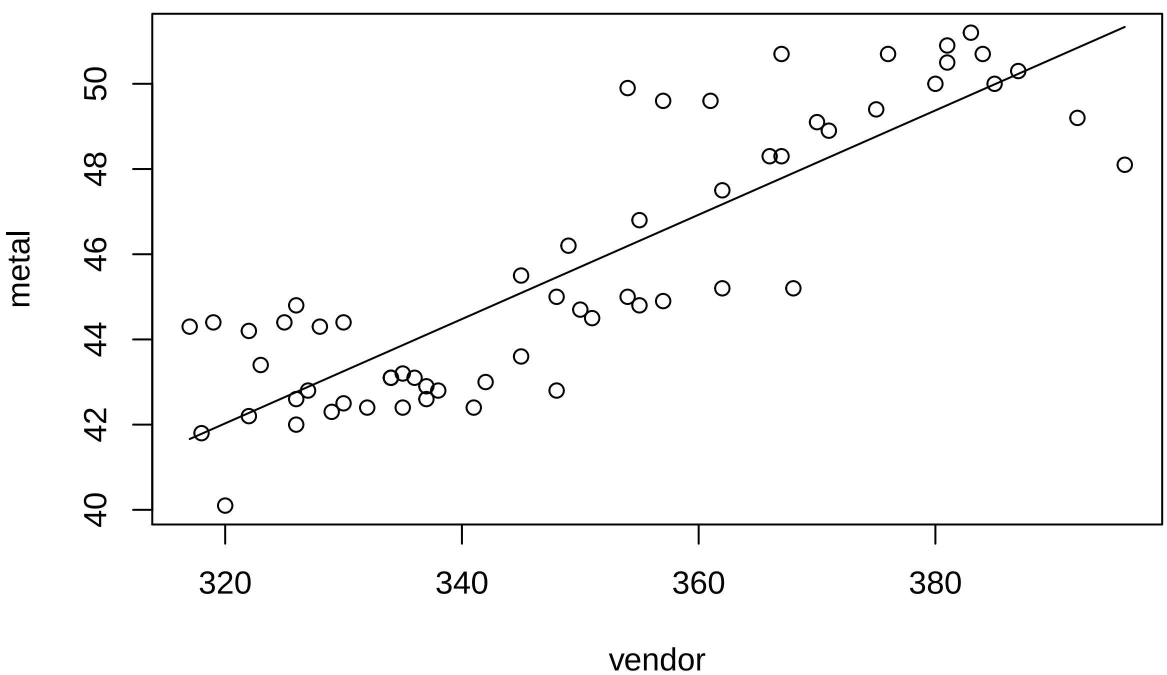
Load the employee data. Fit a simple linear regression model of metal vs vendor. Create a scatterplot of the data with a regression line. Create a scatterplot of the residuals vs time order. Create a plot of partial autocorrelations of the residuals. Use the dwt function in the car package to conduct the Durbin-Watson test on the residuals. Perform the Cochrane-Orcutt procedure to transform the variables.

```
employee <- read.table("./Data/employee.txt", header=T)
attach(employee)

model.1 <- lm(metal ~ vendor)
summary(model.1)
```

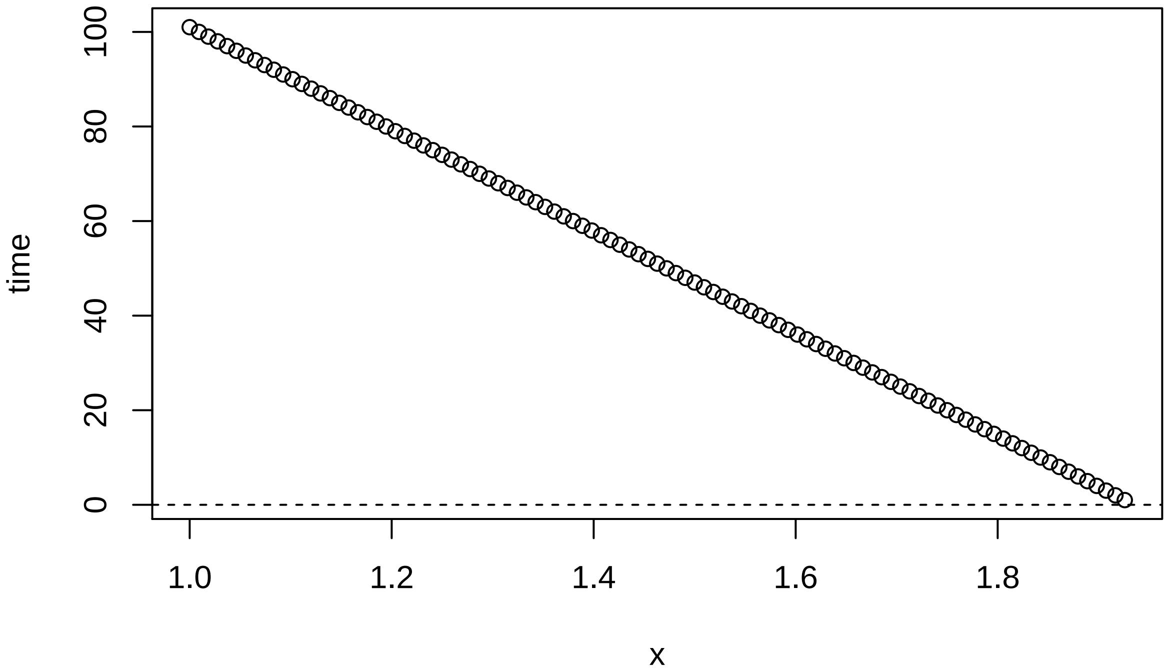
```
##
## Call:
```

```
## lm(formula = metal ~ vendor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2348 -1.2393 -0.0311  1.0022  3.7077
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.847911   3.299962   0.863    0.392
## vendor      0.122442   0.009423  12.994   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.59 on 58 degrees of freedom
## Multiple R-squared:  0.7443, Adjusted R-squared:  0.7399
## F-statistic: 168.8 on 1 and 58 DF,  p-value: < 2.2e-16
```

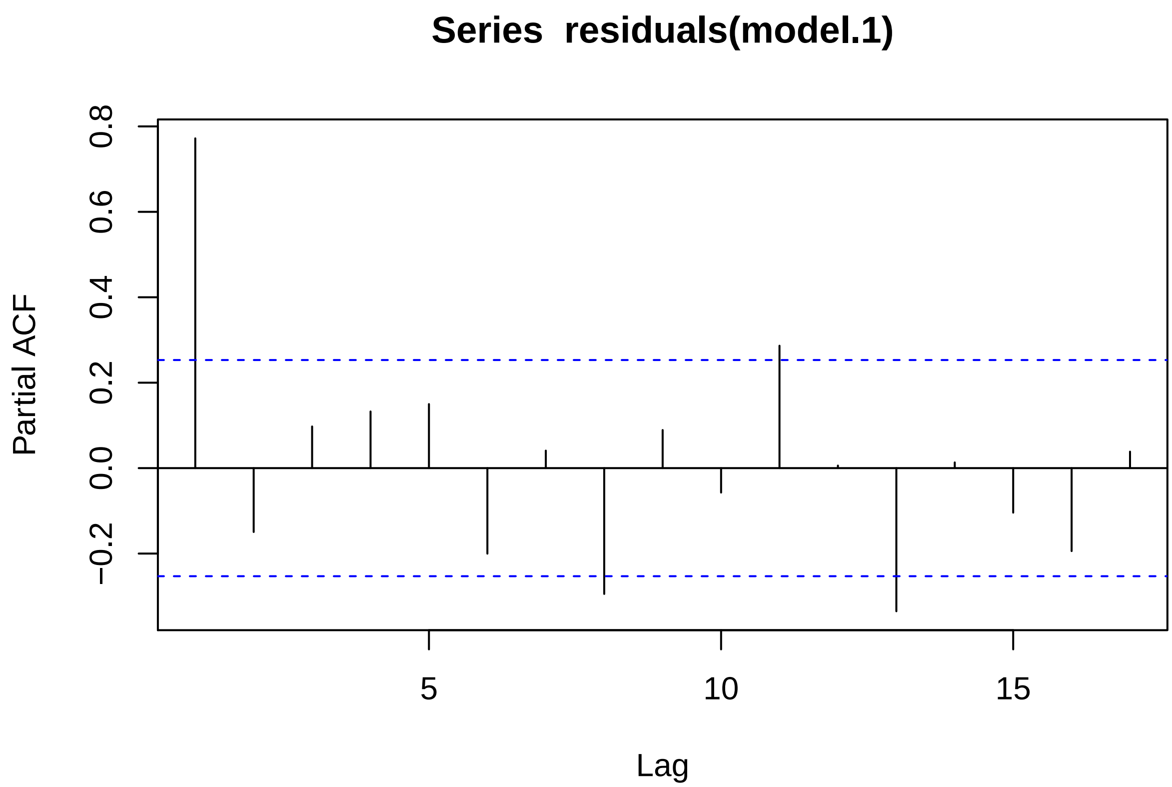
```
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept) 2.847911   3.299962   0.863    0.392
# vendor      0.122442   0.009423  12.994   <2e-16 ***

plot(x=vendor, y=metal,
     panel.last = lines(sort(vendor), fitted(model.1)[order(vendor)]))
```




```
plot(x=time, y=residuals(model.1), type="b",
     panel.last = abline(h=0, lty=2))
```




```
pacf(residuals(model.1))
```




```
# Durbin-Watson Test
dwt(model.1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1        0.772038     0.3592396       0
##  Alternative hypothesis: rho != 0
```

```r
# lag Autocorrelation D-W Statistic p-value
#   1        0.772038     0.3592396       0
# Alternative hypothesis: rho != 0

# Cochrane-Orcutt Procedure
res.ts <- ts(residuals(model.1))
lag1res <- lag(res.ts, -1)
lagdata1 <- ts.intersect(res.ts, lag1res)
acp <- coef(lm(res.ts ~ lag1res -1, data=lagdata1)) # 0.831385
y.ts <- ts(metal)
x.ts <- ts(vendor)
lag1y <- lag(y.ts, -1)
lag1x <- lag(x.ts, -1)
y.co <- y.ts-acp*lag1y
x.co <- x.ts-acp*lag1x
model.2 <- lm(y.co ~ x.co)
summary(model.2)
```

```
##
## Call:
## lm(formula = y.co ~ x.co)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1944 -0.4425  0.1461  0.5125  1.2218
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.87560    0.78655   6.199 6.78e-08 ***
## x.co         0.04795    0.01300   3.688 0.000505 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7342 on 57 degrees of freedom
## Multiple R-squared:  0.1927, Adjusted R-squared:  0.1785
## F-statistic:  13.6 on 1 and 57 DF,  p-value: 0.0005054
```

```r
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)  4.87560    0.78655   6.199 6.78e-08 ***
# x.co         0.04795    0.01300   3.688 0.000505 ***

coef(model.2)[1]/(1-acp) # 28.91557
```

```
## (Intercept)
##    28.91557
```

```r
sqrt(vcov(model.2)[1,1])/(1-acp) # se = 4.664789
```

```
##  lag1res
## 4.664789
```

```r
detach(employee)
```