Lab 6 - Case Studies 2

Benjamin M. Taylor, Kevin Hayes

January 20, 2023

1 Average monthly temperature at Nottingham Castle

The time series object `nottem` contains average air temperatures at Nottingham Castle in degrees Fahrenheit for 20 years.

[ ]:
```
options(repr.plot.width=15, repr.plot.height=15) # makes plots bigger in the␣
↪webpage
```

[ ]:
```
##  Code 1
##  plot the data...

plot(nottem, main = "nottem data",ylab =  "Average monthly temperature at␣
↪Nottingham Castle (deg. F)")
```

[ ]:
```
##  Code 2
##  a few plots to visualise the seasonal components...

Temperature = matrix(c(nottem), nrow = 12, byrow = FALSE) # note "byrow =␣
↪FALSE" is the default, so strictly, it is not necessary to specify this␣
↪argument

matplot(Temperature, type = "l")
```

[ ]:
```

```

[ ]:
```
Temperature = c(nottem)
Month = c(cycle(nottem))

boxplot(Temperature ~ Month)
```

[ ]:
```
cpgram(nottem)
```

[ ]:
```
plot(stl(nottem, s.window = 4))
```

[ ]:
```
##  Code 3
##  Differencing

nottem.D1 = diff(nottem, lag = 12)
```

```
cpgram(nottem.D1)
```

[ ]:
```
par(mfrow=c(1,2))

acf(nottem.D1, lag.max=40)

pacf(nottem.D1, lag.max=40)
```

[ ]:
```
##  Code 4
##  Fit S-ARIMA(p,d,q)(P,D,Q)

fit <- arima(nottem, order = c(1,0,0), list(order = c(2,1,0), period = 12))

tsdiag(fit)
```

[ ]:
```
cpgram(residuals(fit))
```

[ ]:
```
##  Code 5
##  Forecasting...

nott.fore <- predict(fit, n.ahead = 36)

ts.plot(nottem, nott.fore$pred, nott.fore$pred+2*nott.fore$se,
        nott.fore$pred-2*nott.fore$se, gpars = list(col = c(1,1,4,4)))
```

2 Airline Passenger Data

Monthly totals of international airline passengers, 1949 to 1960.

[ ]:
```
########################################
# code 01: look at the data

plot(AirPassengers, ylab="Air Passengers (1000's)")

segments(x0=1951, y0=100, x1=1961, y1=360, col = 2, lty = 2, lwd = 2 )

segments(x0=1950, y0=170, x1=1960.5, y1=620, col = 2, lty = 2, lwd = 2 )
```

[ ]:
```
########################################

# code 02: apply a transformation

APs = log(AirPassengers)

plot(APs)
```

[ ]:
```
#######################################
# code 03: autocorrelation dominated by trend:

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))

plot(APs, ylab ="", main=expression(X[i]))

acf(APs, main = "")

pacf(APs, main = "")
```

[ ]:
```
#######################################
# code 04: difference the data

AP1dif = diff(APs)

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))

plot(AP1dif, ylab = "", main=expression((1-B)*X[i]))

acf(AP1dif, lag.max = 48, main = "")

pacf(AP1dif, lag.max = 48, main = "")
```

[ ]:
```
#######################################
# code 05: difference the data, at lag 12

AP12dif = diff(APs,lag=12)

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))

plot(AP12dif, ylab="", main=expression((1-B^{12})*X[i]))

acf(AP12dif, lag.max = 48, main = "")

pacf(AP12dif, lag.max = 48, main = "")
```

[ ]:
```
#######################################
# code 06: difference the data at lag 12, and difference again at lag 1

Ys = diff(AP12dif)

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))

plot(Ys, ylab="",main=expression(Y[i]==(1-B)*(1-B^{12})*X[i]))

acf(Ys, lag.max = 48, main = "")
```

```
pacf(Ys, lag.max = 48, main = "")
```

[ ]:
```
#######################################
# code 07: select an ARIMA model

APfit =  arima(APs, order=c(0, 1, 0), seasonal = list(order=c(0, 1 ,0),␣
 ↪period=12))

tsdiag(APfit)
```

[ ]:
```
APfit =  arima(APs, order=c(0, 1, 1), seasonal = list(order=c(0, 1 ,1),␣
 ↪period=12))

tsdiag(APfit)
```

[ ]:
```
APfit
```

[ ]:
```
#######################################
# code 08: produce forecasts...

AP.pred = predict(APfit, n.ahead = 30)

AP.pred
```

[ ]:
```
#######################################
# code 09: plot forecasts...

plot(APs, xlim = c(1949, 1963), ylim = c(4.6,6.8),

     main="log[ Air Passengers (1000's) ]")

lines(AP.pred$pred,col="red")

lines(AP.pred$pred+2*AP.pred$se,col="red",lty=3)

lines(AP.pred$pred-2*AP.pred$se,col="red",lty=3)
```

[ ]:
```
# Homework: transform the prediction back to the
# original scale and display forecasts with prediction
# bands where Y axis is in 1000's airline passengers
```

[ ]:
```
#######################################
# code 10: Periodograms

cpgram(AirPassengers)
```

[ ]:
```
cpgram(APs)
```

[ ]:
```
cpgram(AP1dif)
```

[ ]:
```
cpgram(AP12dif)
```

[ ]:
```
cpgram(Ys)
```

[ ]:
```
cpgram(residuals(APfit))
```

3 Irish House Completions

Stuart's Data on Irish house completions used in class.

[ ]:
```
Grafton = read.table("data/Grafton78.txt", header = TRUE)

head(Grafton)
```

[ ]:
```
houses = ts(Grafton$Completions, start = c(1978, 1), frequency = 4)
```

[ ]:
```
## Code 01: Plot (all) the data...

plot(houses, main = "Housing completions, quarterly, 1978-2000",
     ylab = "Completions", xlab = "Year", type="n")

lines(lowess(houses, f=.10), col = "blue", lty = 1, lwd = 2)

points(houses, type="l")

points(x=1993, y=houses[4*15+1], pch=16)

text(1993,houses[4*15+1],"Q1 1993",pos=1)

points(x=1989.5, y=houses[4*11+3], pch=16)

text(1989.5,houses[4*11+3],"Q3 1989",pos=3)

points(x=1981, y=houses[13], pch=16)

text(1981,houses[13],"Q1 1981",pos=3)
```

[ ]:
```
## Code 02: Focus on Celtic Tiger...

plot(window(houses, start = 1993), ylab = "Completions", xlab = "Year",
     main = "Housing completions, quarterly, 1993-2000")
```

```
for(i in 1:4) {
    y = window(houses, start = c(1993, i), freq=1) ; x = time(y)
    points(x,y, pch=16, cex=2,col = switch(i, "blue","magenta","red","green"))
}
```

[ ]:
```
## Code 03: Celtic Tiger period dominated by trend...

houses = window(houses, start = 1993)

layout(mat=matrix(c(1,1,2,3),byrow=TRUE,ncol=2))

plot(houses, type= "o", main= "Housing completions, quarterly, 1993-2000")

acf(houses)

pacf(houses)
```

[ ]:
```
cpgram(houses)
```

[ ]:
```
## Code 04: Lag-1 differencing

houses.d1 = diff(houses)

layout(mat=matrix(c(1,1,2,3),byrow=TRUE,ncol=2))

plot(houses.d1, type= "o", main= "Detrended Housing completions, 1993-2000")

acf(houses.d1)

pacf(houses.d1)
```

[ ]:
```
cpgram(houses.d1)
```

[ ]:
```
# differenced series is dominated by seasonal variation

## Code 05: Lag-1 differencing, followed by Lag-4 differencing

houses.d1.D1 = diff(houses.d1, lag = 4)

layout(mat=matrix(c(1,1,2,3),byrow=TRUE,ncol=2))

plot(houses.d1.D1, type= "o", main= "Detrended & Deseasonalised Housing Data")

acf(houses.d1.D1, lag.max=55)

pacf(houses.d1.D1, lag.max=55)
```

[ ]:
```
cpgram(houses.d1.D1)
```

[ ]:
```
## Code 06: Model selection

fit1 = arima(houses, order = c(0,1,0), seasonal = list(order = c(0,1,1),␣
 ↪period=4))

cpgram(residuals(fit1), main= "SARIMA(0,1,0)x(0,1,1)")
```

[ ]:
```
fit1
```

[ ]:
```
fit2 = arima(houses, order = c(4,1,0), seasonal = list(order = c(0,1,1),␣
 ↪period=4))

cpgram(residuals(fit2), main= "SARIMA(4,1,0)x(0,1,1)")
```

[ ]:
```
fit2
```

[ ]:
```
tsdiag(fit2)
```

[ ]:
```
houses.pred = predict(fit2, n.ahead=8)

plot(houses, xlim=c(1992,2005), ylim=c(3800,15500),
     main = "Forecasting from Grafton Group Data")

lines(houses.pred$pred, col="blue", lwd=2)

lines(houses.pred$pred + 2*houses.pred$se, col="red", lwd=1)

lines(houses.pred$pred - 2*houses.pred$se, col="red", lwd=1)
```