Anexo VII

Código de TSeries

```
# Librerias auxiliares
library(ggplot2)
library(zoo)
library(forecast)

# Instalamos y cargamos TSeries
install.packages("tseries")
library(tseries)

# Cargamos los datos
datos <- read.csv("./monthly-traffic-fatalities-in-on.csv")
accidentes <- ts(datos, start = c(1960, 1), frequency = 12)
accidentes.zoo <- zoo(datos,
                      order.by = seq.Date(from = as.Date("1960/01/01"), by = "month",
                                                  length.out = nrow(datos)), frequency = 12)
colnames(accidentes.zoo) <- "accidentes"

# Función genérica para plotting de test vs pred a través de ggplot2
combine <- function(test, pred) {
  require(ggplot2)
  p <- ggplot() +
    geom_line(aes(x = index(as.zoo(test)),
                  y = coredata(as.zoo(test)), colour = "Test")) +
    geom_line(aes(x = index(as.zoo(acc.test)),
                  y = pred, colour = "Prediccion")) +
    scale_color_manual(name = "", values = c("Test" = "black",
                                             "Prediccion" = "red"),
                       labels = c("Test","Predicción")) +
    ylab("Número de accidentes de tráfico") +
    xlab("Año 1974") + scale_x_continuous(breaks = c(), labels = c())
  p
}

# Ajustamos estacionalmente la serie y volvemos a dividir en dos conjuntos
decomposition <- decompose(accidentes, type = "additive")
accidentes.adj <- seasadj(decomposition) # forecast
acc.train.adj <- window(accidentes.adj, start = c(1960,1), end = c(1973,12))
acc.test.adj <- window(accidentes.adj, start = c(1974,1))

# Plotting con ggplot del training y el test desestacionalizados
adj <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.adj)),
                y = coredata(as.zoo(acc.train.adj)),
                colour = "Training")) +
  geom_line(aes(x = index(as.zoo(acc.test.adj)),
                y = coredata(as.zoo(acc.test.adj)),
                colour = "Test")) +
  scale_color_manual(name = "", values = c("Training" = "blue",
                                           "Test" = "red"),
                     labels = c("Test","Training")) +
  xlab("Año") + ylab("Número de accidentes de \n tráfico")
adj

# ¿Es estacionaria?
adf.test(acc.train.adj, alternative = "stationary") # Es estacionaria
adf.test(acc.train.adj, alternative = "explosive") # No es explosiva
```

```
pp.test(acc.train.adj, alternative = "stationary") # Es estacionaria
pp.test(acc.train.adj, alternative = "explosive") # No es explosiva
kpss.test(acc.train.adj, null = "Trend") # 0.055

# Diferenciamos para hacerla estacionaria en tendencia (evidencias visuales)
accidentes.dif.adj <- diff(accidentes.adj)
acc.train.dif.adj <- window(accidentes.dif.adj,
                            start = c(1960,2), end = c(1973,12))
acc.test.dif.adj <- window(accidentes.dif.adj, start = c(1974,1))

# Repetimos los tests
adf.test(acc.train.dif.adj, alternative = "stationary") # Es estacionaria
adf.test(acc.train.dif.adj, alternative = "explosive") # No es explosiva
pp.test(acc.train.adj, alternative = "stationary") # Es estacionaria
pp.test(acc.train.adj, alternative = "explosive") # No es explosiva
kpss.test(acc.train.dif.adj, null = "Trend") # Es estacionaria en tendencia

# Plotting con ggplot del nuevo training y el test
dif.adj <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Training")) +
  geom_line(aes(x = index(as.zoo(acc.test.dif.adj)),
                y = coredata(as.zoo(acc.test.dif.adj)),
                colour = "Test")) +
  scale_color_manual(name = "", values = c("Training" = "blue",
                                           "Test" = "red"),
                     labels = c("Test","Training")) +
  xlab("Año") + ylab("Primeras diferencias")
dif.adj

# Linealidad en media
terasvirta.test(acc.train.dif.adj) # Es lineal en media
white.test(acc.train.adj) # Es lineal en media

# Calculamos la mayor diferencia en la serie diferenciada
maxdrawdown(acc.train.dif.adj)
acc.train.dif.adj.drop <- window(acc.train.dif.adj,
                                 start = c(1967, 9), end = c(1969, 7))
p <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj))),
            col = "blue") +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj.drop)),
                y = coredata(as.zoo(acc.train.dif.adj.drop))),
            col = "red") +
  xlab("Año") + ylab("Primeras diferencias")
p

# Bootstrap
boots.block <- tsbootstrap(acc.train.dif.adj, type = "block") # block
block <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Original")) +
  geom_line(aes(x = index(as.zoo(boots.block)),
                y = coredata(as.zoo(boots.block)),
                colour = "Bootstrap")) +
  scale_color_manual(name = "", values = c("Original" = "black",
                                           "Bootstrap" = "red"),
```

```
                     labels = c("Bootstrap","Original")) +
  xlab("Año") + ylab("Primeras diferencias")
block

boots.stationary <- tsbootstrap(acc.train.dif.adj,
                                type = "stationary") # stationary
stationary <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Original")) +
  geom_line(aes(x = index(as.zoo(boots.stationary)),
                y = coredata(as.zoo(boots.stationary)),
                colour = "Bootstrap")) +
  scale_color_manual(name = "", values = c("Original" = "black",
                                           "Bootstrap" = "red"),
                     labels = c("Bootstrap","Original")) +
  xlab("Año") + ylab("Primeras diferencias")
stationary

initial.acf <- function(serie) {
  return(acf(serie, plot = FALSE)$acf[2:8])
}

tsbootstrap(acc.train.dif.adj, nb= 500, type = "stationary",
            statistic = initial.acf)

# Surrogate
surr <- surrogate(acc.train.dif.adj, ns = 1, fft = TRUE, amplitude = TRUE)
surr.plot <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Original")) +
  geom_line(aes(x = index(as.zoo(surr)),
                y = coredata(as.zoo(surr)),
                colour = "Surrogated")) +
  scale_color_manual(name = "", values = c("Original" = "black",
                                           "Surrogated" = "red"),
                     labels = c("Original","Surrogate")) +
  xlab("Año") + ylab("Primeras diferencias")
surr.plot

# Estudiamos los correlogramas
autoplot(Acf(acc.train.dif.adj))
autoplot(Pacf(acc.train.dif.adj))

# Ajustamos un ARiMA(1,1,1)
model.1 <- arma(acc.train.dif.adj, order = c(1,1), include.intercept = FALSE)
summary(model.1)  # AIC =  1419.3
autoplot(Acf(model.1$residuals))
jarque.bera.test(na.remove(model.1$residuals))
bds.test(na.remove(model.1$residuals))
# Ajustamos un ARiMA(2,1,1)
model.2 <- arma(acc.train.dif.adj, order = c(2,1), include.intercept = FALSE)
summary(model.2) # AIC = 1414.81
autoplot(Acf(model.2$residuals))
jarque.bera.test(na.remove(model.2$residuals))
bds.test(na.remove(model.2$residuals))
# Graficamos ambos modelos
m.1 <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
```

```
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Original")) +
  geom_line(aes(x = index(as.zoo(fitted(model.1))),
                y = coredata(as.zoo(fitted(model.1))),
                colour = "Ajustada")) +
  scale_color_manual(name = "", values = c("Original" = "black",
                                           "Ajustada" = "blue"),
                     labels = c("Ajustada", "Original")) +
  xlab("Año") + ylab("Primeras diferencias")
m.1

m.2 <- ggplot() +
  geom_line(aes(x = index(as.zoo(acc.train.dif.adj)),
                y = coredata(as.zoo(acc.train.dif.adj)),
                colour = "Original")) +
  geom_line(aes(x = index(as.zoo(fitted(model.2))),
                y = coredata(as.zoo(fitted(model.2))),
                colour = "Ajustada")) +
  scale_color_manual(name = "", values = c("Original" = "black",
                                           "Ajustada" = "blue"),
                     labels = c("Ajustada", "Original")) +
  xlab("Año") + ylab("Primeras diferencias")
m.2
```