lab2_sridhar

Sridhar Adhikarla (sriad858)

September 28, 2019

```r
#Required libs
library(astsa)
library(kernlab)
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:kernlab':
##
##     alpha
```

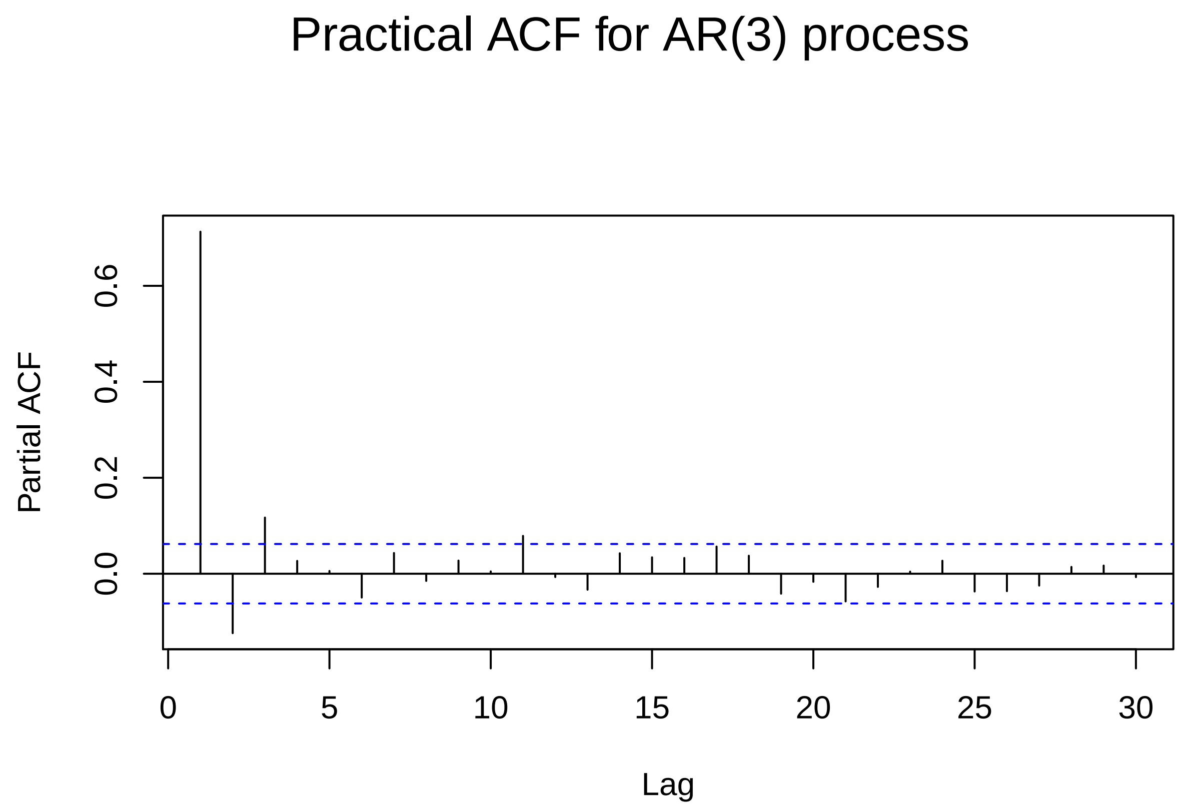
```r
set.seed(12345)
```

Assignment 1

a - Simualte values from AR(3)

The pcaf

```r
ar3 <- arima.sim(model = list(ar = c(0.8,-0.2,0.1)), n = 1000)

simulationCorrelation = function(ar3){
  df = ts.intersect("xt" = ar3, "xt1" = lag(ar3, k = -1), "xt2" = lag(ar3, k = -2))
  mod = lm(xt ~ xt1 + xt2, data = df)
  coeffs = c(mod$coefficients[2], mod$coefficients[3])
  xthPrime = lag(ar3, k = -3) - (coeffs[2]*lag(ar3, k = -1)+coeffs[1]*lag(ar3, k = -2))
  xtPrime = ar3 - (coeffs[1]*lag(ar3, k = -1)+coeffs[2]*lag(ar3, k = -2))

  return(cor(xthPrime, xtPrime))
}

par(mfrow=c(1,1),oma = c(0, 0, 2, 0))
pacfVal <- pacf(ar3, main=NA)$acf[3]
mtext("Practical ACF for AR(3) process", outer = TRUE, cex = 1.5)
```




```r
cat("Simulated", simulationCorrelation(ar3), "\n",
    "PACF value", pacfVal, "\n",
    "Theoretical", 0.1, "\n")
```

```
## Simulated 0.1248488
##  PACF value 0.1170643
##  Theoretical 0.1
```

b)

```r
ar2 <- arima.sim(model = list(ar = c(0.8,0.1)), n = 100)

ywl <- ar(ar2, order = 2, method = "yule-walker", aic = FALSE)
cls <- ar(ar2, order = 2, method = "ols", aic = FALSE)
mle <- ar(ar2, order = 2, method = "mle", aic = FALSE)

df <- data.frame("Yule-Walker" = c(ywl$ar[1],ywl$ar[2]),
                 "Conditional LS" = c(cls$ar[1],cls$ar[2]),
                 "MLE" = c(mle$ar[1], mle$ar[2]))
rownames(df) <- c("Component 1", "Component 2")
df
```

```
##             Yule.Walker Conditional.LS         MLE
```

```
## Component 1  0.85717519     0.93860754  0.90150780
## Component 2 -0.01999019    -0.09108315 -0.03544039
```

```
mle <- arima(ar2, order = c(2,0,0), method = "ML")
ci <- unname(c(mle$coef[2]+1.96*(sqrt(mle$var.coef[2,2])),
        mle$coef[2]-1.96*(sqrt(mle$var.coef[2,2]))))
cat("Confidence Interval - ", ci, "\nTheoritical Value - ", mle$coef[2], "\n")
```

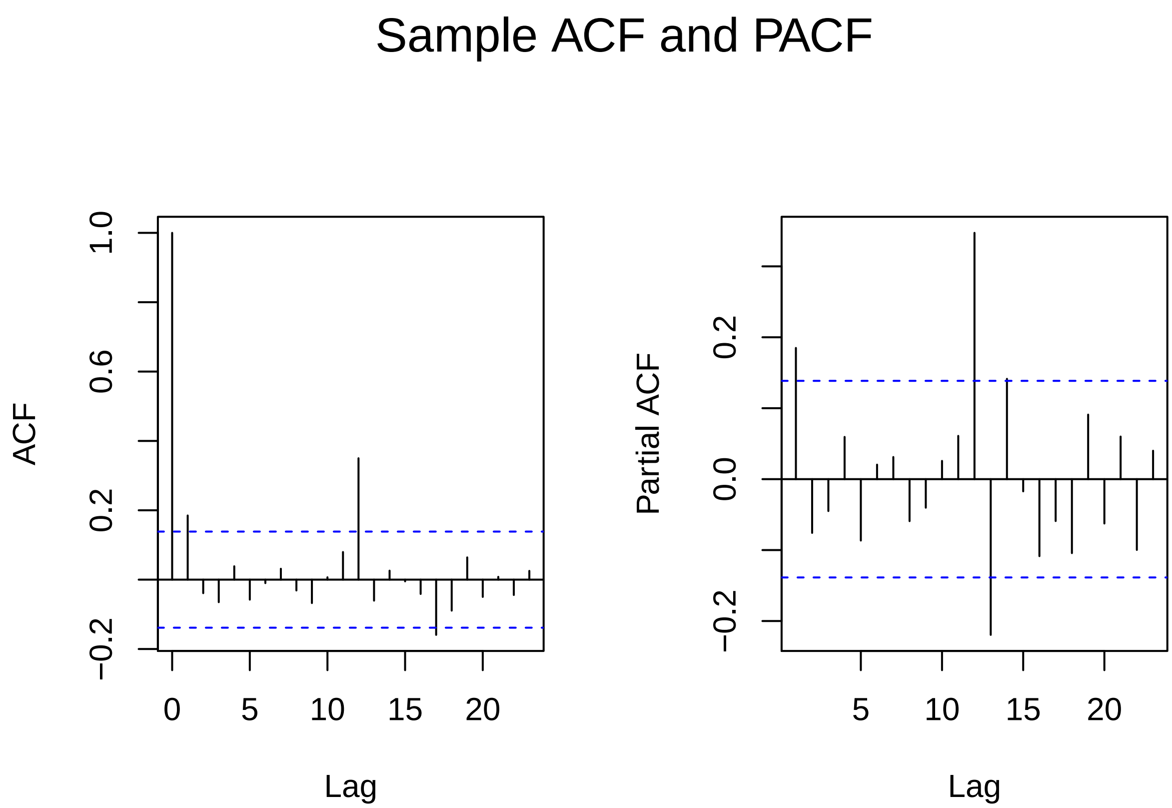
```
## Confidence Interval -  0.1068769 -0.2991362
## Theoritical Value -  -0.09612965
```

```
print(ifelse(((0.1-ci[1])*(ci[2]-0.1)>0),
       "Theoretical value fall within confidence",
       "Theoretical value does not fall within confidence"))
```

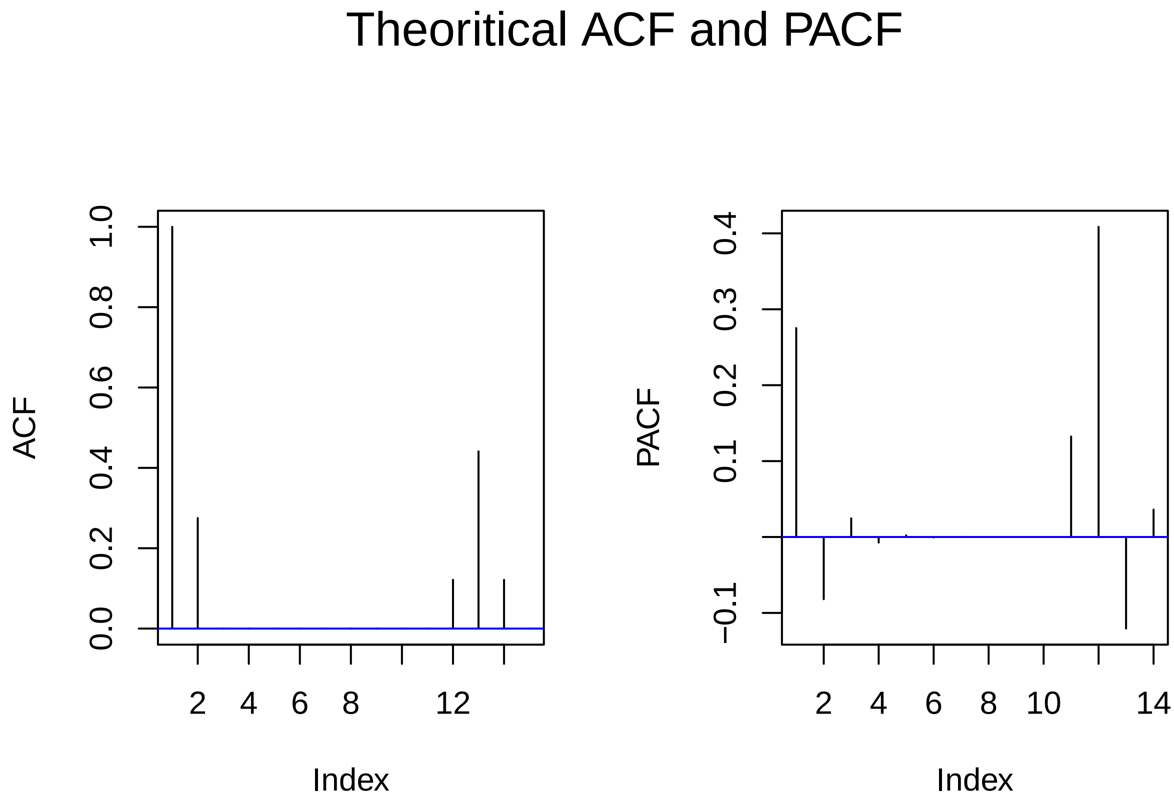
```
## [1] "Theoretical value fall within confidence"
```

c)

```
arimaSeason <- arima.sim(list(order = c(0,0,12),
                              ma = c(0.3,rep(0,10),0.6)), n = 200)
par(mfrow=c(1,2),oma = c(0, 0, 2, 0))
acf(arimaSeason, main=NA)
pacf(arimaSeason, main=NA)
mtext("Sample ACF and PACF", outer = TRUE, cex = 1.5)
```




```
#Theoretical
armaSeasonT <- ARMAacf(ma = c(0.3,rep(0,10),0.6,0.6*0.3))
armaSeasonTP <- ARMAacf(ma = c(0.3,rep(0,10),0.6,0.6*0.3), pacf = TRUE)

plot(armaSeasonT, type = "h", ylab = "ACF")
abline(h = 0, col = "blue")
plot(armaSeasonTP, type = "h", ylab = "PACF")
abline(h = 0, col = "blue")
mtext("Theoritical ACF and PACF", outer = TRUE, cex = 1.5)
```


Theoritical ACF and PACF
ACF
0.0
0.2
0.4
0.6
0.8
1.0
2
4
6
8
12
Index
PACF
−0.1
0.1
0.2
0.3
0.4
2
4
6
8
10
14
Index

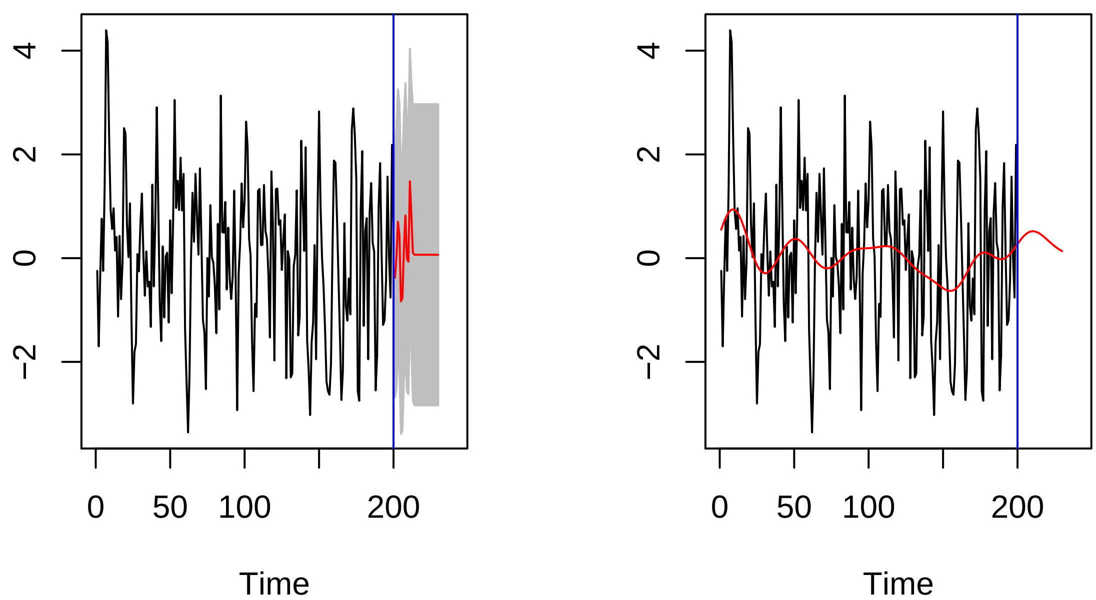
d)

```r
par(mfrow=c(1,2),oma = c(0, 0, 2, 0))
seasonArimaSim = arima.sim(list(order = c(0,0,12), ma = c(0.7,rep(0,10),0.6)), n = 200)
seasonDF = data.frame(y = as.vector(seasonArimaSim), x = 1:200)

#Fitting arima model
seasonArimaFit = arima(seasonArimaSim, order = c(0,0,1), seasonal = list(order = c(0,0,1),period = 12))
seasonArimaPred = predict(seasonArimaFit, n.ahead = 30)

plot(seasonArimaSim, xlim = c(0,240), ylab = NA)
upper <- seasonArimaPred$pred + 1.96*seasonArimaPred$se
lower <- seasonArimaPred$pred - 1.96*seasonArimaPred$se
polygon(c(time(upper),rev(time(upper))),c(lower, rev(upper)),border = 8, col = "grey")
lines(seasonArimaPred$pred, col = "red")
abline(v = 200, col = "blue", lty = 1)

#Using gausspr to fit the data
seasonGaussModel = gausspr(y ~ x, data = seasonDF)
```

```
## Using automatic sigma estimation (sigest) for RBF or laplace kernel
```

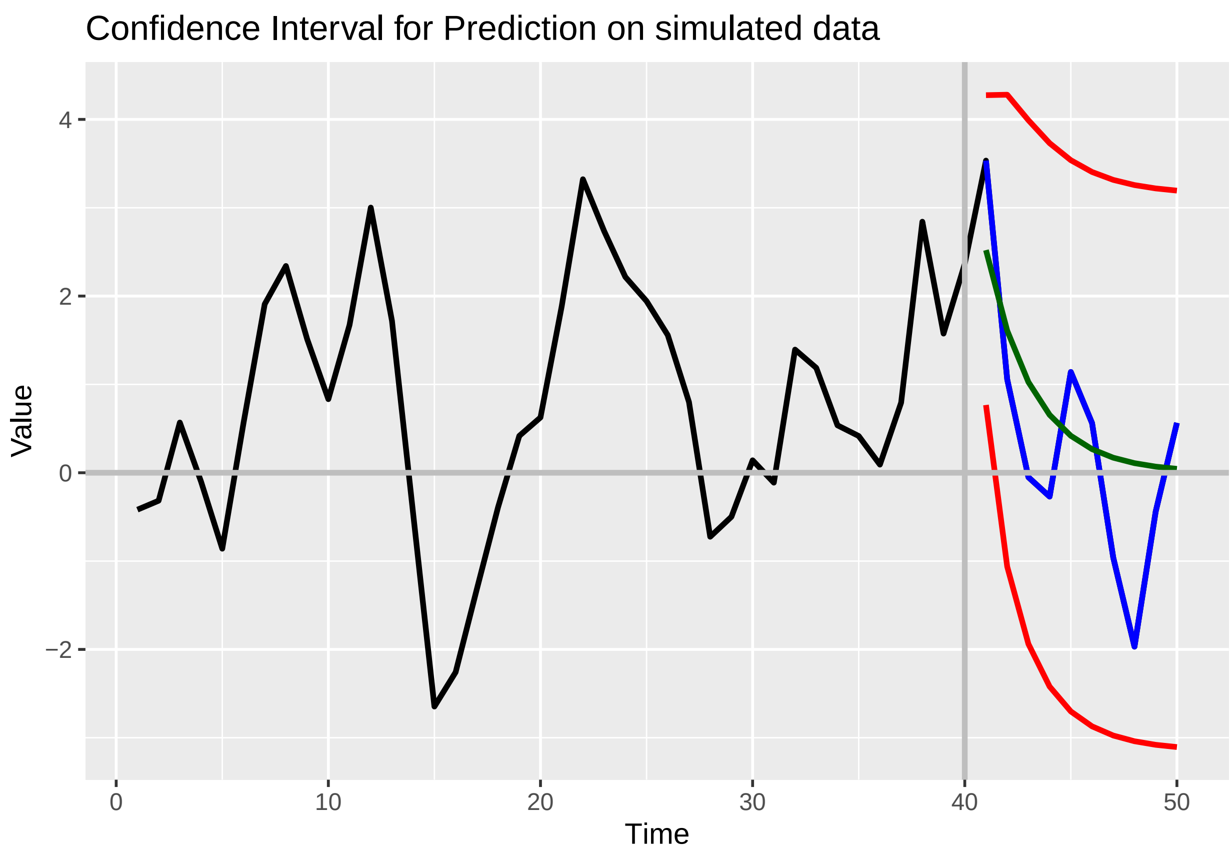
```r
seasonGaussPred = predict(seasonGaussModel, newdata = data.frame(x = 1:230))


plot(seasonArimaSim, col = "black", type = "l",ylab = NA, xlab = "Time", xlim=c(0,240))
lines(seasonGaussPred, col = "red")
mtext("Arima Fit vs Gausspr on Seasonal data", outer = TRUE, cex = 1.5)
abline(v = 200, col = "blue", lty = 1)
```

Arima Fit vs Gausspr on Seasonal data



e

```
arimaE <- arima.sim(list(order = c(1,0,1), ar = 0.7, ma = 0.5), n = 50)

arimasample <- arimaE[1:40]

modelArima <- arima(arimasample, order = c(1,0,1), include.mean = 0)

pred101 <- predict(modelArima, n.ahead = 10)

ggplot() +
  geom_line(aes(x=1:50, y=arimaE[1:50]), col="black", lwd=1) +
  geom_line(aes(x=41:50, y=arimaE[41:50]), col="blue", lwd=1) +
  geom_line(aes(x=41:50, y=pred101$pred[1:10]), col="darkgreen", lwd=1) +
  geom_line(aes(y = pred101$pred + 1.96*pred101$se, x = 41:50), col="red", lwd=1) +
  geom_line(aes(y = pred101$pred - 1.96*pred101$se, x = 41:50), col="red", lwd=1) +
  geom_vline(xintercept = 40, col="gray", lwd=1) +
  geom_hline(yintercept = 0, col="gray", lwd=1) +
  ggtitle("Confidence Interval for Prediction on simulated data") +
  xlab("Time") + ylab("Value")
```



Question 2

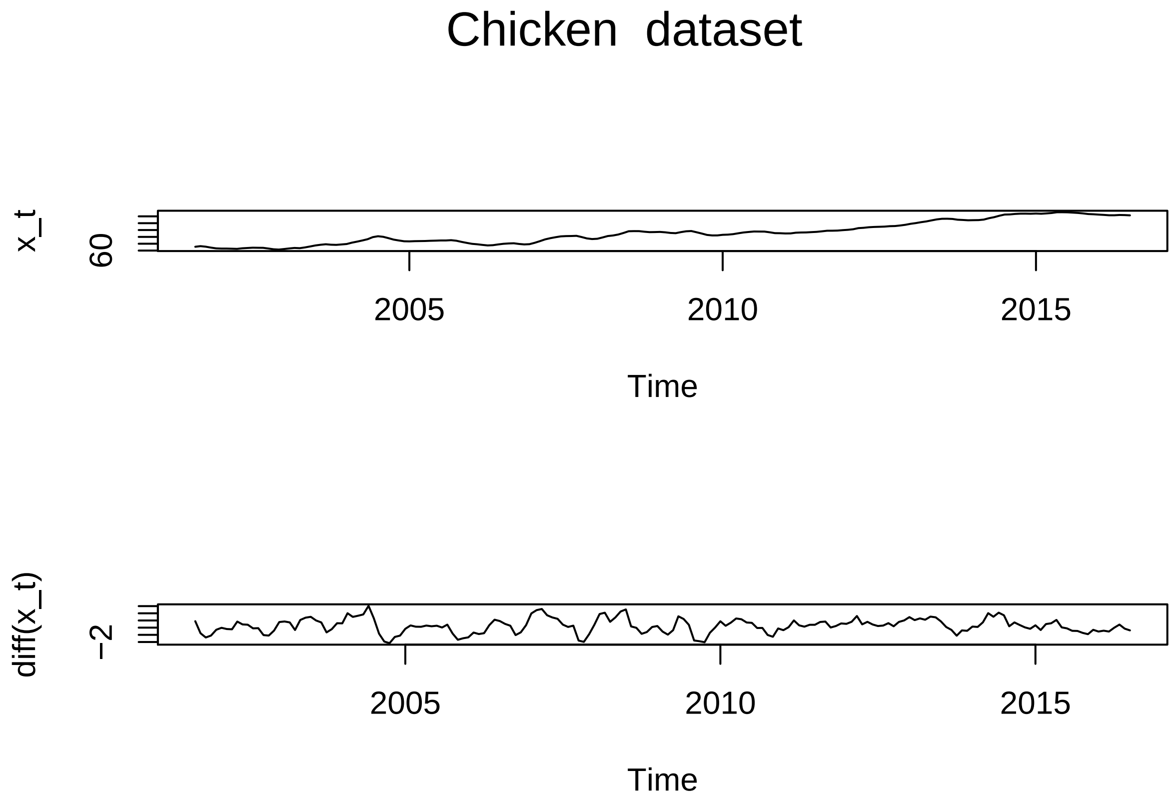
```
genPlots = function(x_t, dataset){
  par(mfrow=c(2,1),oma = c(0, 0, 2, 0))
  plot(x_t)
  plot(diff(x_t))
  mtext(paste(dataset, " dataset"), outer = TRUE, cex = 1.5)

  par(mfrow=c(2,2),oma = c(0, 0, 2, 0))
  acf(x_t, lag.max = 40, main="")
  pacf(x_t, lag.max = 40, main="")
  acf(diff(x_t), lag.max = 40, main="")
  pacf(diff(x_t), lag.max = 40, main="")
  mtext(paste(dataset, " dataset ACF and PACF plots"), outer = TRUE, cex = 1.5)
}
```

Datasets

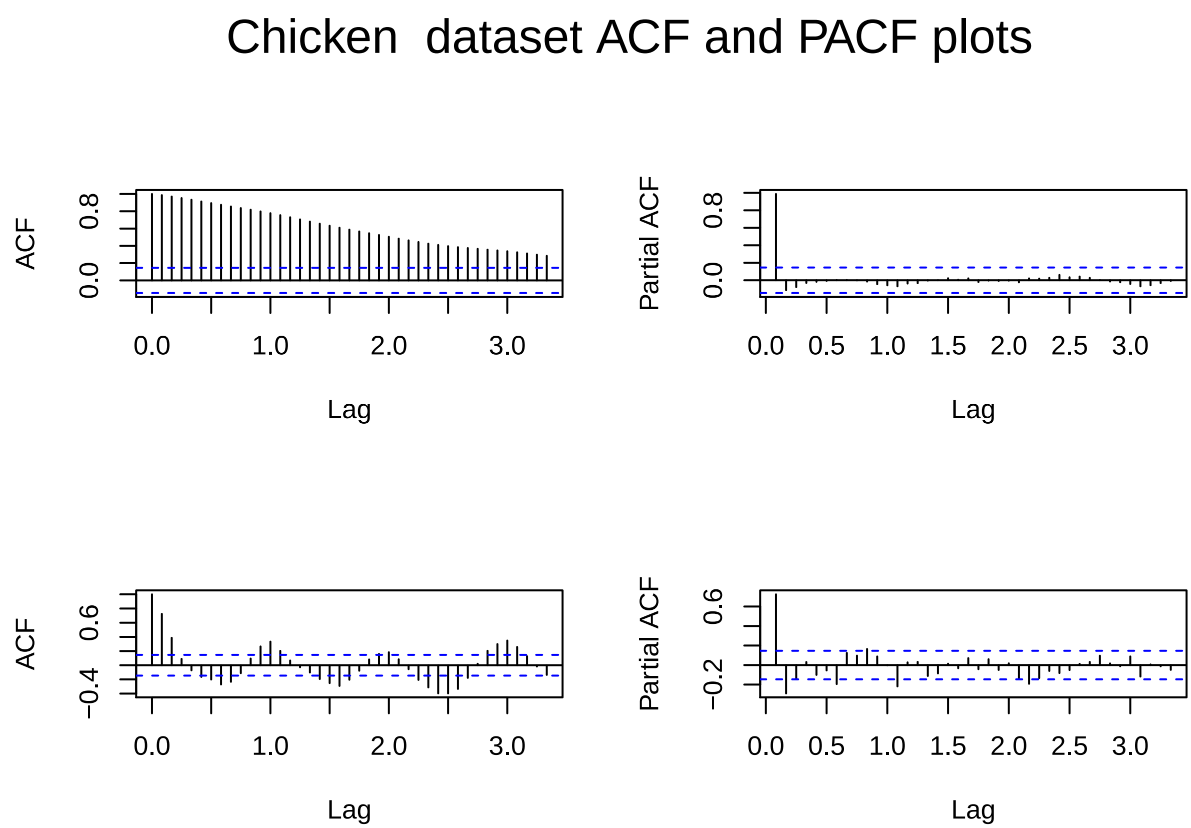
Chicken

The decreasing trend in the ACF plot of chicken and the cutoff in the PCAF plot suggests that this could be an AR process. The positive ACF at lag 1 for the differenced data confirms that this is a AR process. The PCAF of the differenced data suggests that it is a AR(3) process, since there is a cutoff after lag 3 in the plot. We can see seasonality in the differenced dataset. The arima model $ARIMA(3,1,0)(3,1,0)_{12}$. would be good for this dataset.

```
genPlots(chicken, "Chicken")
```



Chicken dataset
x_t
60
2005
2010
2015
Time
diff(x_t)
−2
2005
2010
2015
Time

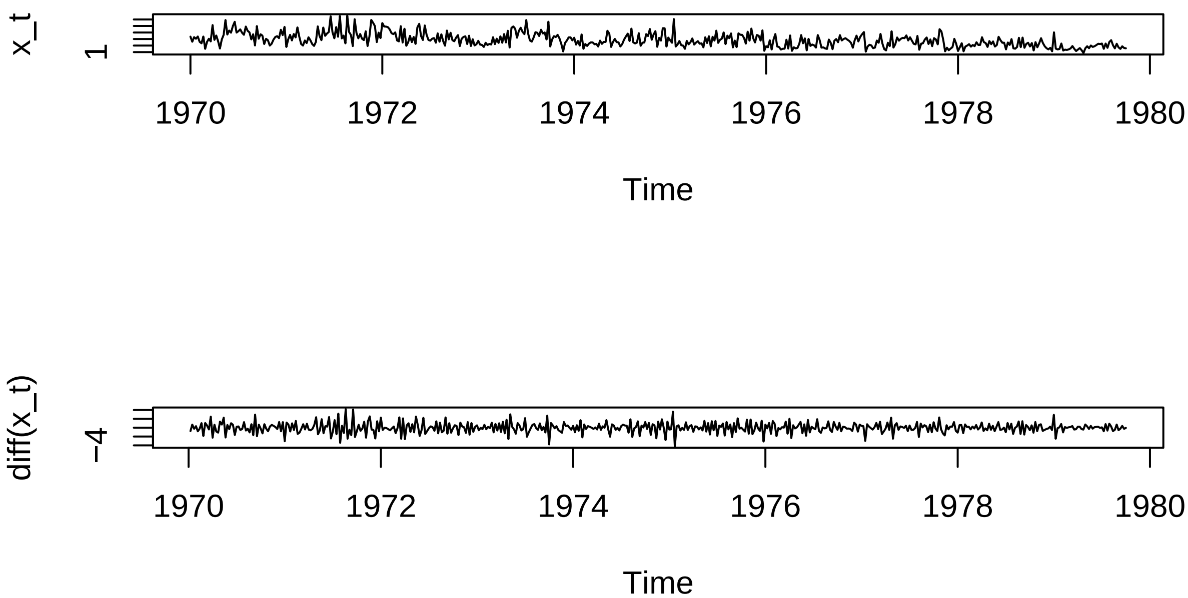


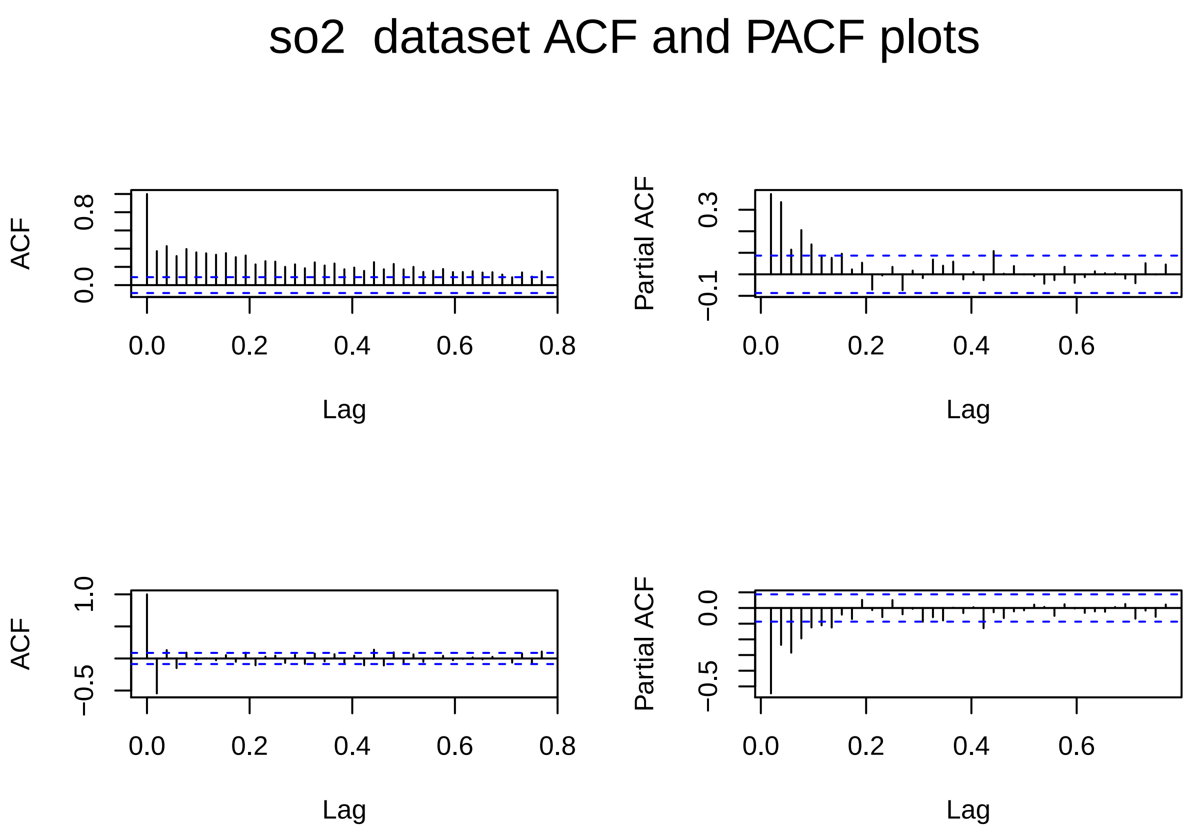
so2

The decreasing trend in the ACF plot of the dataset suggests an ARIMA model could be a good fit for it. The negative ACF at lag 1 for the differenced dataset suggests to use an MA model. The pcaf plot of differenced dataset tells us that MA(7) model would be a good fit for the dataset. $ARIMA(0, 1, 7)$ would be a good model for this data.

```
genPlots(so2, "so2")
```

so2 dataset


x_t
1
1970
1972
1974
1976
1978
1980
Time
diff(x_t)
−4
1970
1972
1974
1976
1978
1980
Time

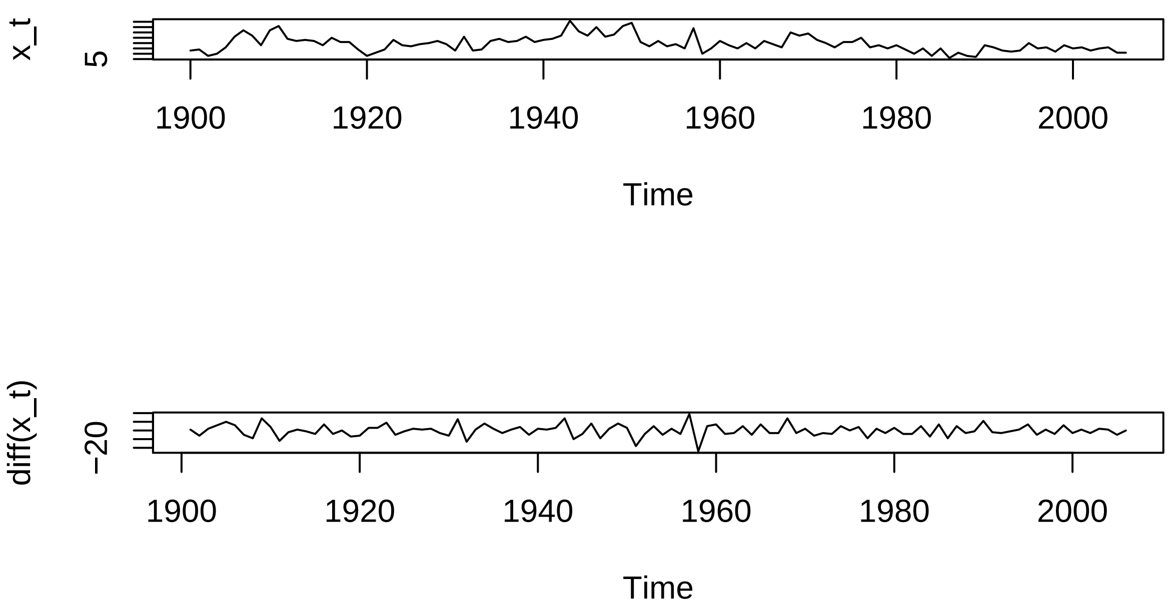


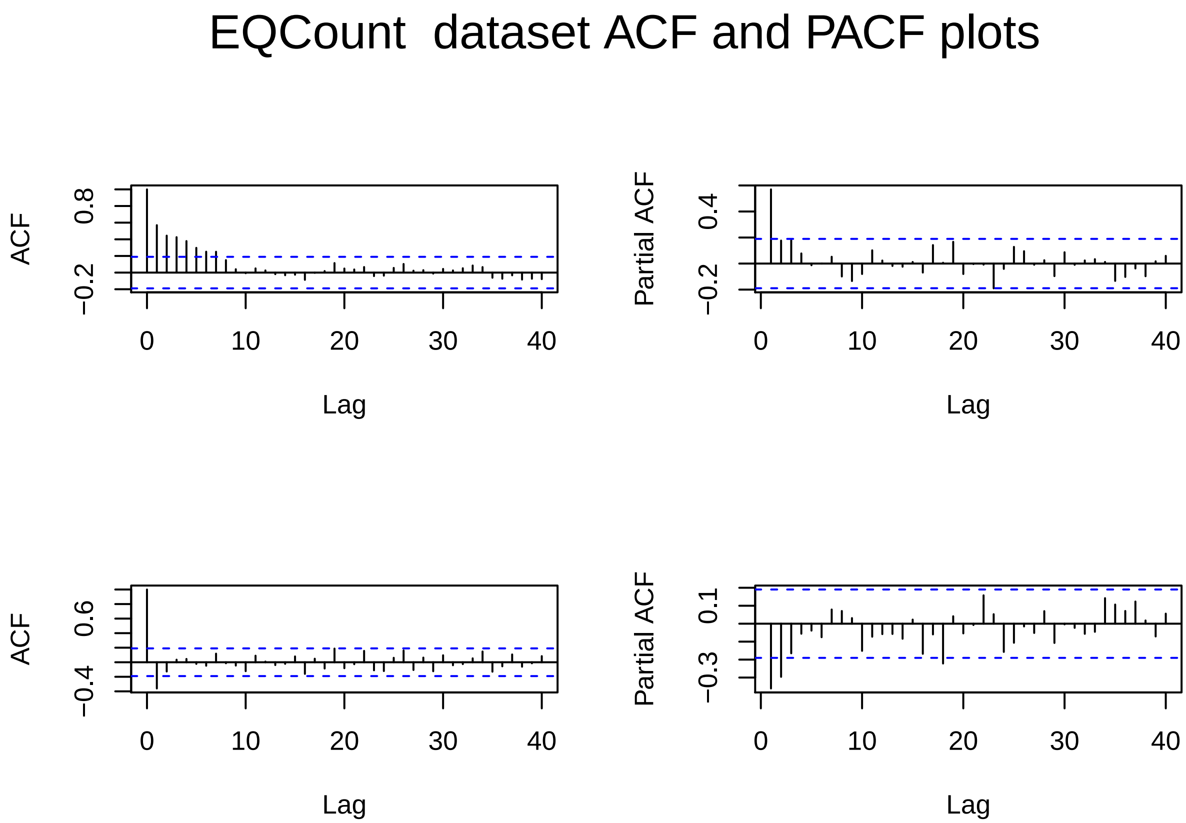
EQcount

The decreasing trend in the ACF plot of the dataset suggests an ARIMA model could be a good fit for it. The negative ACF at lag 1 for the differenced dataset suggests to use an MA model. The pcaf plot of differenced dataset tells us that MA(2) model would be a good fit for the dataset. $ARIMA(0,1,2)$ would be a good model for this data.

```
genPlots(EQcount, "EQCount")
```

EQCount dataset



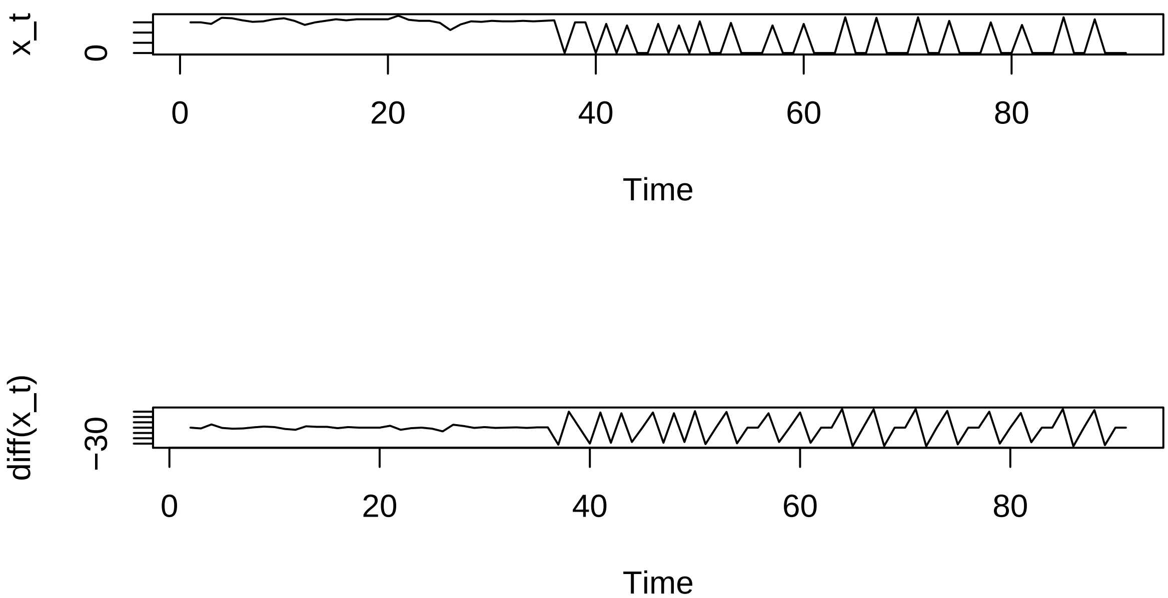


HCT

The decreasing trend in the ACF plot of the dataset suggests an ARIMA model could be a good fit for it. The negative ACF at lag 1 for the differenced dataset suggests to use an MA model. The pcaf plot of differenced dataset tells us that MA(7) model would be a good fit for the dataset. $ARIMA(0,1,7)$ would be a good model for this data. From the difference of order 1 we can see some seasonality after seven lags(seven days).

```
genPlots(HCT, "HCT")
```

HCT dataset



HCT dataset ACF and PACF plots