Time Series Introduction

Davis Townsend

December 20, 2016

```
 data(AirPassengers)
#tells you that this data is in time series format
 class(AirPassengers)
```

```
## [1] "ts"
```

```
 #find what year data starts in
 start(AirPassengers)
```

```
## [1] 1949    1
```

```
 #and when it ends
 end(AirPassengers)
```

```
## [1] 1960   12
```

```
 #find the cycle of the time series to be 12 months via:
 frequency(AirPassengers)
```

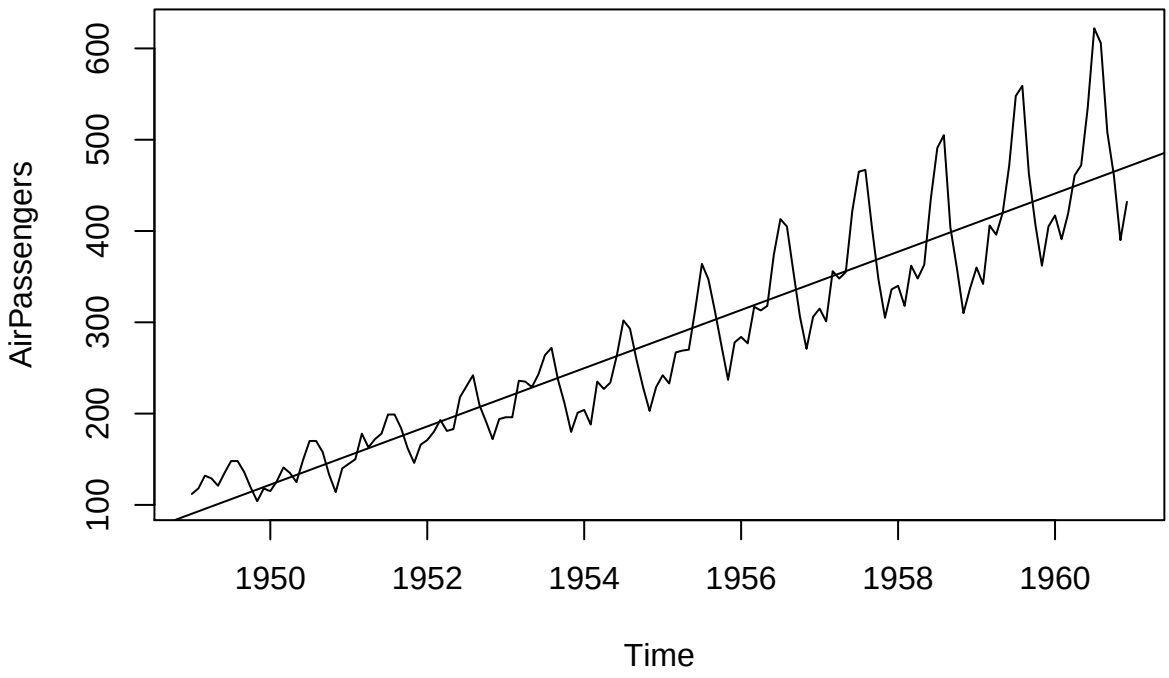
```
## [1] 12
```

```
 summary(AirPassengers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   104.0   180.0   265.5   280.3   360.5   622.0
```

plot the time series and fit a line to it

```
plot(AirPassengers)
abline(reg=lm(AirPassengers~time(AirPassengers)))
```

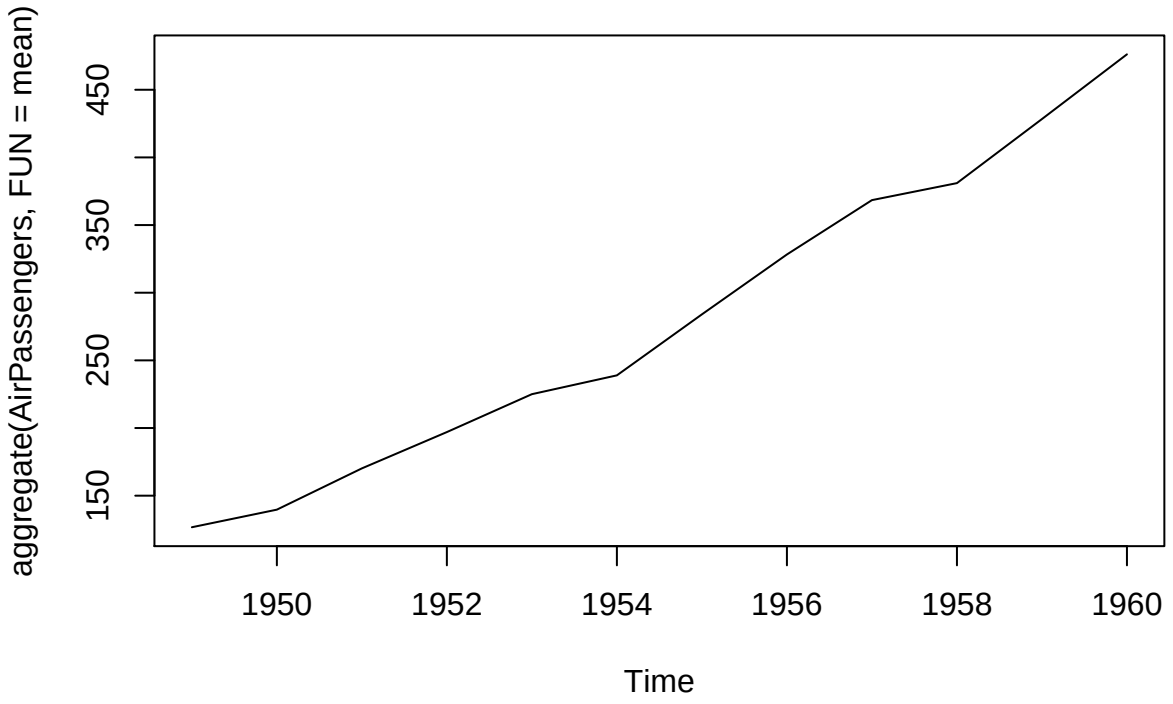


Here are other possible time series operations

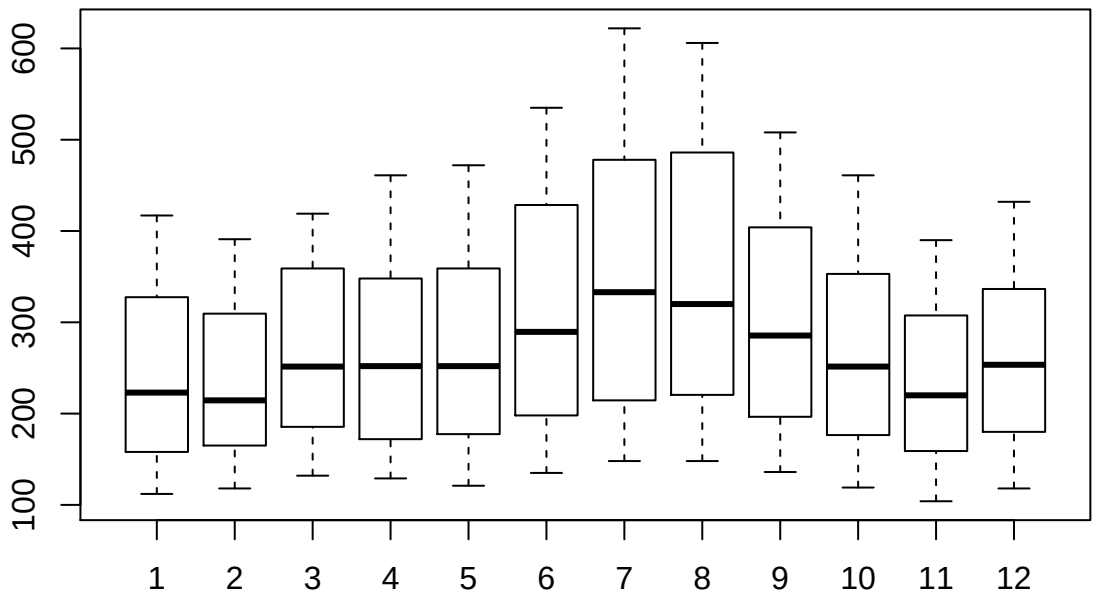
```
#prints the cycle across years
cycle(AirPassengers)
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949   1   2   3   4   5   6   7   8   9  10  11  12
## 1950   1   2   3   4   5   6   7   8   9  10  11  12
## 1951   1   2   3   4   5   6   7   8   9  10  11  12
## 1952   1   2   3   4   5   6   7   8   9  10  11  12
## 1953   1   2   3   4   5   6   7   8   9  10  11  12
## 1954   1   2   3   4   5   6   7   8   9  10  11  12
## 1955   1   2   3   4   5   6   7   8   9  10  11  12
## 1956   1   2   3   4   5   6   7   8   9  10  11  12
## 1957   1   2   3   4   5   6   7   8   9  10  11  12
## 1958   1   2   3   4   5   6   7   8   9  10  11  12
## 1959   1   2   3   4   5   6   7   8   9  10  11  12
## 1960   1   2   3   4   5   6   7   8   9  10  11  12
```

```
#aggregates cycles and displays year on year trend
plot(aggregate(AirPassengers,FUN=mean))
```




```
#boxplot over years gives sense of seasonal affect
boxplot(AirPassengers~cycle(AirPassengers))
```




```
#library for time series models
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 3.2.5
```

```
#take log of data to remove unequal variance, and we take difference to address the trend component (gr
adf.test(diff(log(AirPassengers)), alternative="stationary", k=0)
```

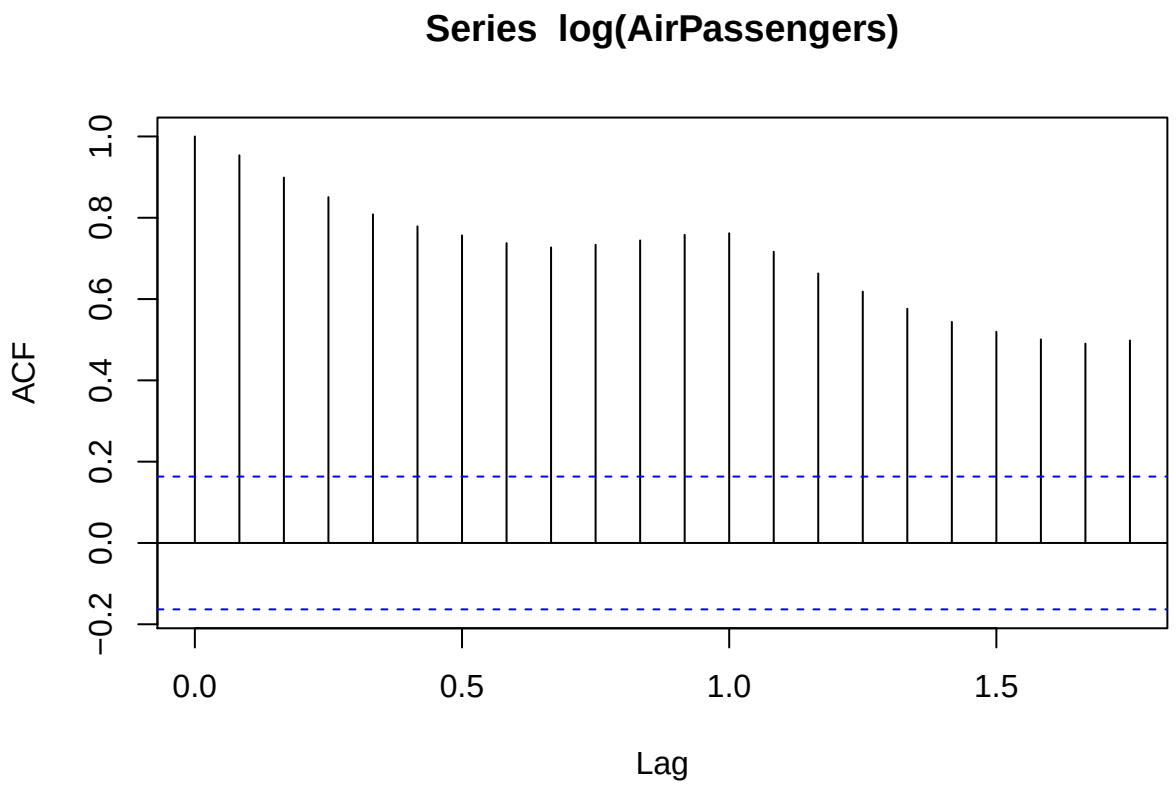
```
## Warning in adf.test(diff(log(AirPassengers)), alternative = "stationary", :
## p-value smaller than printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  diff(log(AirPassengers))
## Dickey-Fuller = -9.6003, Lag order = 0, p-value = 0.01
## alternative hypothesis: stationary
```

```
#look at p-value, see that series is stationary enough
```

Next we need to find the right parameters for the ARIMA model. We already know that d=1 since it took 1 difference to make the series stationary but we need to find p and q still.

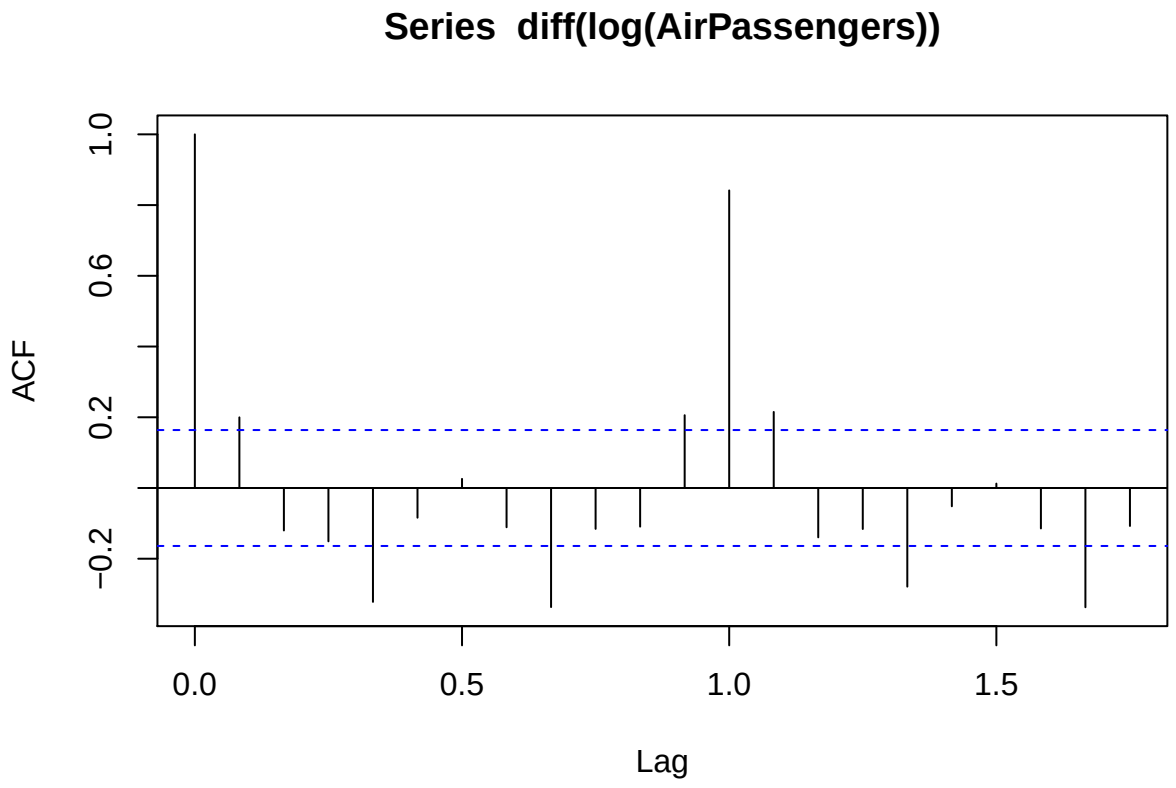
We can find this by looking at the correlation ACF and PACF plots

```r
#plot acf
acf(log(AirPassengers))
```

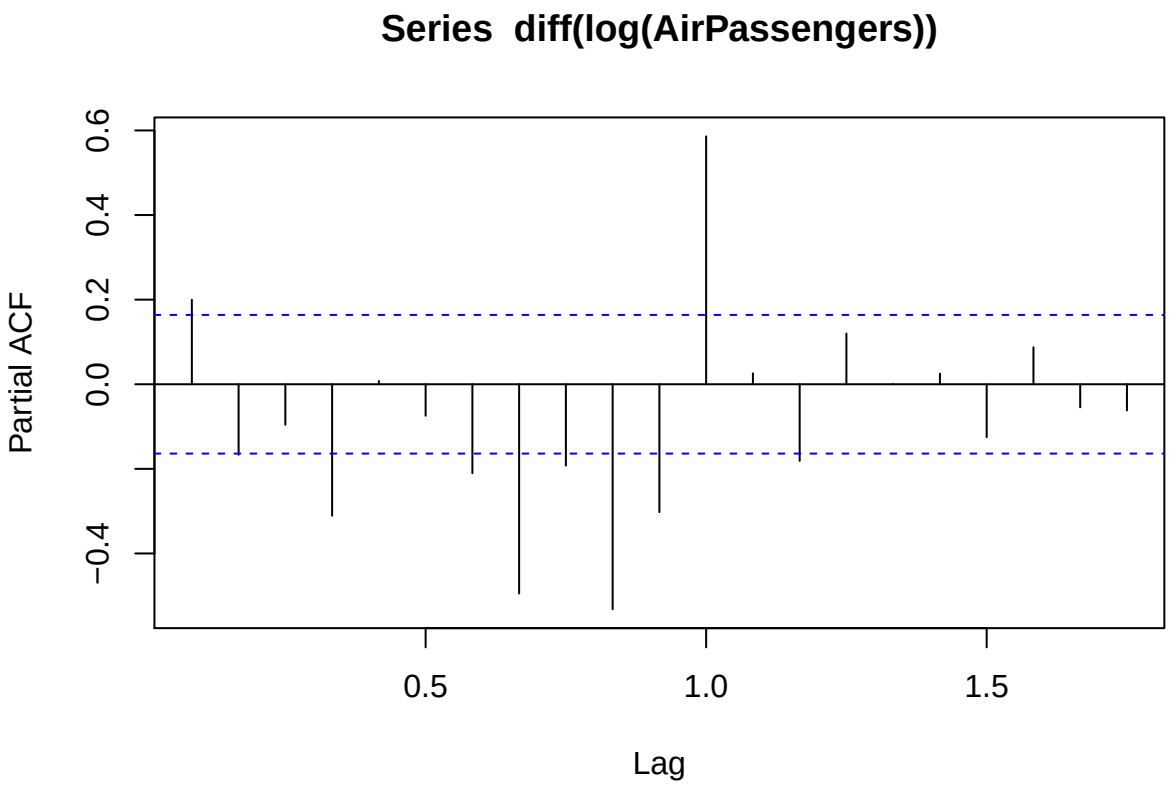


Clearly, the decay of ACF chart is very slow, which means that the population is not stationary. We have already discussed above that we now intend to regress on the difference of logs rather than log directly. Let's see how ACF and PACF curve come out after regressing on the difference.

```r
#acf plot
acf(diff(log(AirPassengers)))
```




```
#pacf plot
pacf(diff(log(AirPassengers)))
```



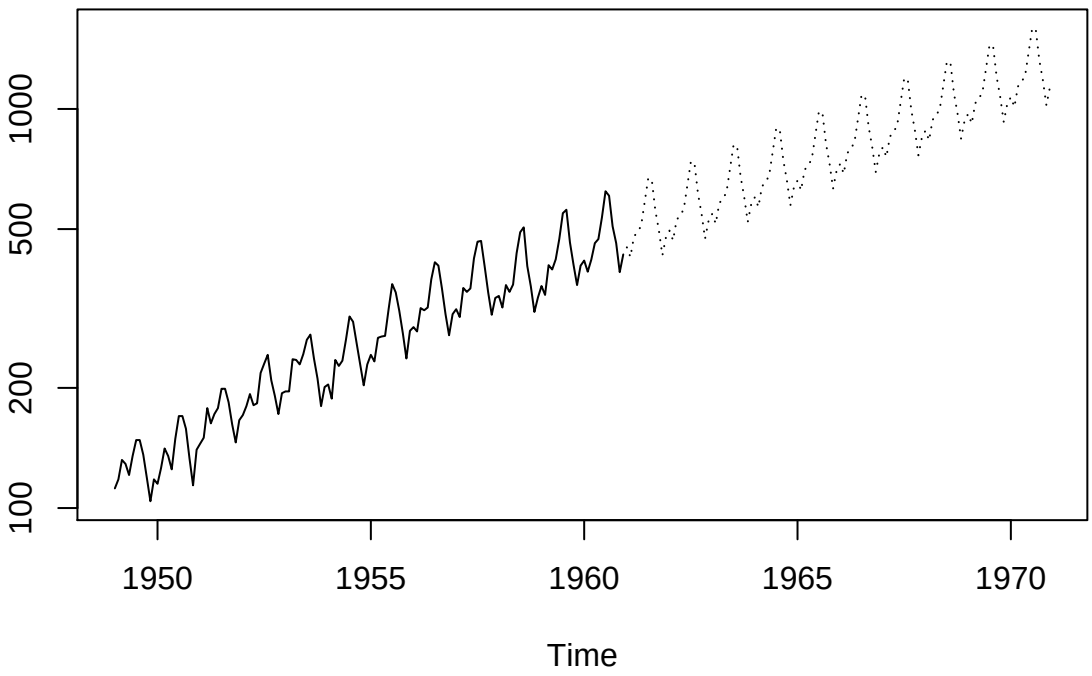
Clearly, ACF plot cuts off after the first lag. Hence, we understood that value of p should be 0 as the ACF is the curve getting a cut off. While value of q should be 1 or 2. After a few iterations, we found that (0,1,1) as (p,d,q) comes out to be the combination with least AIC and BIC.

Let's fit an ARIMA model and predict the future 10 years. Also, we will try fitting in a seasonal component in the ARIMA formulation. Then, we will visualize the prediction along with the training data. You can use the following code to do the same :

```
(fit <- arima(log(AirPassengers), c(0, 1, 1),seasonal = list(order = c(0, 1, 1), period = 12)))
```

```
##
## Call:
## arima(x = log(AirPassengers), order = c(0, 1, 1), seasonal = list(order = c(0,
##     1, 1), period = 12))
##
## Coefficients:
##           ma1     sma1
##       -0.4018  -0.5569
## s.e.   0.0896   0.0731
##
## sigma^2 estimated as 0.001348:  log likelihood = 244.7,  aic = -483.4
```

```
pred <- predict(fit, n.ahead = 10*12)
ts.plot(AirPassengers,2.718^pred$pred, log = "y", lty = c(1,3))
```


100
200
500
1000
1950
1955
1960
1965
1970
Time