Homework 6

Lei Yang, Ron Cordell, Subhashini Raghunathan

March 16, 2016

Exercise 1:

a. Discuss the mean and variance functions and how the similarities and differences from those we studied in classical linear model

The mean of a time series is defined as:

$$\mu_x(t) = \int_{-\infty}^{\infty} x_t f_t(x_t) dx_t$$

The mean is a function of time t, and the mean at any t is the expectation over all possible time series values at time t that could be generated by the stochastic process. f_t is the marginal probability distribution of x_t derived from the complete joint probability distribution. Note that this is a theoretical concept; in practice we have only one observation from the time series and we do not know the joint distribution, hence we cannot estimate the mean.

The variance function is defined as:

The variance is also a function of time t. This is also a theoretical concept; since we have only one time series it is impossible to estimate the variance.

In a classical linear model, the mean is the expected value of the iid variable X. In a sample we have multiple values of X and hence can calculate an estimate for the mean. The mean is a constant parameter. Similarly, we can estimate the variance from the sample values. The variance is also a constant parameter.

b. Define strict and weak statonarity

A time series x_t is strictly stationary if the joint distributions $F(x_{t1}, ., x_{tn})$ and $F(x_{t1+m}, ., x_{tn+m})$ are the same,for all $t_{1} .. t_{n} and m. This is a very strong condition; it implies that the distribution is unchanged for any time shift.

A time series x_t is weakly stationary if it is mean and variance stationary and its autocovariance Cov(xt,xt+k) depends on the time placement k alone.

Exercise 2:

a. Generate a zero-drift random walk model using 500 simulation
b. Provide the descriptive statistics of the simulated realizations. The descriptive statistics should include the mean, standard deviation, 25th, 50th, and 75th quantiles, minimum, and maximum

```
library(astsa)
```

```
## Warning: package 'astsa' was built under R version 3.2.3
```

```r
set.seed(898)
sigma_w = 1
#generate white noise simulation
x <- w <- rnorm(500,0,sigma_w)
#the randon walk model
for (t in 2:500) x[t] <- x[t-1] + w[t]

#summary statistics
mean(x)
```

```
## [1] -12.19499
```

```r
sd(x)
```

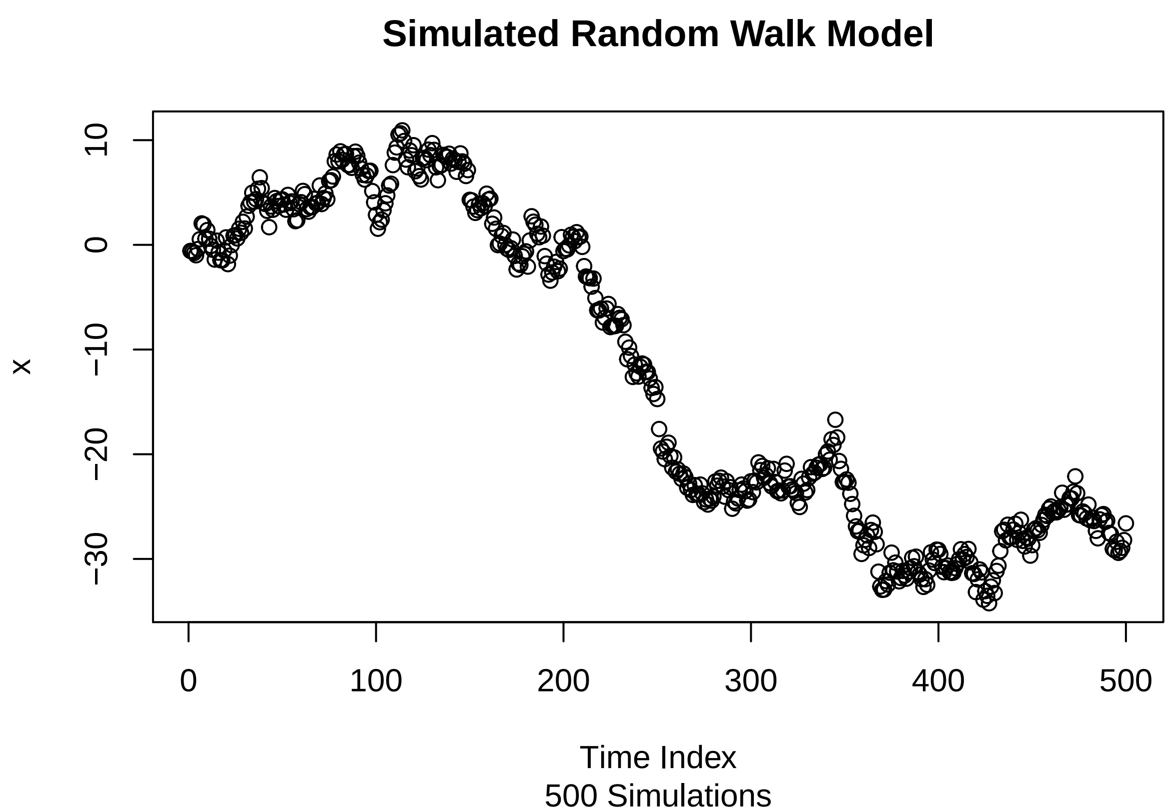
```
## [1] 14.77081
```

```r
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -34.230 -25.800 -15.720 -12.190   2.705  10.940
```

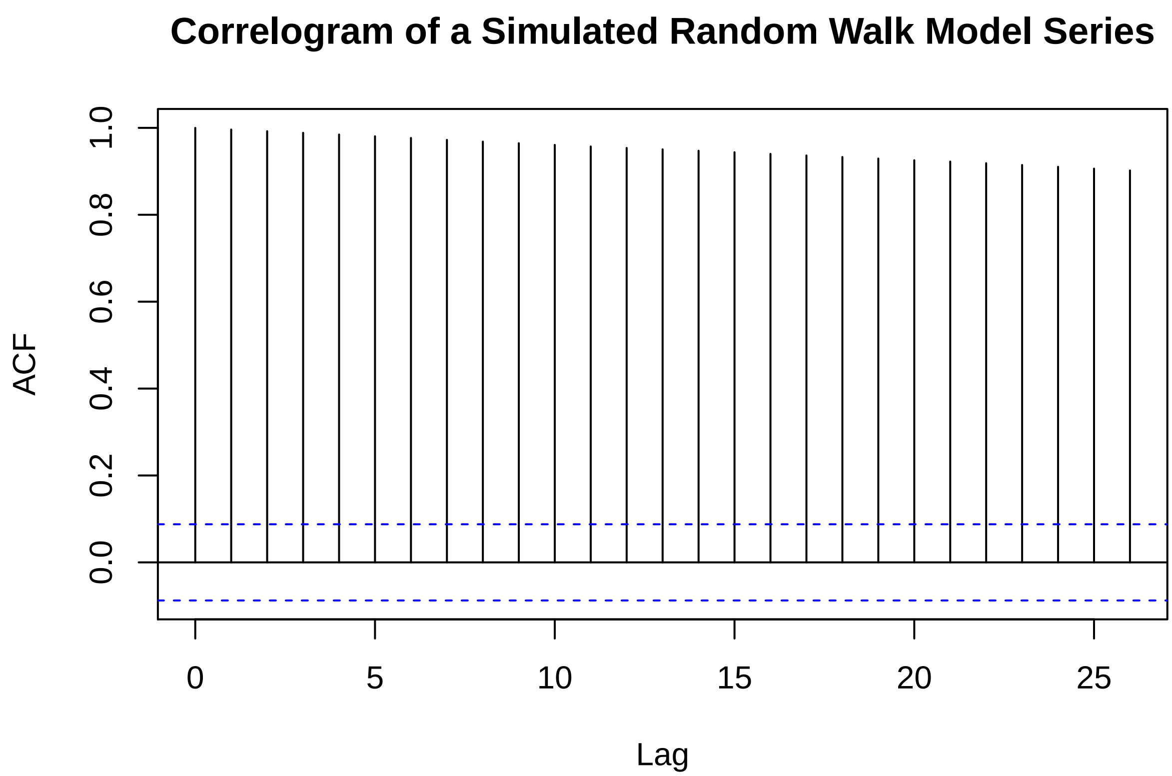
c. Plot the time-series plot of the simulated realizations

```r
plot(x, type="b", xlab="Time Index")
title("Simulated Random Walk Model", "500 Simulations")
```



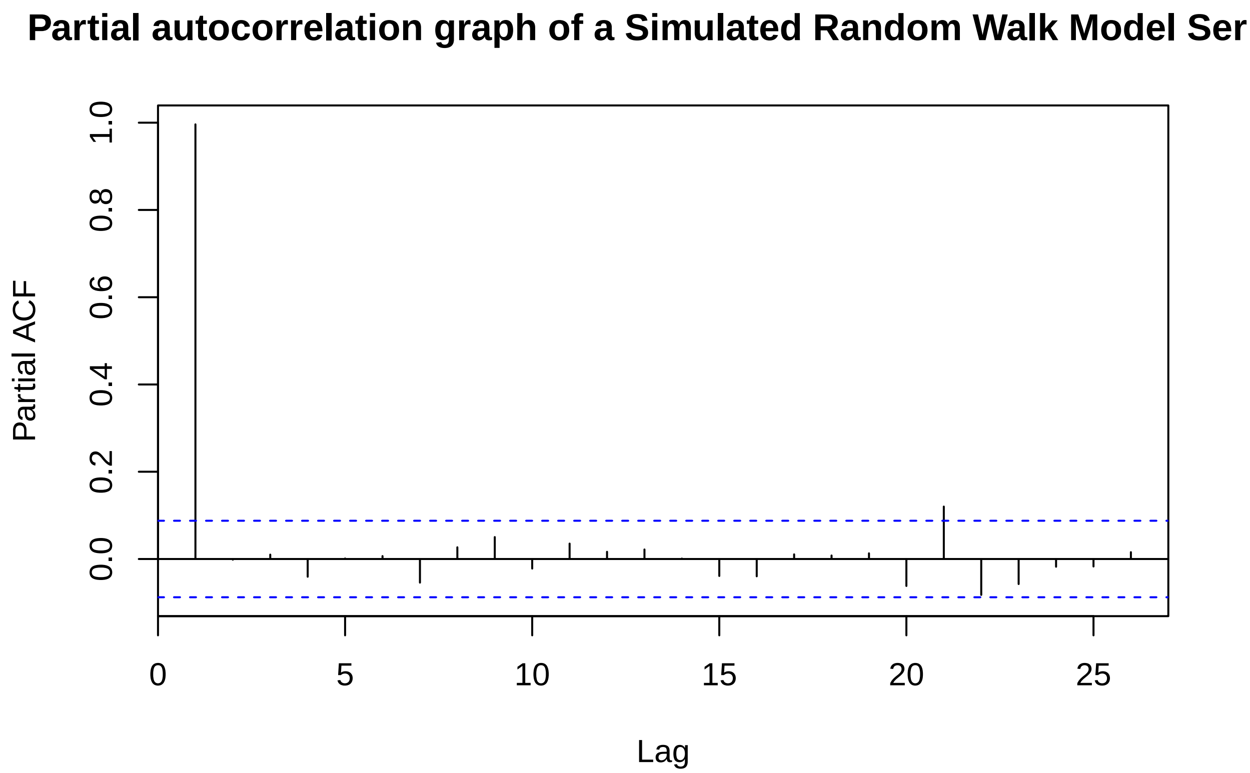
d. Plot the autocorrelation graph

```
acf(x, main="")
title("Correlogram of a Simulated Random Walk Model Series")
```



e. Plot the partial autocorrelation graph

```
pacf(x, main="")
title("Partial autocorrelation graph of a Simulated Random Walk Model Series")
```



Exercise 3:

a. Generate arandom walk with drift model using 500 simulation, with the drift = 0.5
b. Provide the descriptive statistics of the simulated realizations. The descriptive statistics should include the mean, standard deviation, 25th, 50th, and 75th quantiles, minimum, and maximum

```
set.seed(898)
sigma_w = 1
alpha0 = 0.5
#generate white noise simulation
x <- w <- rnorm(500,0,sigma_w)
#the randon walk model with drift
for (t in 2:500) x[t] <- alpha0 + x[t-1] + w[t]

#summary statistics
mean(x)
```

```
## [1] 112.555
```

```
sd(x)
```

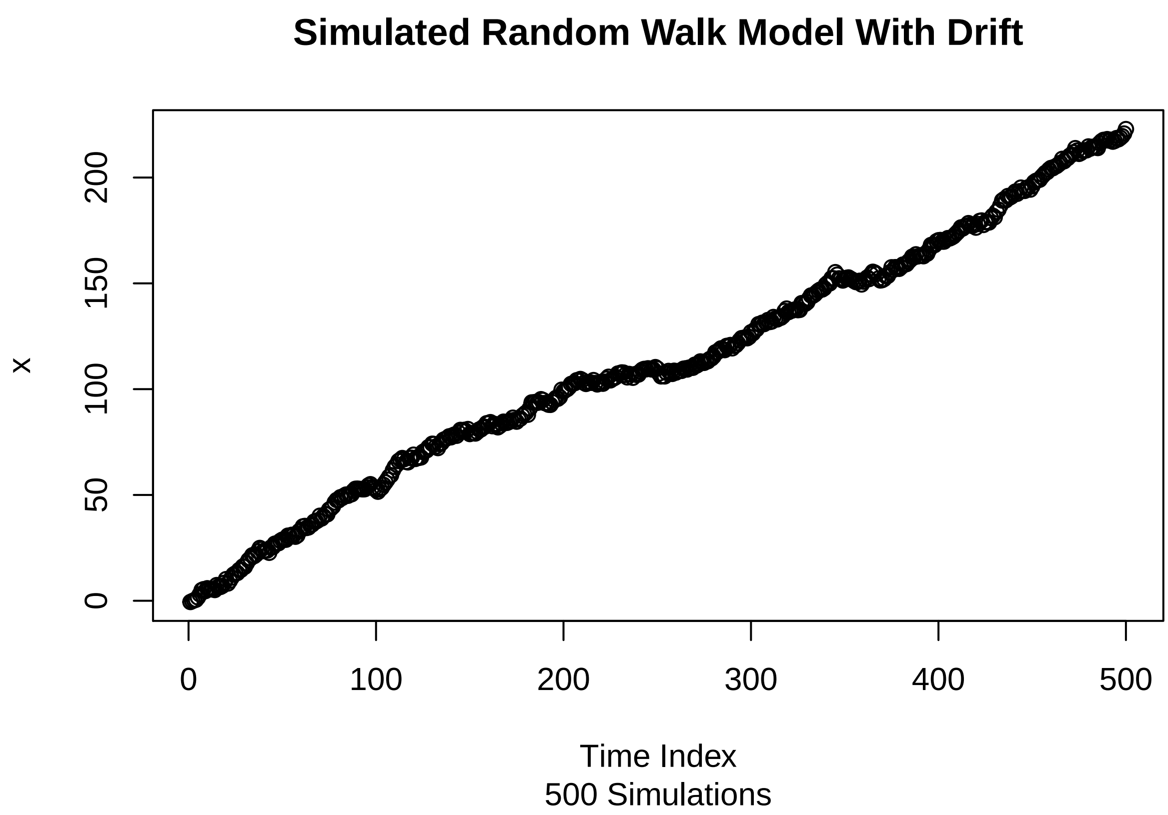
```
## [1] 59.16307
```

```r
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -0.579  70.680 108.600 112.600 156.500 222.900
```

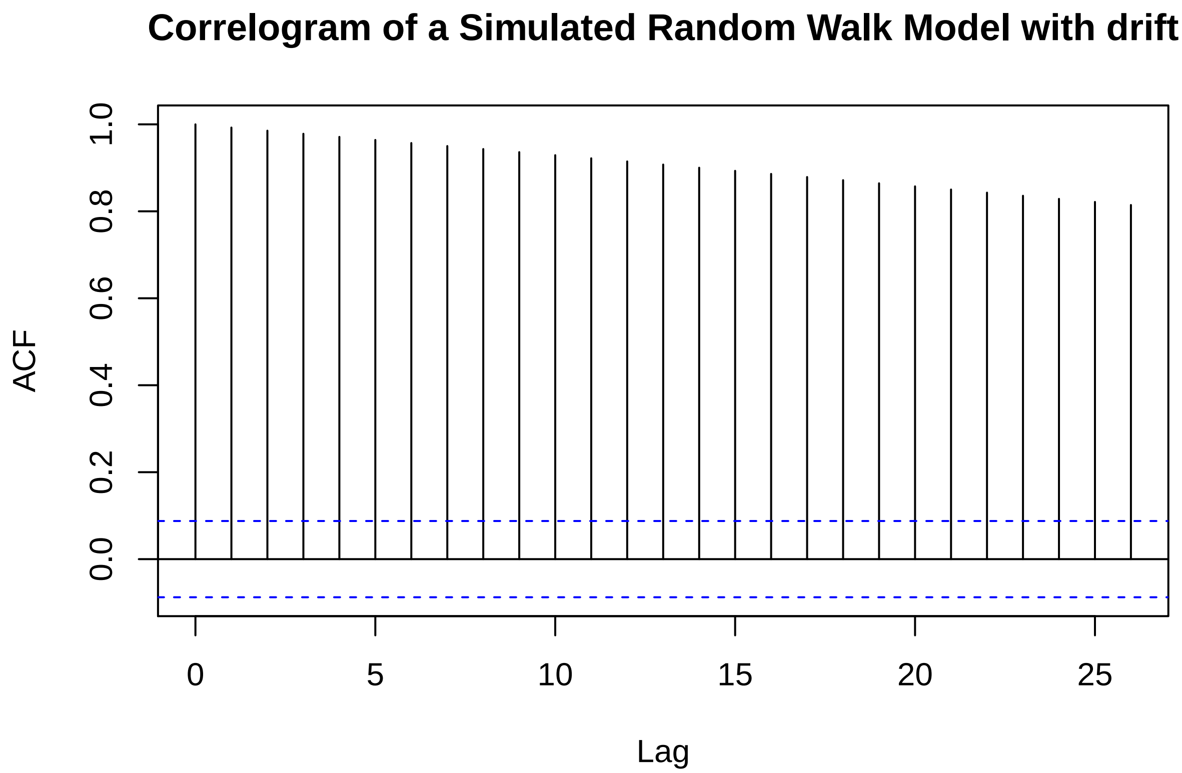
c. Plot the time-series plot of the simulated realizations

```r
plot(x, type="b", xlab="Time Index")
title("Simulated Random Walk Model With Drift", "500 Simulations")
```



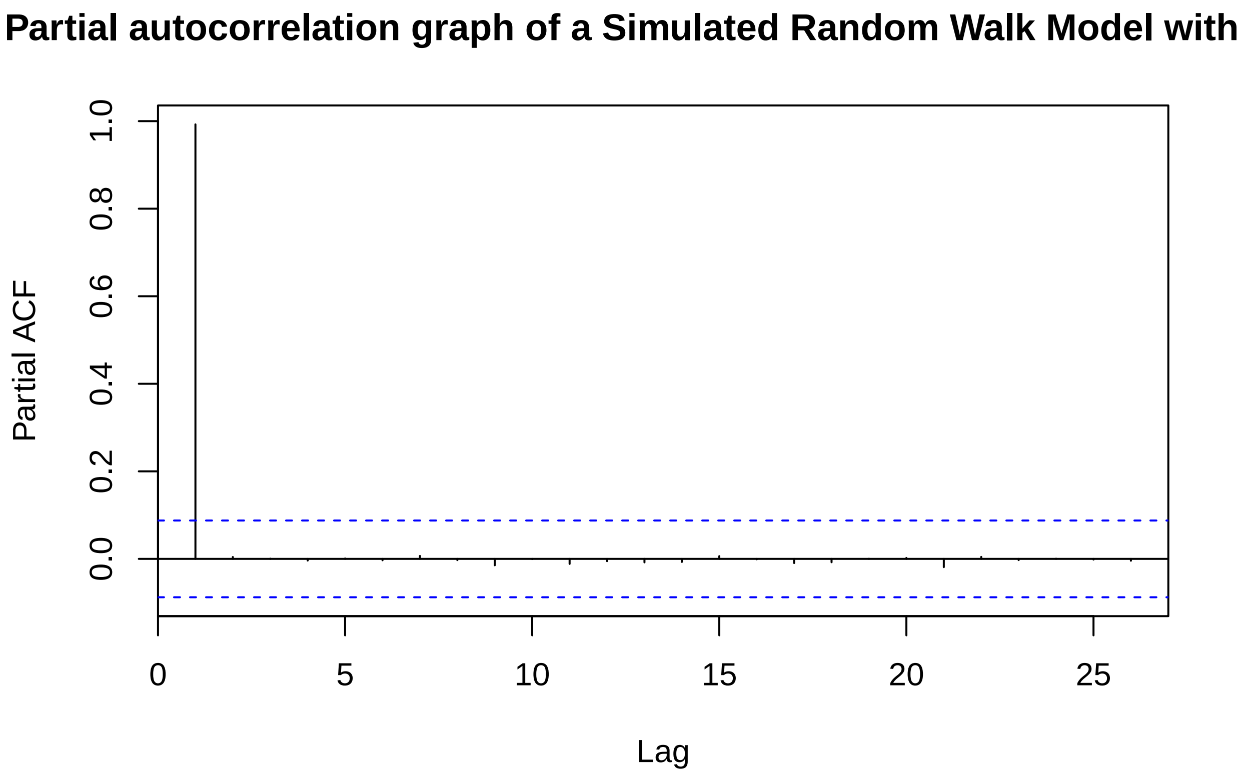
d. Plot the autocorrelation graph

```r
acf(x, main="")
title("Correlogram of a Simulated Random Walk Model with drift")
```



e. Plot the partial autocorrelation graph

```
pacf(x, main="")
title("Partial autocorrelation graph of a Simulated Random Walk Model with drift")
```



Exercise 4:

Use the series from INJCJC.csv a. Load the data and examine the basic structure of the data using str(), dim(), head(), and tail() functions

```
unemp = read.csv("C:/Subha/WS271-Regression/Labs/data/INJCJC.csv")
str(unemp)
```

```
## 'data.frame':    1300 obs. of  3 variables:
##  $ Date   : Factor w/ 1300 levels "1-Apr-05","1-Apr-11",..: 1102 143 442 784 483 1271 312 654 498 128
##  $ INJCJC : int  355 369 375 345 368 367 348 350 351 349 ...
##  $ INJCJC4: num  362 366 364 361 364 ...
```

```
dim(unemp)
```

```
## [1] 1300    3
```

```
head(unemp)
```

```
##        Date INJCJC INJCJC4
## 1  5-Jan-90    355  362.25
## 2 12-Jan-90    369  365.75
```

```
## 3 19-Jan-90    375  364.25
## 4 26-Jan-90    345  361.00
## 5  2-Feb-90    368  364.25
## 6  9-Feb-90    367  363.75
```

```r
tail(unemp)
```

```
##           Date INJCJC INJCJC4
## 1295 24-Oct-14    288  281.25
## 1296 31-Oct-14    278  279.00
## 1297  7-Nov-14    293  285.75
## 1298 14-Nov-14    292  294.25
## 1299 21-Nov-14    314  294.25
## 1300 28-Nov-14    297  299.00
```

b. Convert the variables INJCJC into a time series object frequency=52, start=c(1990,1,1), end=c(201411,28). Examine the converted data series

```r
unemp.ts = ts(unemp$INJCJC, frequency=52, start=c(1990,1,1),end=c(2014,11,28))
str(unemp.ts)
```

```
##  Time-Series [1:1259] from 1990 to 2014: 355 369 375 345 368 367 348 350 351 349 ...
```

```r
summary(unemp.ts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   259.0   327.0   355.0   373.3   408.0   665.0
```

c. Define a variable using the command INJCJC.time<-time(INJCJC)

```r
INJCJC.time<-time(unemp.ts)
```

d. Using the following command to examine the first 10 rows of the data. Change the parameter to examine different number of rows of data head(cbind(INJCJC.time, INJCJC),10)

```r
head(cbind(INJCJC.time, unemp.ts),10)
```

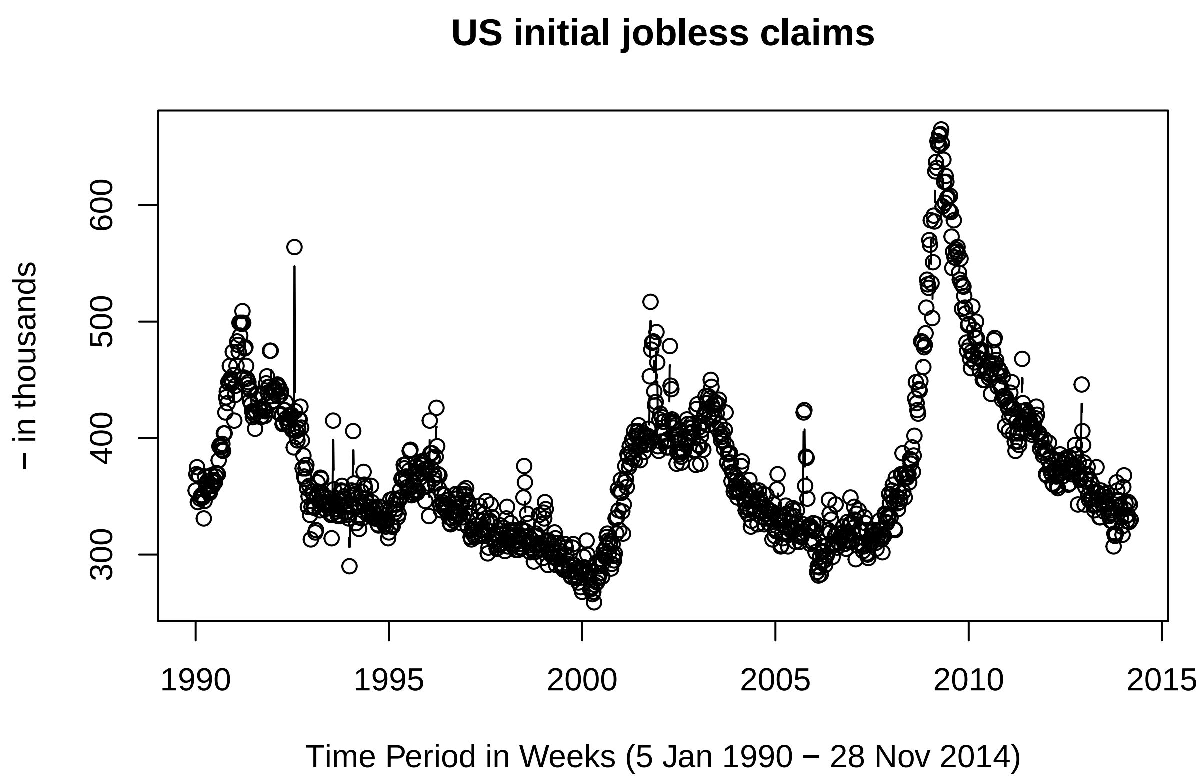
```
##       INJCJC.time unemp.ts
##  [1,]    1990.000      355
##  [2,]    1990.019      369
##  [3,]    1990.038      375
##  [4,]    1990.058      345
##  [5,]    1990.077      368
##  [6,]    1990.096      367
##  [7,]    1990.115      348
##  [8,]    1990.135      350
##  [9,]    1990.154      351
## [10,]    1990.173      349
```

```r
head(cbind(INJCJC.time, unemp.ts),5)
```

```
##      INJCJC.time unemp.ts
## [1,]    1990.000      355
## [2,]    1990.019      369
## [3,]    1990.038      375
## [4,]    1990.058      345
## [5,]    1990.077      368
```

e1. Plot the time series plot of INJCJC. Remember that the graph must be well labelled.

```r
plot(unemp.ts, type="b", xlab="Time Period in Weeks (5 Jan 1990 - 28 Nov 2014)", ylab = "- in thousands
title("US initial jobless claims")
```



e2. Plot the histogram of INJCJC. What is shown and not shown in a histogram? How do you decide the number of bins used?

e3. Plot the autocorrelation graph of INJCJC series e4. Plot the partial autocorrelation graph of INJCJC series 2 e5. Plot a 3x3 Scatterplot Matrix of correlation against lag values f1. Generate two symmetric Moving Average Smoothers. Choose the number of moving average terms such that one of the smoothers is very smoother and the other one can trace through the dynamics of the series. Plot the smoothers and the original series in one graph. f2. Generate two regression smoothers, one being a cubic trend regression and the other being a periodic regression. Plot the smoothers and the original series in one graph. f3. Generate kernel smoothers. Choose the smoothing parametrs such that one of the smoothers is very smoother and the other one can trace through the dynamics of the series. Plot the smoothers and the original series in one graph. f4. Generate two nearest neighborhood smoothers. Choose the smoothing parametrs such that one of the smoothers is very smoother and the other one can trace through the dynamics of the series. Plot the smoothers and the original series in one graph. f5. Generate two LOWESS smoothers. Choose the smoothing parametrs such that one of the smoothers is very smoother and the other one can trace through the dynamics of the series. Plot the smoothers and the original series in one graph. f5. Generate two spline smoothers. Choose the smoothing parametrs such that one of the smoothers is very smoother and the other one can trace through the dynamics of the series. Plot the smoothers and the original series in one graph. 3