# **W271**-2 – Spring 2016 – **HW 6**

**Juanjo Carin, Kevin Davis, Ashley Levato, Minghu Song**

March 16, 2016

## Contents



---

# Exercises

## Exercise 1

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

...

b. **Define strict and weak statonarity**

...

## Exercise 2

a. **Generate a zero-drift random walk model using 500 simulation.**

```
set.seed(123)
N <- 500 # number of simulations / time periods
wn <- rnorm(n = N, mean = 0, sd = 1) # white noise (can use any mean and sd)
rw <- cumsum(wn)
```

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.**

```
# See the definition of the function in ## @knitr Libraries-Functions-Constants
desc_stat(rw, 'Random walk',
          'Descriptive statistics of the simulated random walk')
```

Table 1: Descriptive statistics of the simulated random walk

| | Random walk |
|---|---|
| Mean | 4.58 |
| St. Dev | 4.87 |
| 1st Quartile | 0.97 |
| Median | 3.42 |
| 3rd Quartile | 8.27 |
| Min | -5.16 |
| Max | 19.35 |

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

See the last part of this Exercise in the following page.

d. **Plot the autocorrelation graph.**

See the last part of this Exercise in the following page.

e. **Plot the partial autocorrelation graph.**

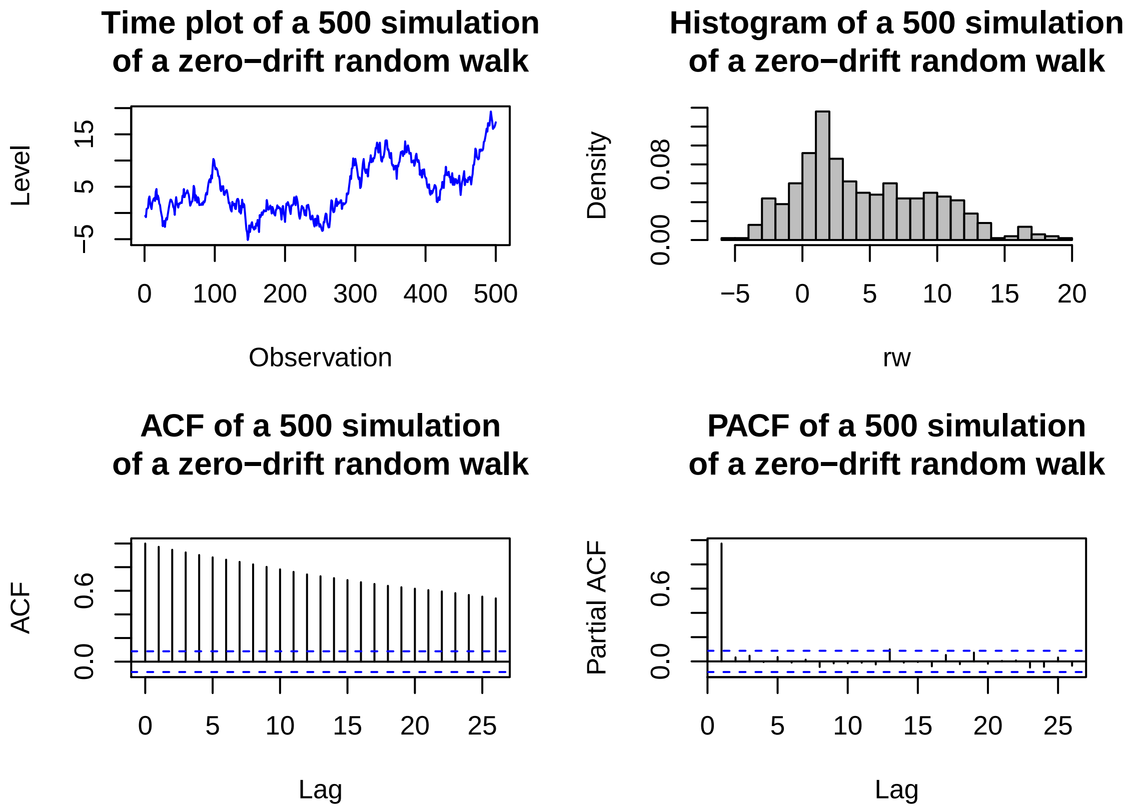


Figure 1: Time-series plot, histogram, correlogram, and partial autocorrelogram of the 500 simulation of a zero-drift random walk

## Exercise 3

a. **Generate a random walk with drift model using 500 simulation, with the drift = 0.5.**

```
drift <- 0.5 # drift
# Use the same GWN that genereate the prev. zero-drift RW
# set.seed(123); wn <- rnorm(n = N, mean = 0, sd = 1)
rw_drift <- cumsum(wn + drift)
```

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.**

```
# See the definition of the function in ## @knitr Libraries-Functions-Constants
desc_stat(cbind(rw, rw_drift), c('Random walk', 'Random walk with 0.5 drift'),
          'Descriptive statistics of the two simulated random walks')
```

Table 2: Descriptive statistics of the two simulated random walks

| | Random walk | Random walk with 0.5 drift |
|---|---|---|
| Mean | 4.58 | 129.83 |
| St. Dev | 4.87 | 75.43 |
| 1st Quartile | 0.97 | 64.58 |
| Median | 3.42 | 122.91 |
| 3rd Quartile | 8.27 | 199.68 |
| Min | -5.16 | -0.06 |
| Max | 19.35 | 267.30 |

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

See the last part of this Exercise in the following page.

d. **Plot the autocorrelation graph.**

See the last part of this Exercise in the following page.

e. **Plot the partial autocorrelation graph.**

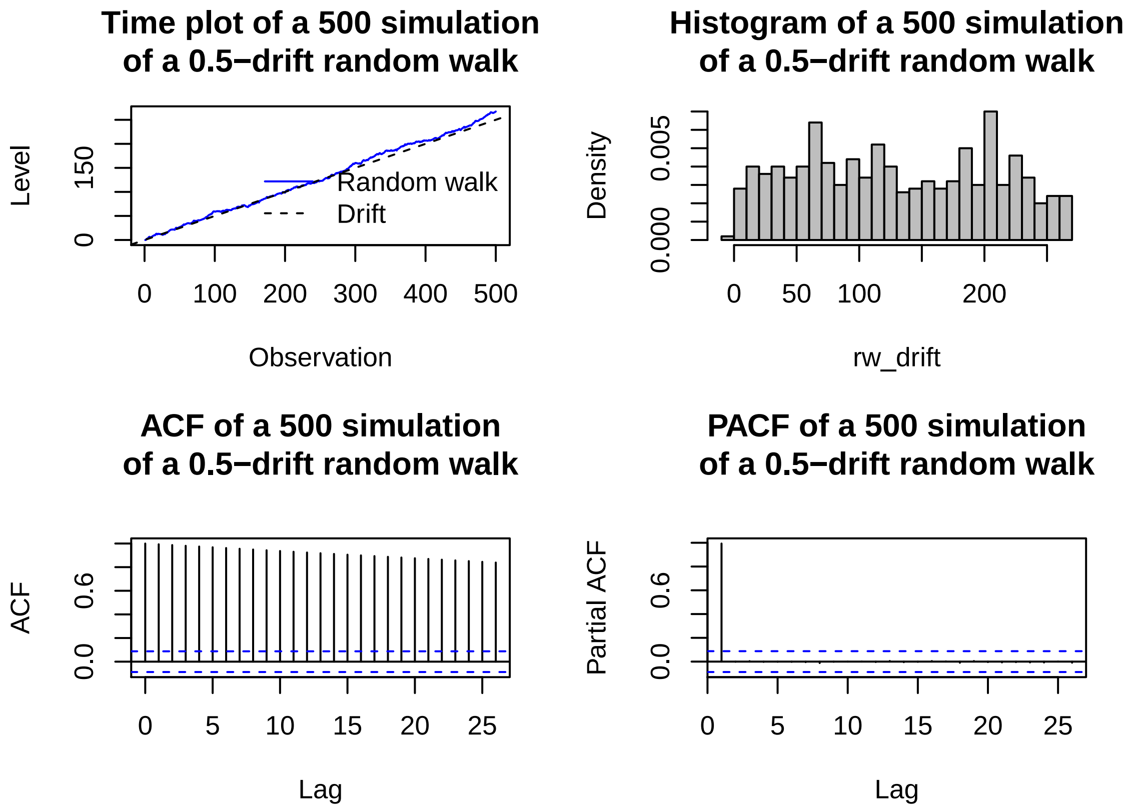


Figure 2: Time-series plot, histogram, correlogram, and partial autocorrelogram of the 500 simulation of a 0.5-drift random walk

## 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.**

...

b. **Convert the variables `INJCJC` into a time series object `frequency=52, start=c(1990,1,1), end=c(2014,11,28)`. Examine the converted data series.**

...

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

...

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)
```

...

e.

1. **Plot the time series plot of `INJCJC`. Remember that the graph must be well labelled.**

...

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

...

3. **Plot the autocorrelation graph of `INJCJC` series.**

...

4. **Plot the partial autocorrelation graph of `INJCJC` series.**

...

5. **Plot a 3x3 Scatterplot Matrix of correlation against lag values.**

...

f.

1. **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.**

...

2. **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.**

...

3. **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.**

...

4. **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.**

...

5. **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.**

...

6. **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.**

...

---