W271 - Applied Regression and Time Series Analysis - HW5

Subhashini R., Lei Yang, Ron Cordell

March 10, 2016

Question 1

1). Install the library "astsa" using the function: install.packages("astsa")

2). Load the library: library(astsa)

```
library(astsa)
```

3). Use the function str() to see the information of a particular data series, such as str(EQ5) for the Seismic Trace of Earthquake number 5 series

```
str(EQ5)
```

```
##  Time-Series [1:2048] from 1 to 2048: 0.01749 0.01139 0.01512 0.01477 0.00651 ...
```

```
str(flu)
```

```
##  Time-Series [1:132] from 1968 to 1979: 0.811 0.446 0.342 0.277 0.248 ...
```

```
str(gas)
```

```
##  Time-Series [1:545] from 2000 to 2010: 70.6 71 68.5 65.1 67.9 ...
```

4). Plot the time series plots and histograms of the following 3 series. Feel free to use the codes provided in the R scripts. Make sure that each of your graph has a title, the axis ticks are clear, the axes are well-labelled, and use color intelligently.
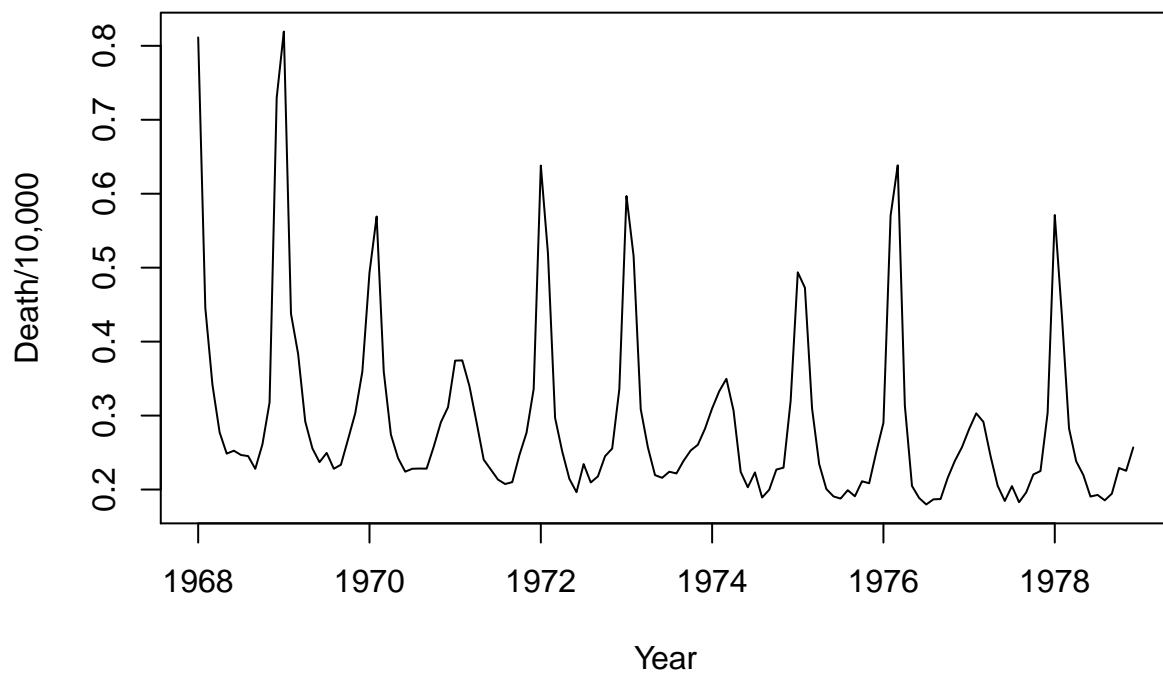
```
#par(mfrow=c(3,1)) # bad viz effect in PDF
plot.ts(EQ5, main="Seismic Trace of Earthquake #5", ylab="Amplitude", xlab="Sec.")
```

Seismic Trace of Earthquake #5
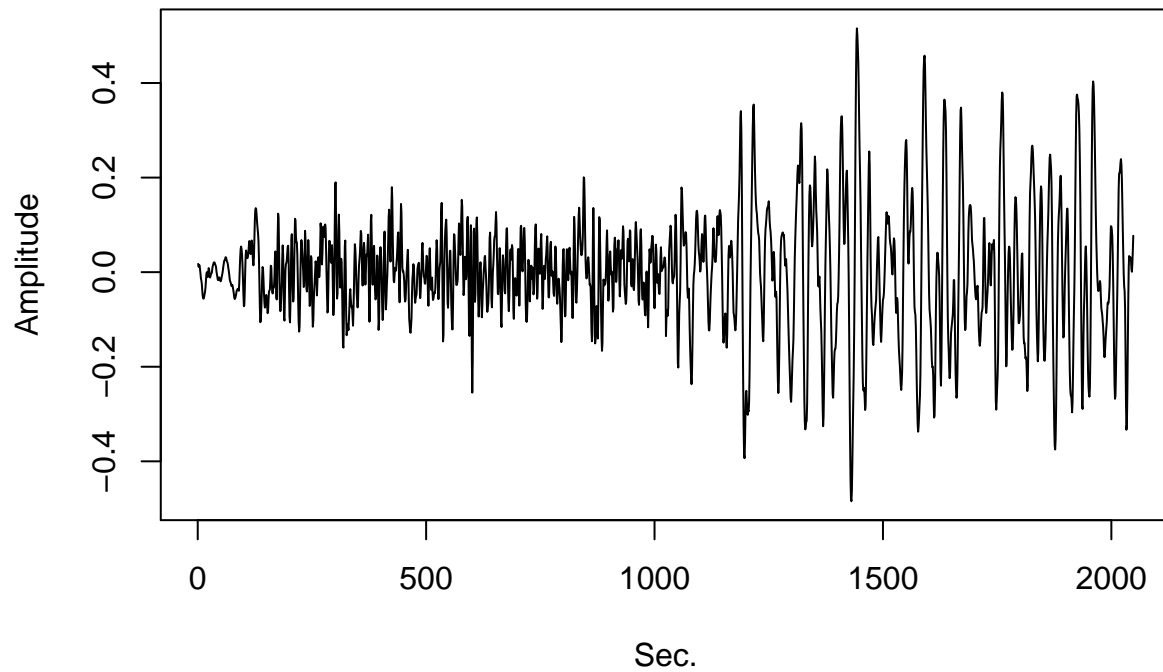

```
plot.ts(flu, main="Monthly pneumonia and influenza deaths
    per 10,000 people in the United States", ylab="Death/10,000", xlab="Year")
```

Monthly pneumonia and influenza deaths
per 10,000 people in the United States

```
plot.ts(gas, main="New York Harbor conventional regular gasoline weekly
    spot price FOB", ylab="cents/gal", xlab="Year")
```

New York Harbor conventional regular gasoline weekly spot price FOB
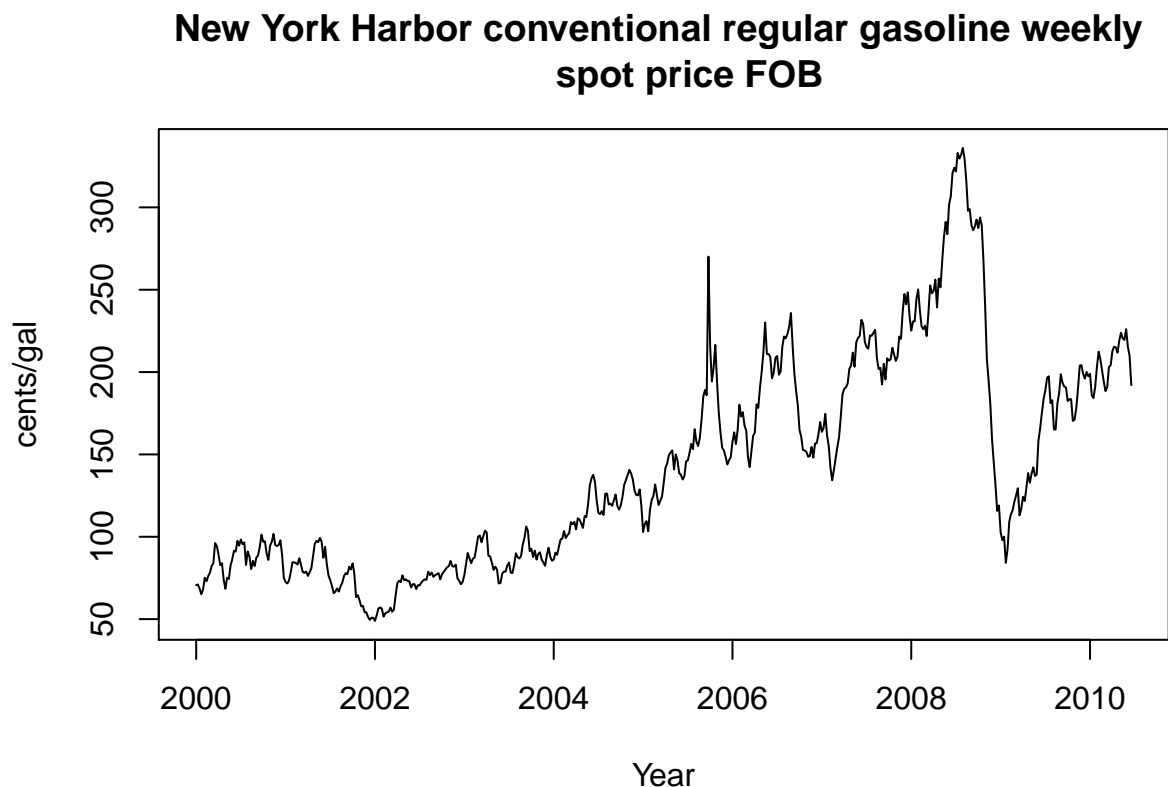
5). Write a few sentences to describe each of the series.

EQ5: the primary wave (t = 1, . . . , 1024) and the shear wave with much larger amplitude and violent fluctuation (t = 1025, . . . , 2048)

flu: there is a seasonal pattern in the death count, with a peak at the beginning of every year, then drops down in mid year, before goes up again end of the year.

gas: gas price stays flat (with minor drop) before 2002, then start an increasing trend, with some fluctuation between late 2005 and early 2008. A big price drop occurs early 2009, before it slows climbs back up.

Question 2

Describe 3 examples you have used in your work or encounter in real life. Ideally, you can even load at least one of these time series, plot it, and the write a few statements to describe its characteristics.

Example 1: Stock price

```
data<-read.csv("googl.csv", header=T)
plot(as.Date(data[,1], "%d-%b-%y"), data$Close, type="l", xlab="Date", ylab="Close Price ($)", col="blu
```

3

GOOG Stock Price
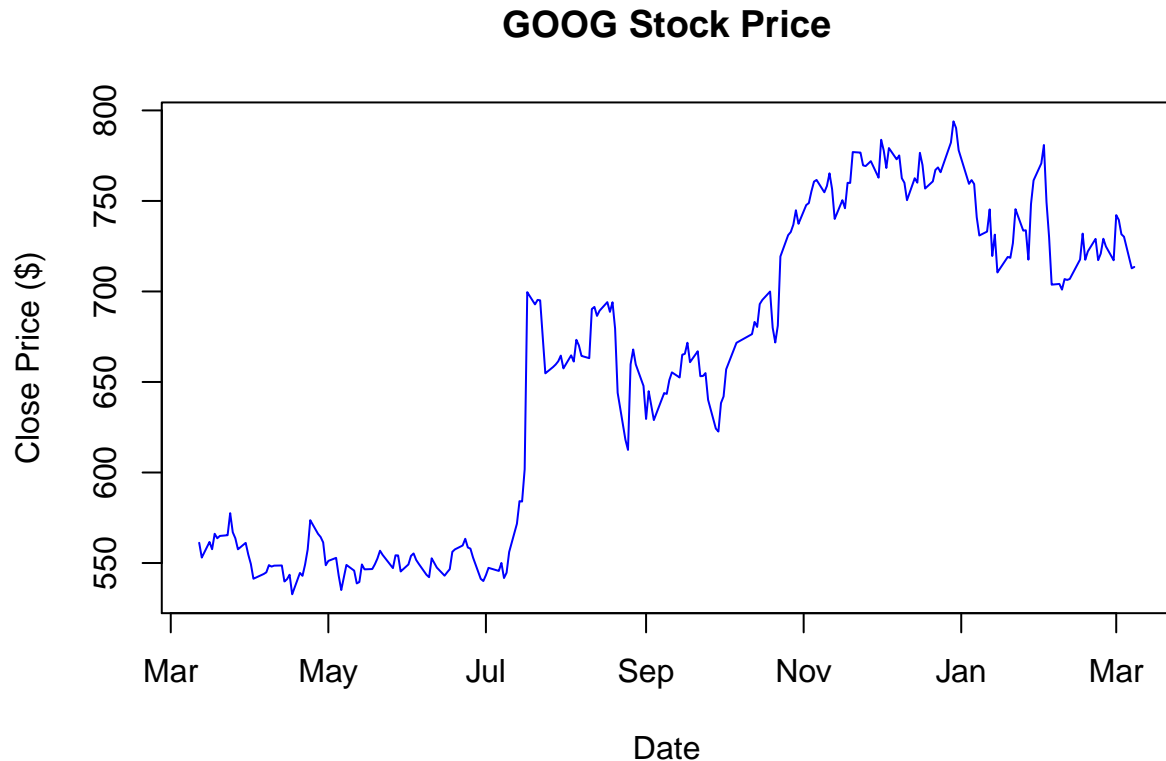
For Google, the price stays stable before mid July 2015, before have a big jump and steadily increase through January 2016, followed by some fluctuation after that.

Example 2: control parameters in a semiconductor process. Various sensors are recording time series data during the process, a feedback controller is adjusting setpoint in real time to ensure a stable chamber condition during the process.

Example 3: biometric data. With various wearables becoming popular, more biometric data are collected from daily life, including heartrate, sleep quality, activities, food intake, etc. Overtime those data yield different time series to indicate one's life style, healthness, and hopefully can be further utilized to predict or diagnose potential health problem.

Question 3

Simulate a white noise series with 1000 random draws and plot (1) a time series plot and (2) a histogram. The usual reqirements on graphics (described) in Question 1) applied.
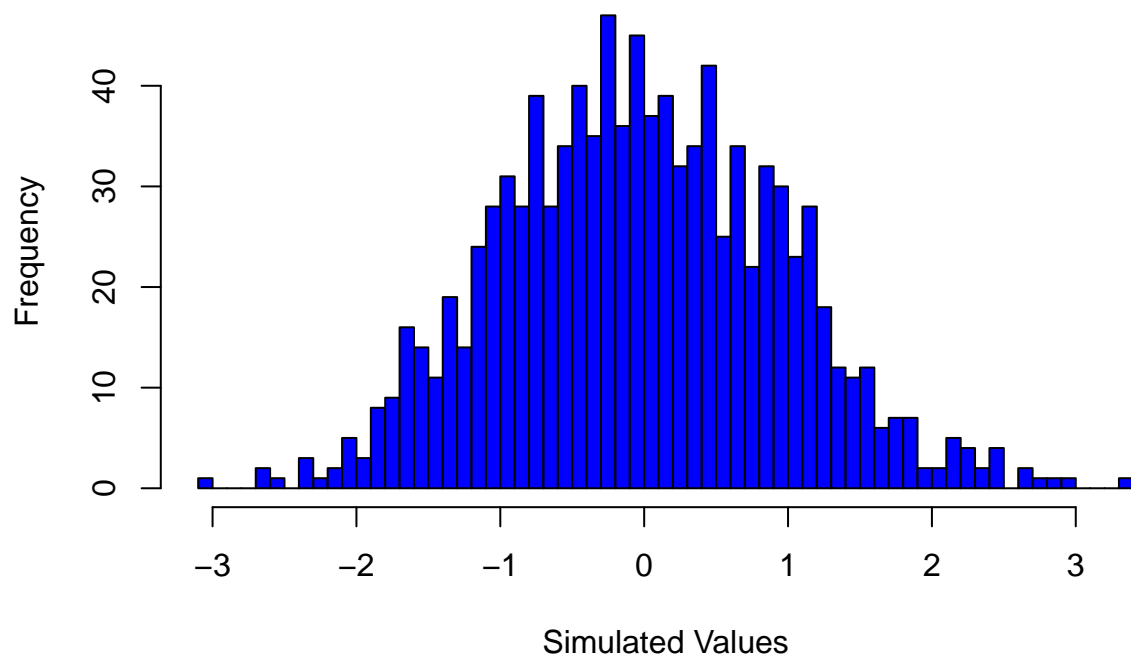
```
# white noise with sample size 1000
w=rnorm(1000,0,1)
# generate plots
plot.ts(w, main='Simulated White Noise Series', ylab='Magnitude')
```

Simulated White Noise Series
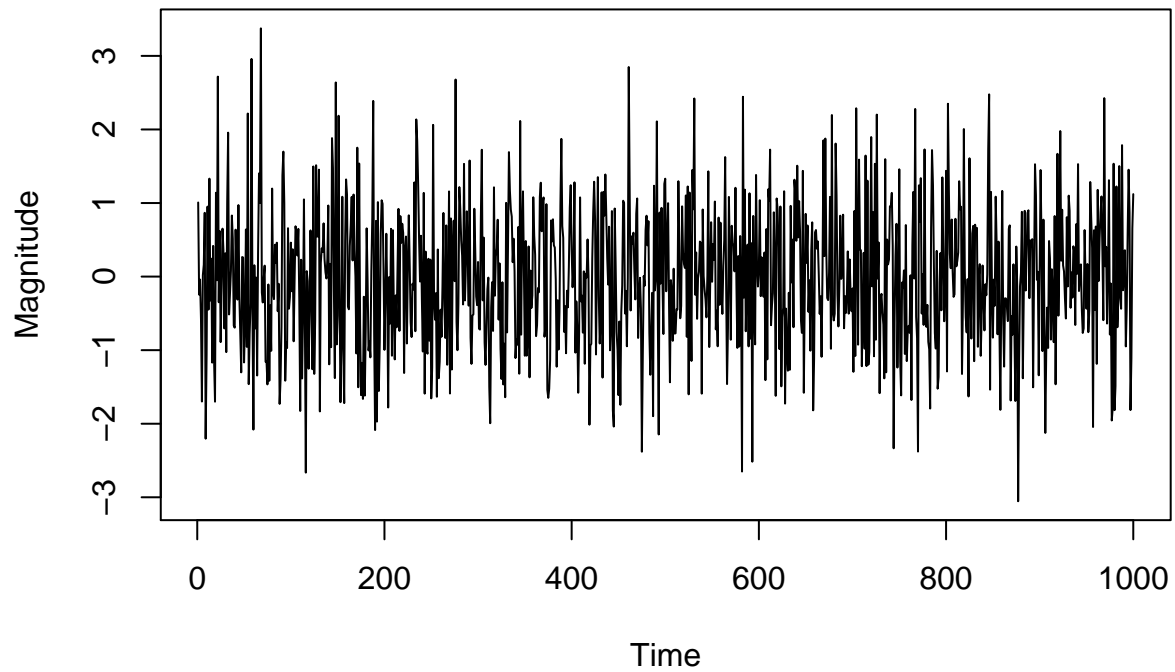

```r
hist(w, breaks=50, main='Histogram of 1000 Simulated White Noise',
     xlab='Simulated Values', col='blue')
```

Histogram of 1000 Simulated White Noise

Question 4

Simulate (with 1000 random draws) two the following two zero-mean autoregressive model with order 1 (i.e. AR(1)) models:

$$y_t = 0.9y_{t1} + w$$

$$y_t = 0.2y_{t1} + w$$

Plot a time plot for each of the simulated series. Graph a histogram for each of thes simulated series. Write a few statements to compare the two series.
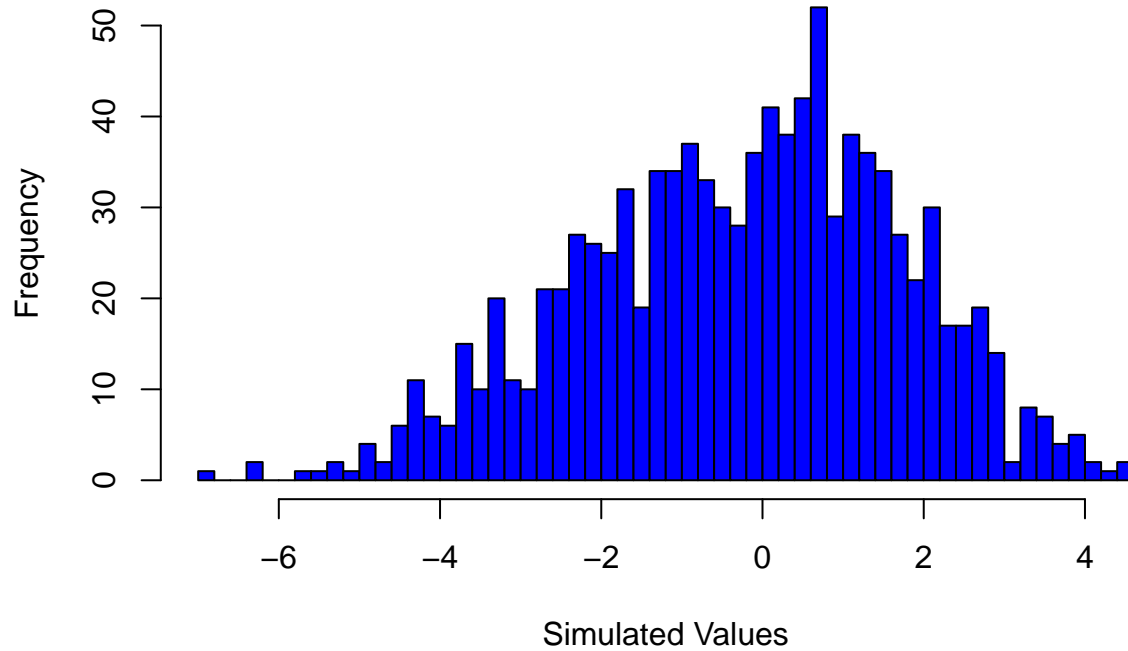
```
# first series
y1 <- w
for (t in 2:length(w)){
  y1[t] <- 0.9*y1[t-1] + w[t] # use the same random normal sequence generated above
}
# generate plots
plot.ts(y1, main="Simulated AR(ar=c(0.9)) Series", col="navy",
        ylab="Values of the Simluated Series",
        xlab="Simulated Time Period")
```

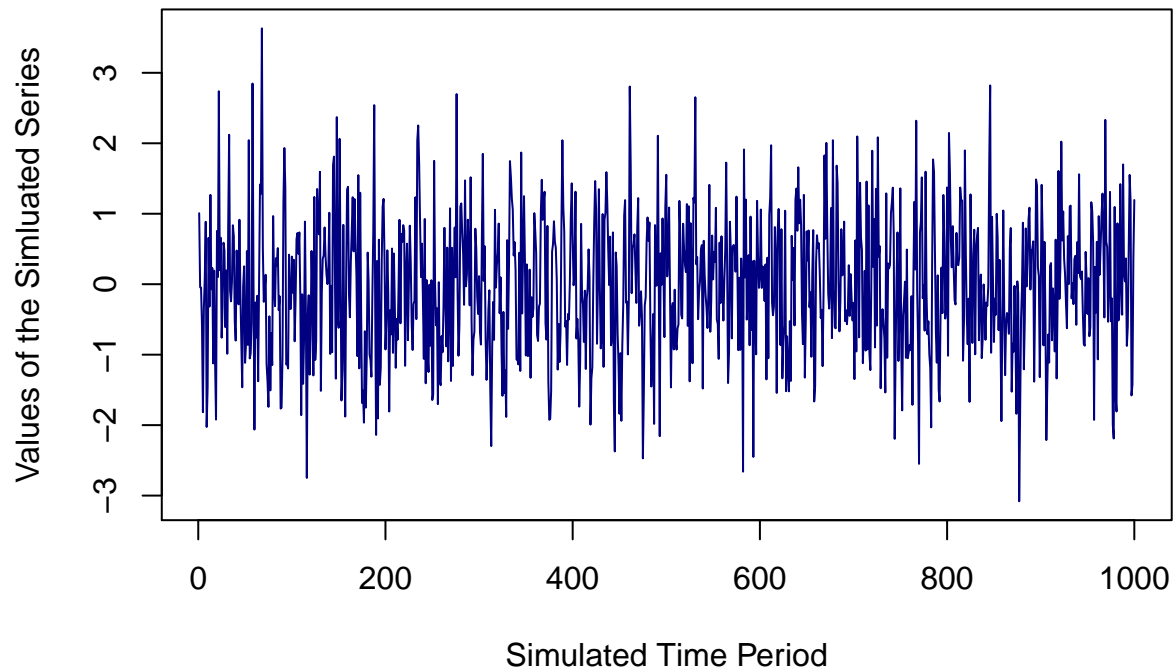## Simulated AR(ar=c(0.9)) Series


```
hist(y1, breaks=50, main='Histogram of Simulated AR(1) Series',
     xlab='Simulated Values', col='blue')
```
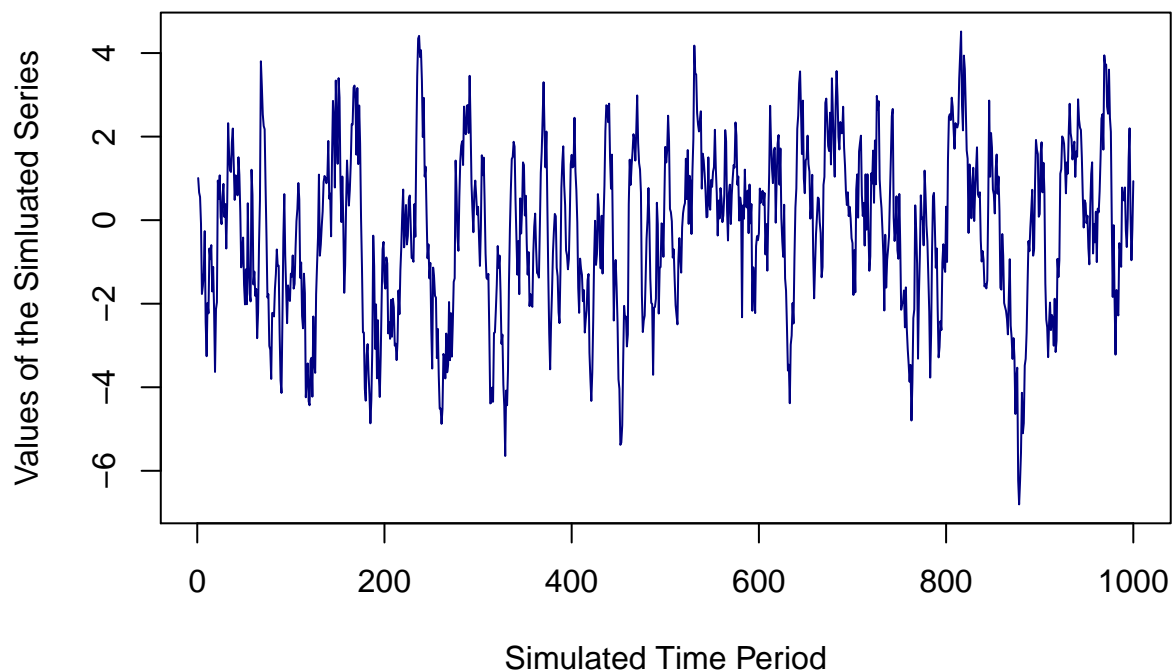
# Histogram of Simulated AR(1) Series


```
# second series
y2 <- w
for (t in 2:length(w)){
  y2[t] <- 0.2*y2[t-1] + w[t] # use the same random normal sequence generated above
}
# generate plots
plot.ts(y2, main="Simulated AR(ar=c(0.2)) Series", col="navy",
        ylab="Values of the Simluated Series",
        xlab="Simulated Time Period")
```

Simulated AR(ar=c(0.2)) Series


```
hist(y2, breaks=50, main='Histogram of Simulated AR(1) Series',
     xlab='Simulated Values', col='blue')
```

Histogram of Simulated AR(1) Series
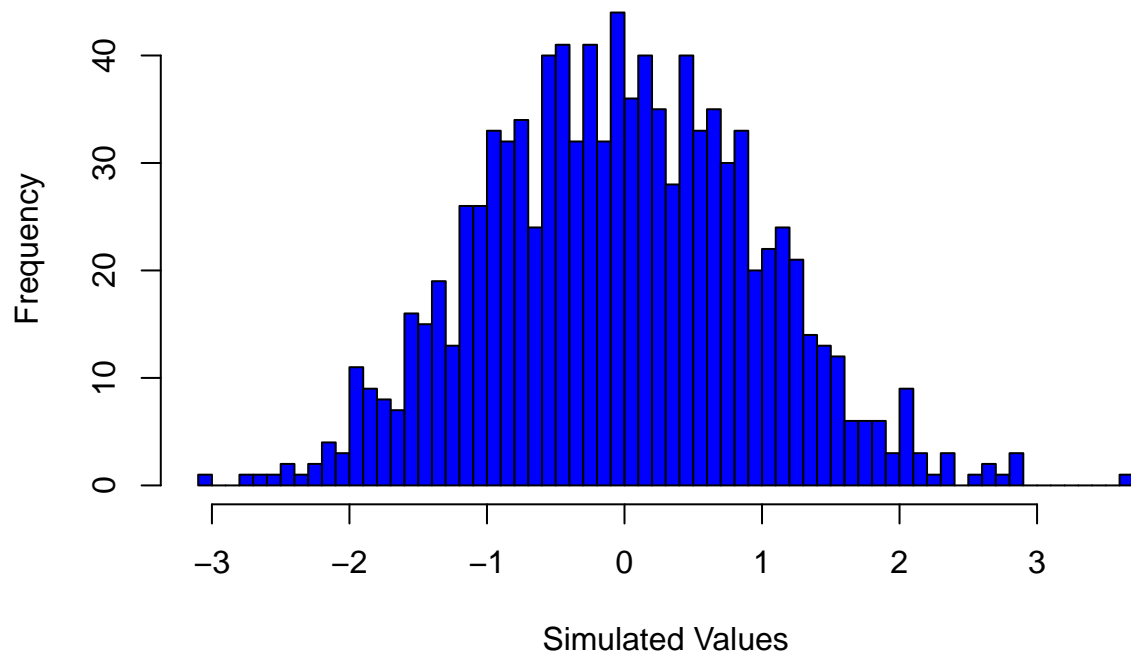
The time series plot of the first simulated series shows some cyclic pattern during the period, while the second series demonstrates a largely random pattern. The reason is first AR model has a bigger coefficient of

0.9, which damps the change of next value, thus the series shows milder fluctuation than the second series. Nevertheless both series follow normal distribution.

Question 5

Simulate (with 1000 random draws) the following 3 models, Plot a time plot for each of the simulated series. Graph a histogram for each of thes simulated series. Write a few statements to compare the two series.

1). A deterministic linear (time) trend of the form: $y_t = 10 + 0.5t$

```
# simulation
y1 <- w
for (t in 1:length(w)){
  y1[t] <- 10 + 0.5*t + w[t] # use the same random normal sequence generated above
}

# generate plots
plot.ts(y1, main="Simulated Deterministic Linear Trend", col="navy",
        ylab="Values of the Simluated Series",
        xlab="Simulated Time Period")
```

Simulated Deterministic Linear Trend
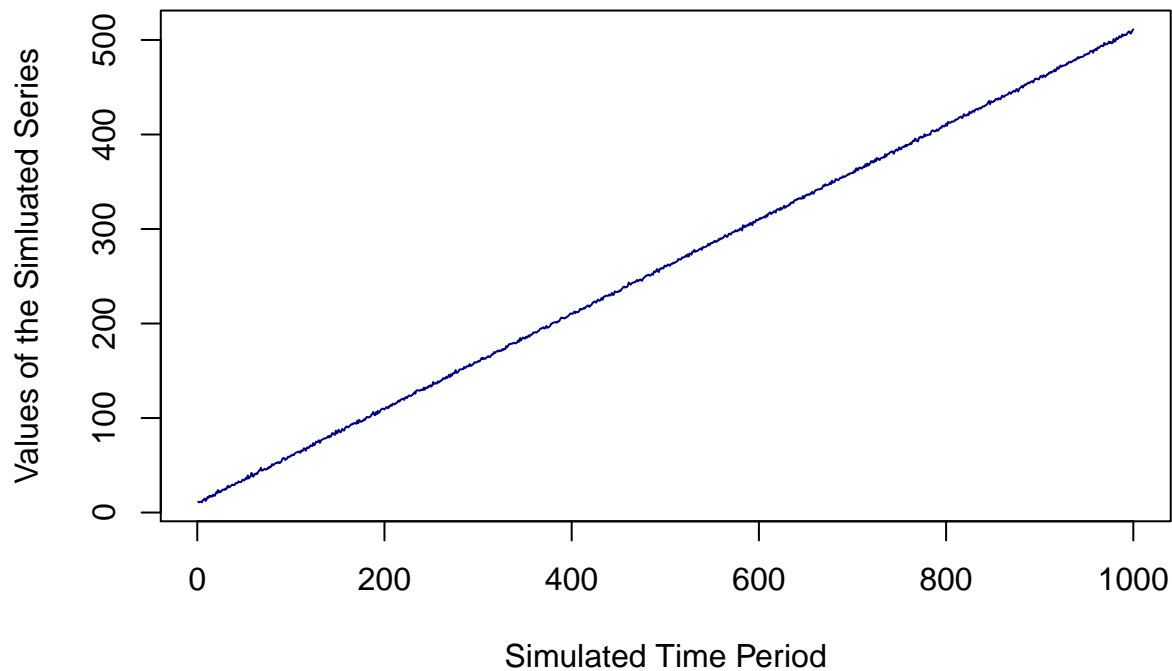

```
hist(y1, breaks=50, main='Histogram of Simulated Series',
     xlab='Simulated Values', col='blue')
```
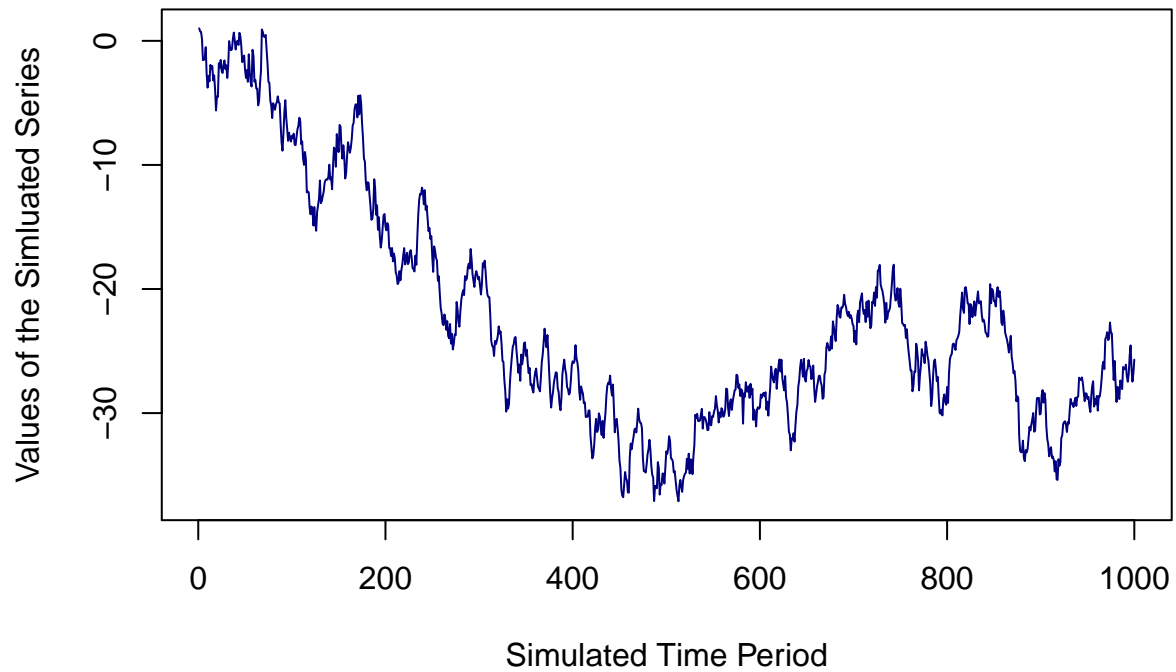
Histogram of Simulated Series
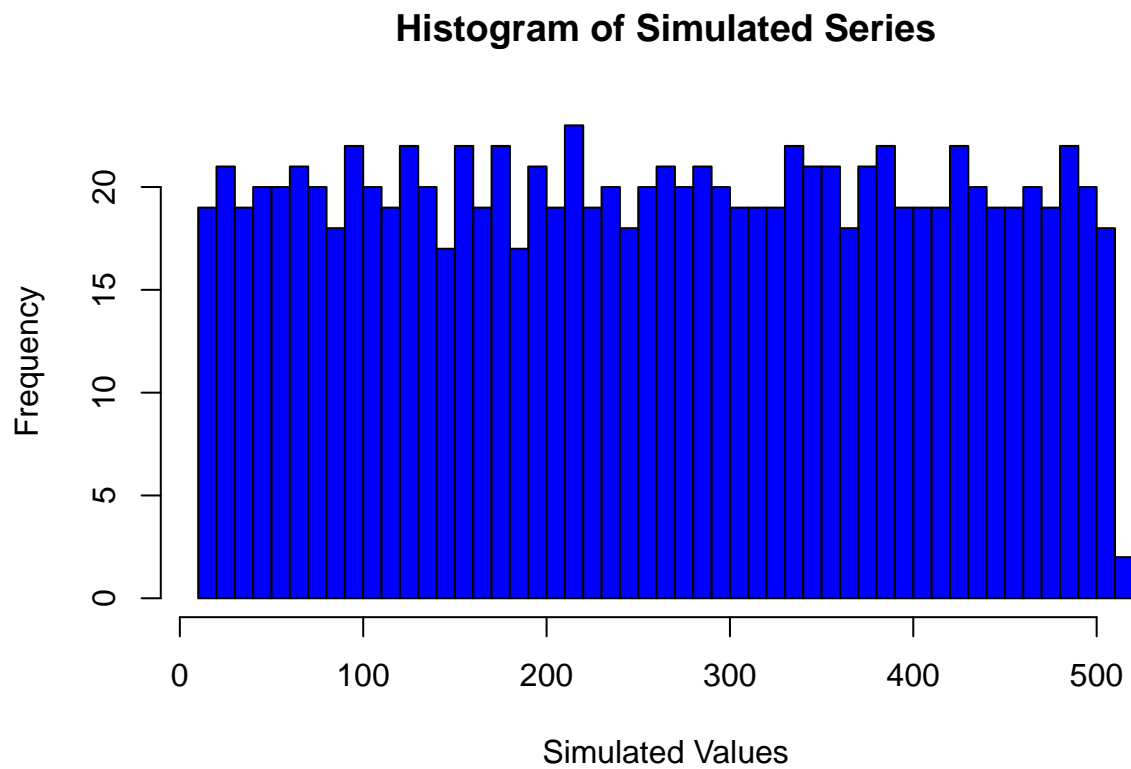
2). Random walk without drift

```
# Random walk with zero drift
x=cumsum(w)

# generate plots
plot.ts(x, main="Random Walk Without Drift", col="navy",
        ylab="Values of the Simluated Series",
        xlab="Simulated Time Period")
```
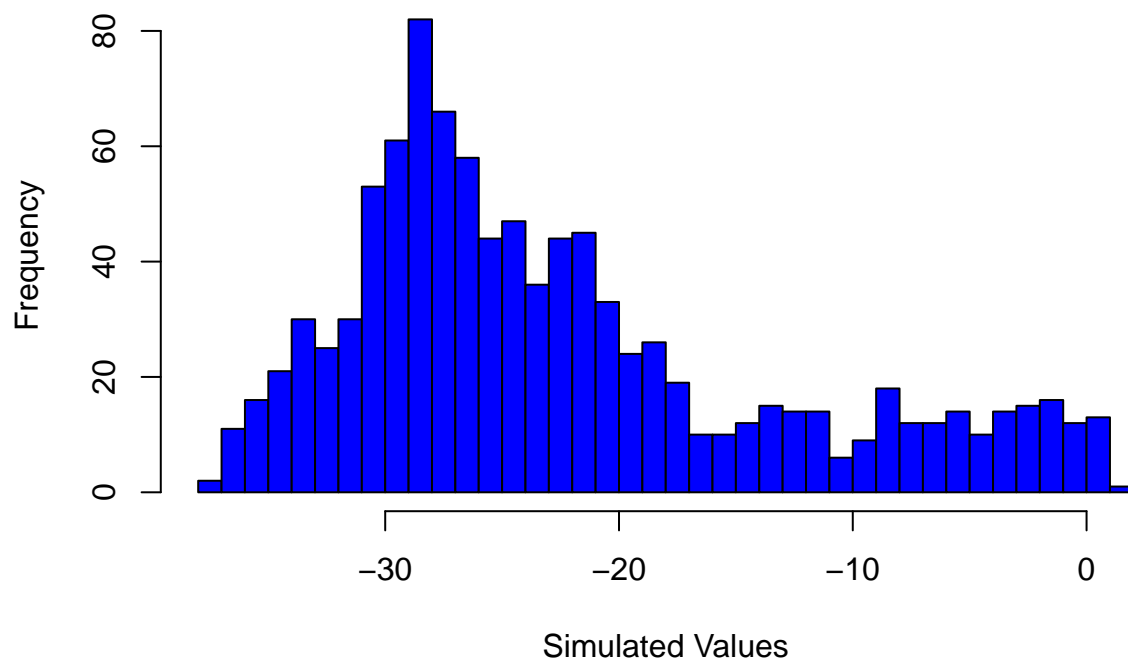
Random Walk Without Drift


```
hist(x, breaks=50, main='Histogram of Simulated Series',
     xlab='Simulated Values', col='blue')
```

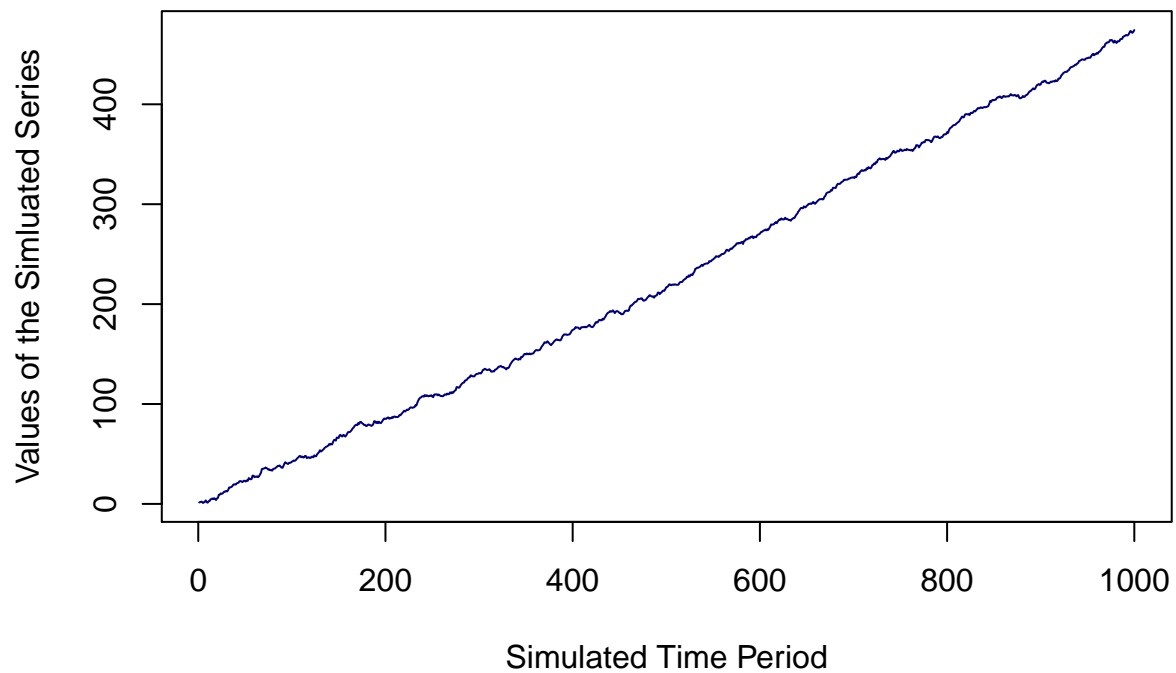## Histogram of Simulated Series
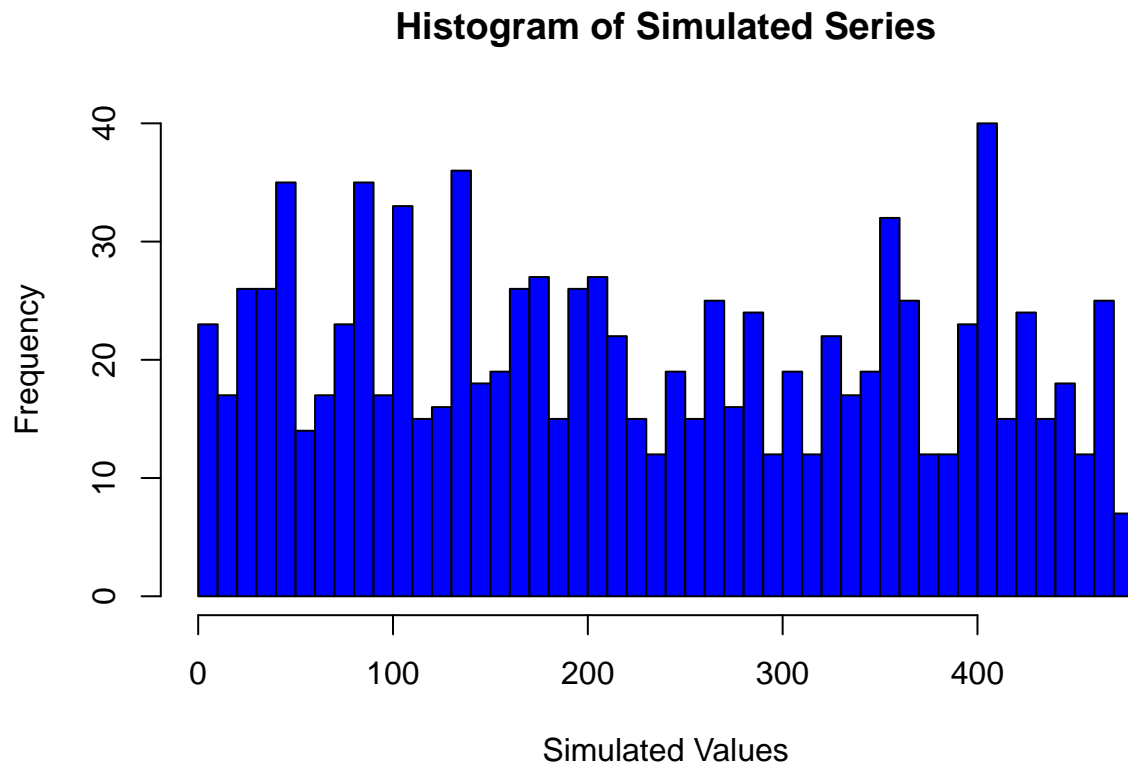

3). Random walk with drift = 0.5

```
# Random walk with drift = 0.5
wd = 0.5 + w;
xd = cumsum(wd)

# generate plots
plot.ts(xd, main="Random Walk With Drift=0.5", col="navy",
        ylab="Values of the Simluated Series",
        xlab="Simulated Time Period")
```

Random Walk With Drift=0.5


```
hist(xd, breaks=50, main='Histogram of Simulated Series',
     xlab='Simulated Values', col='blue')
```

Histogram of Simulated Series

Series (1) and (3) look very similar, in fact the drift factor (0.5) of random walk and the slope (0.5) in deterministic trend have equivalent roles in the series, in that every point is 0.5 bigger than the previous one. Series (2), meanwhile, doesn't have any deterministic pattern, and can demonstrate different shape between simulations.