Homework 1

Tony Silva

January 21, 2018

Contents

Please note: I utilized R Markdown to generate this PDF file.

Number 1

For number 1, I have reviewed the slides and videos multiple times. Additonally, I have been reading our textbooks for further mastering.

Number 2

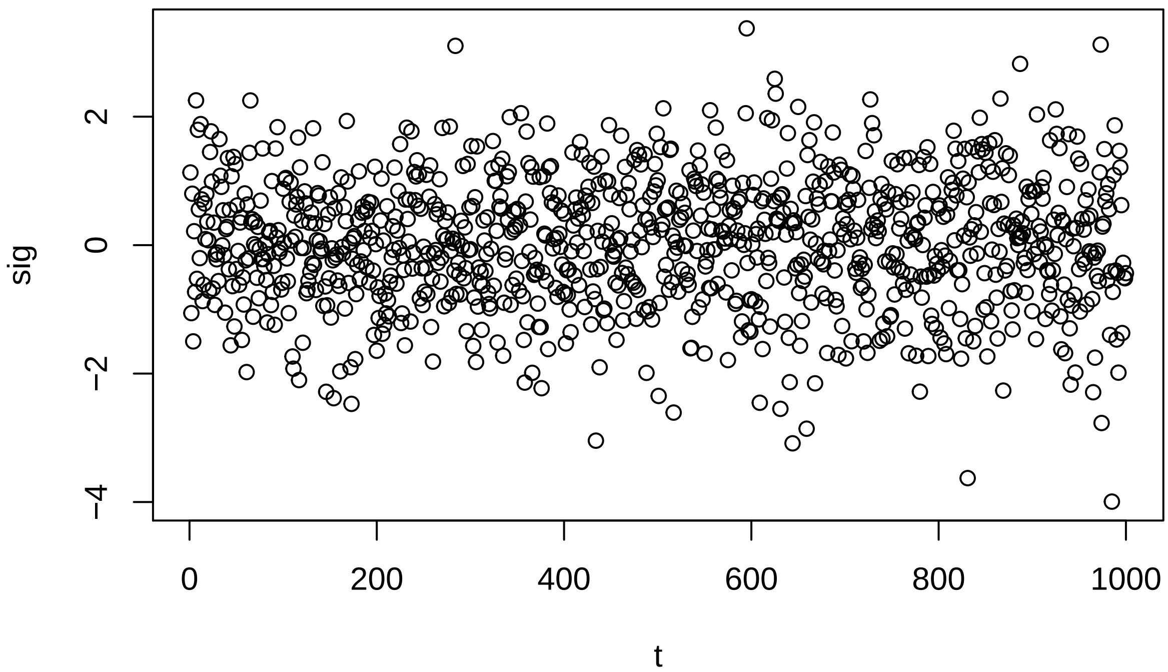
The following R code was utilized to generate sigma for problem 2. A normally distributed random variable of 1000 samples.

```
sig <- rnorm(1000, mean=0, sd=1)
```

Part A

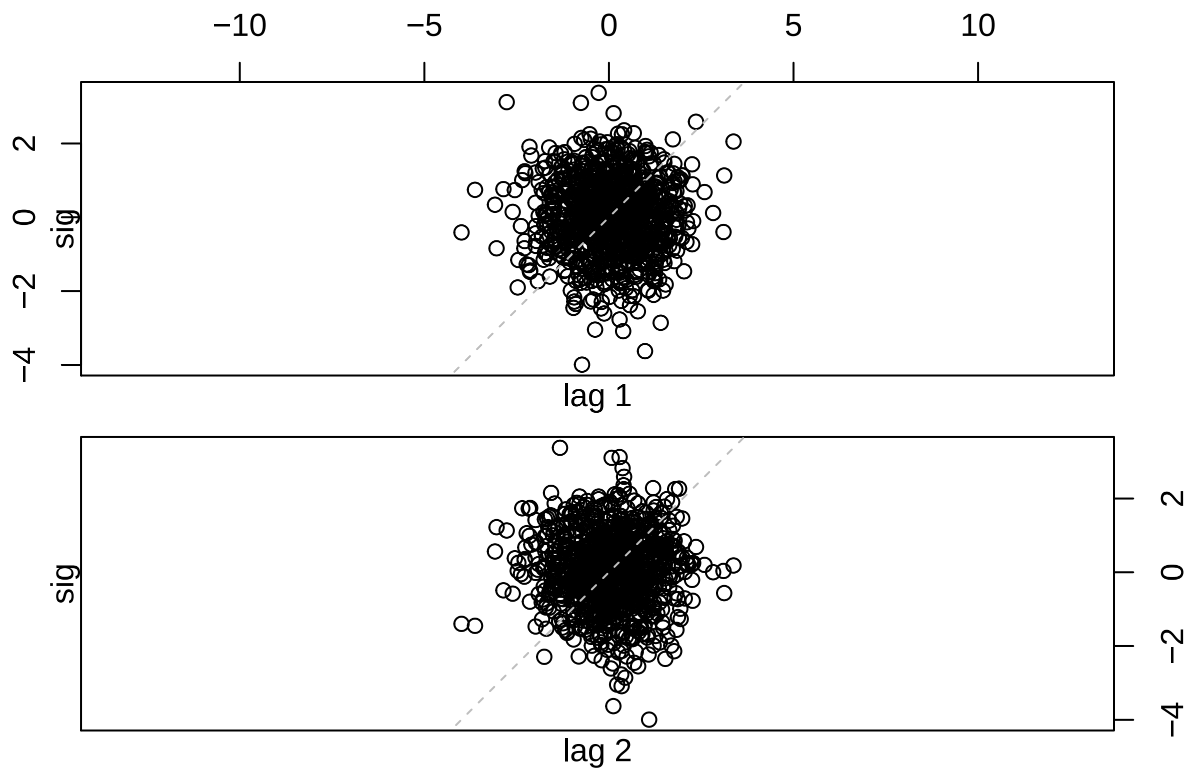
I then created a variable t from 1->1000 and plotted the two variables. This can be seen below.

```
t <- 1:1000
plot(x=t,y=sig)
```



I then plotted sigma versus the two different lags, t-1 and t-2. In order to create the lag plots, I used the lag.plot in the base R library.

```
lag.plot(sig, lags=2)
```



From the plots we can see that there are not any trends between the time series and the lag of t-1 and t-2 of the time series. This is important because these plots are useful in helping us identify randomness, covariance and correlation. The points in the plot are more dense around (0,0) in plot.

Part B

I then calculated the Mean and Variance of the time series using the following:

```r
mean(sig)
```

```
## [1] 0.03910172
```

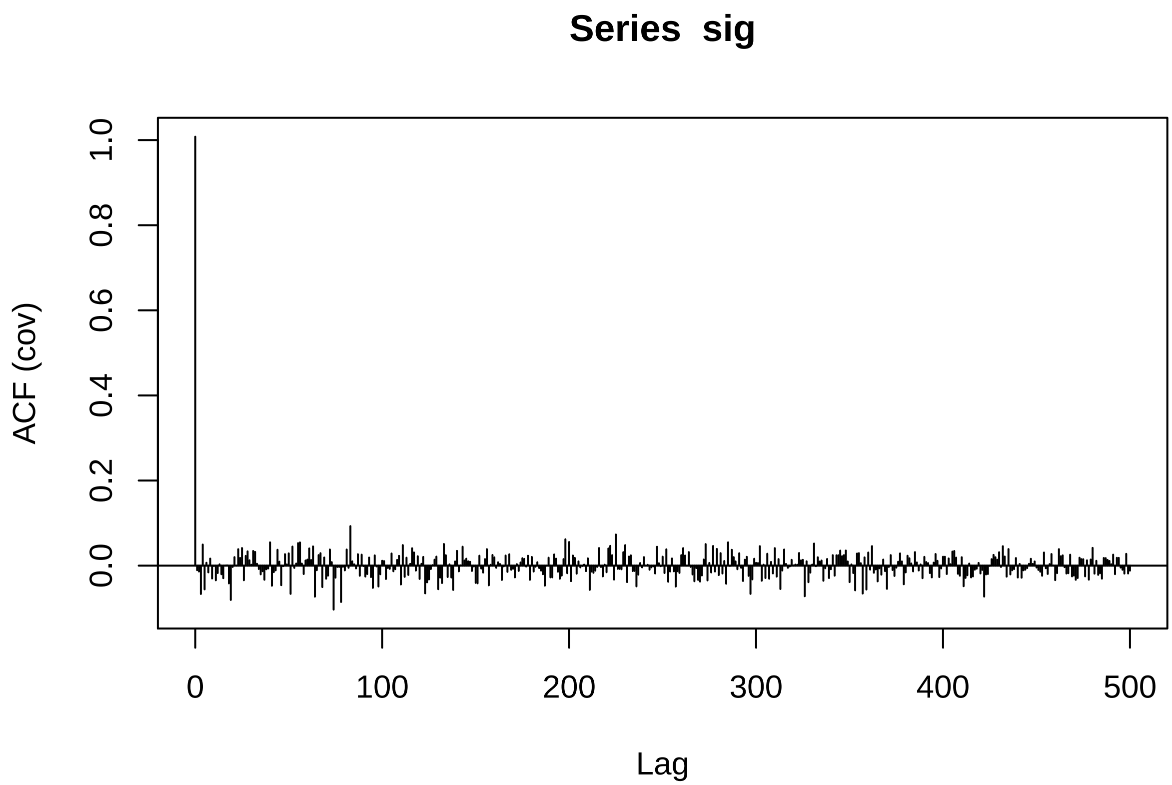
```r
var(sig)
```

```
## [1] 1.008953
```

As you can see the mean and variance are essentially the Gaussian normal, since we took a sample of 1000. As we add more samples the closer and closer we will be to the population parameters.

Next I calculate the auto-covariance and auto-correlation for each of the time lags 0 <= k <= 500. This can be seen below. I utilized the base R package called acf, which enables you to calculate both covariance and correlation across different lag metrics. In our case, our lag.max is 500. Additionally, I calculated the total average of the auto-covariance and auto-correlation using the mean() function. The plots below the auto-covariance and auto-correlation for the different k values. Please note the tall line on the left of each plot is the covariance/correlation of lag=0, which essentially is the covariance/correlation of the time series with itself, which will always be 1.

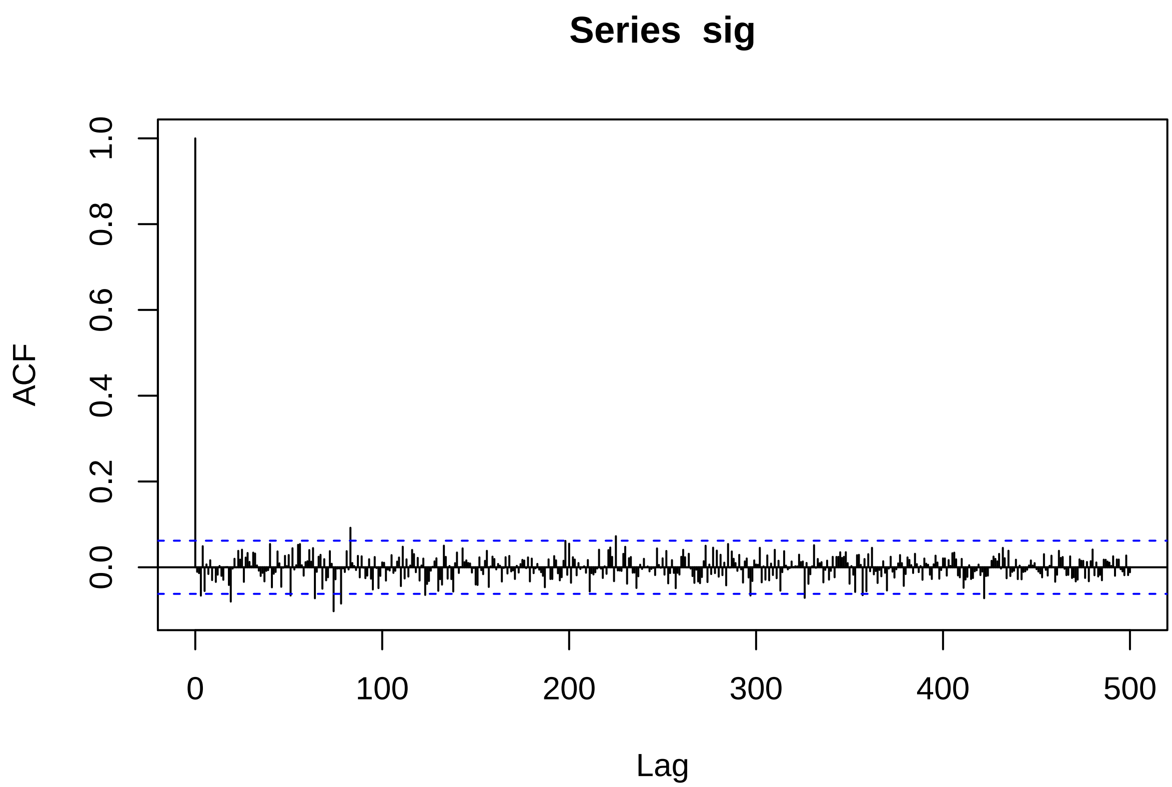
```r
#Auto-Covariance
covar <- acf(sig, lag.max = 500, type = "covariance")
```




```r
# Average Auto-Covariance over all time lags from 0->500
mean(covar$acf)
```

```
## [1] 0.0007833218
```

```r
#Auto-Correlation
auto_cor <- acf(sig, lag.max = 500, type = "correlation")
```




```r
# Average Auto-Correlation over all time lags from 0->500
mean(auto_cor$acf)
```

```
## [1] 0.0007771484
```

Part C

I calculated the min, max, and plotted the histogram using the code below.

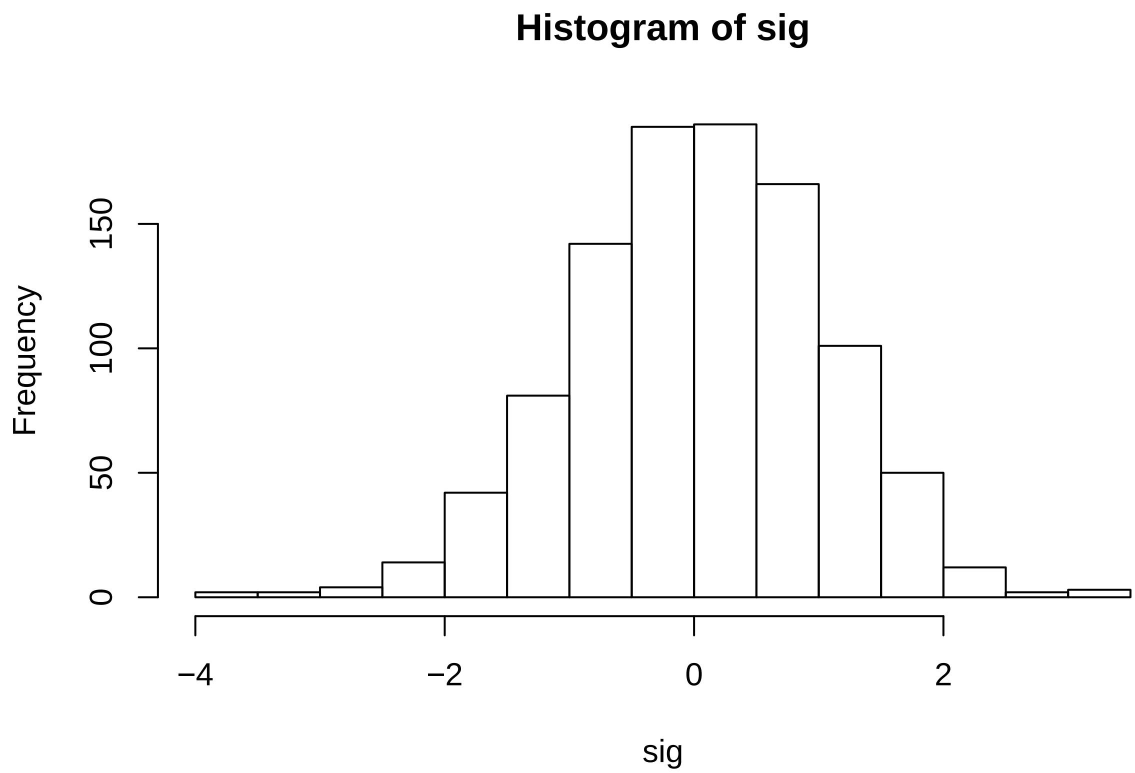
```r
min(sig)
```

```
## [1] -3.993425
```

```r
max(sig)
```

```
## [1] 3.37491
```

```r
hist(sig)
```



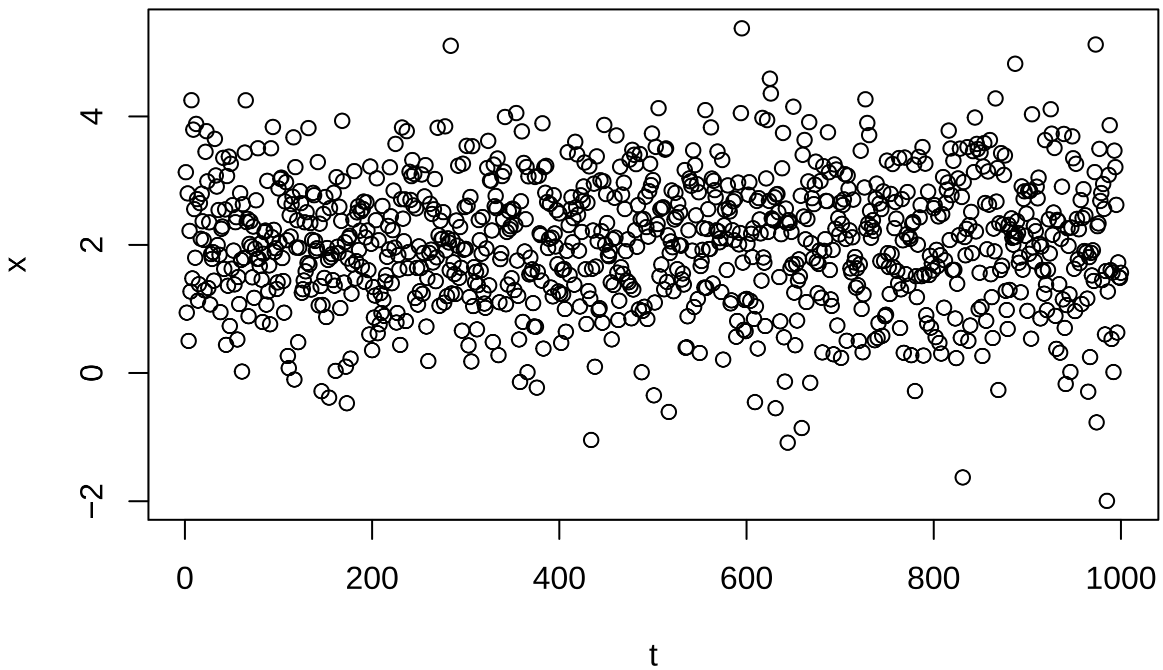
Number 3

Part A

Number 2 is now repeated.

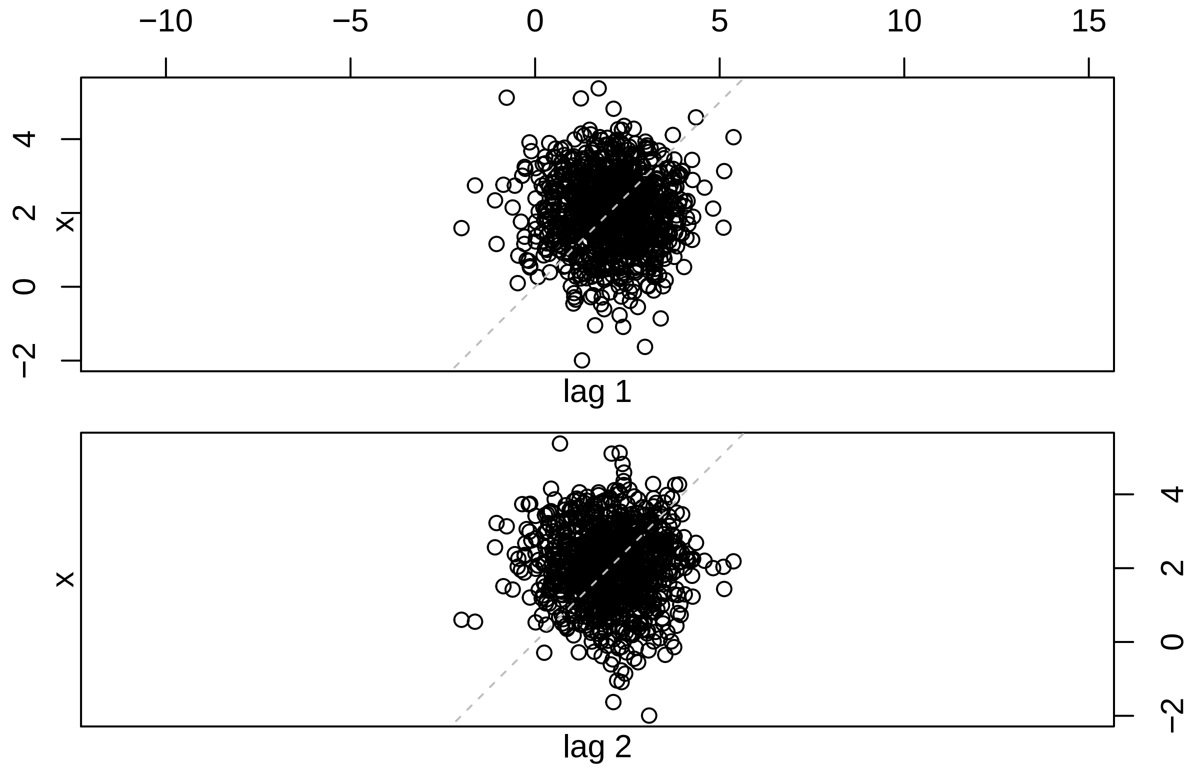
```
x <- 2.0 + sig

plot(t,x)
```



Essentially, the plot moves over two units as a result of adding the constant two for each observation of sigma.

```
lag.plot(x, lags=2)
```



The lag plots look exactly the same as previously. There appears to covariance or correlation within the time series.

The mean and variance were then calculated. These values make sense because when you add a constant value to the population mean then you should expect to see the new calculated value as the new population mean

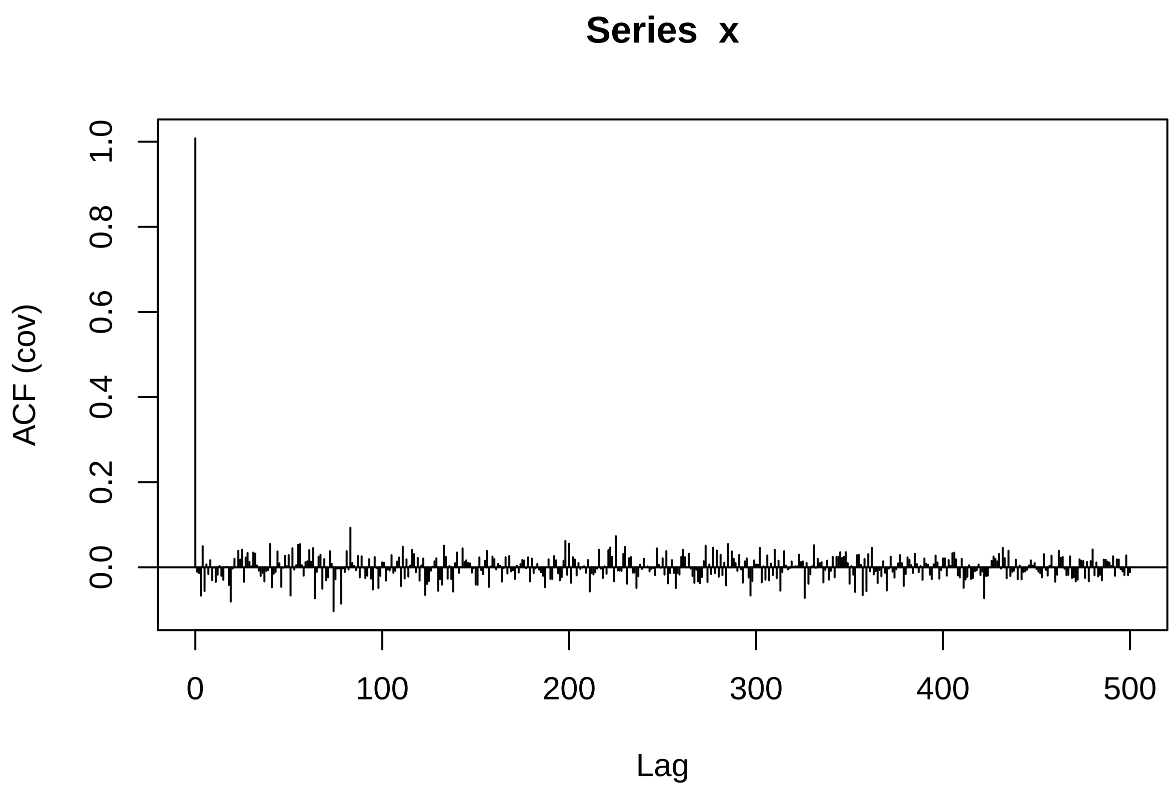
```
mean(x)
```

```
## [1] 2.039102
```

```
var(x)
```

```
## [1] 1.008953
```

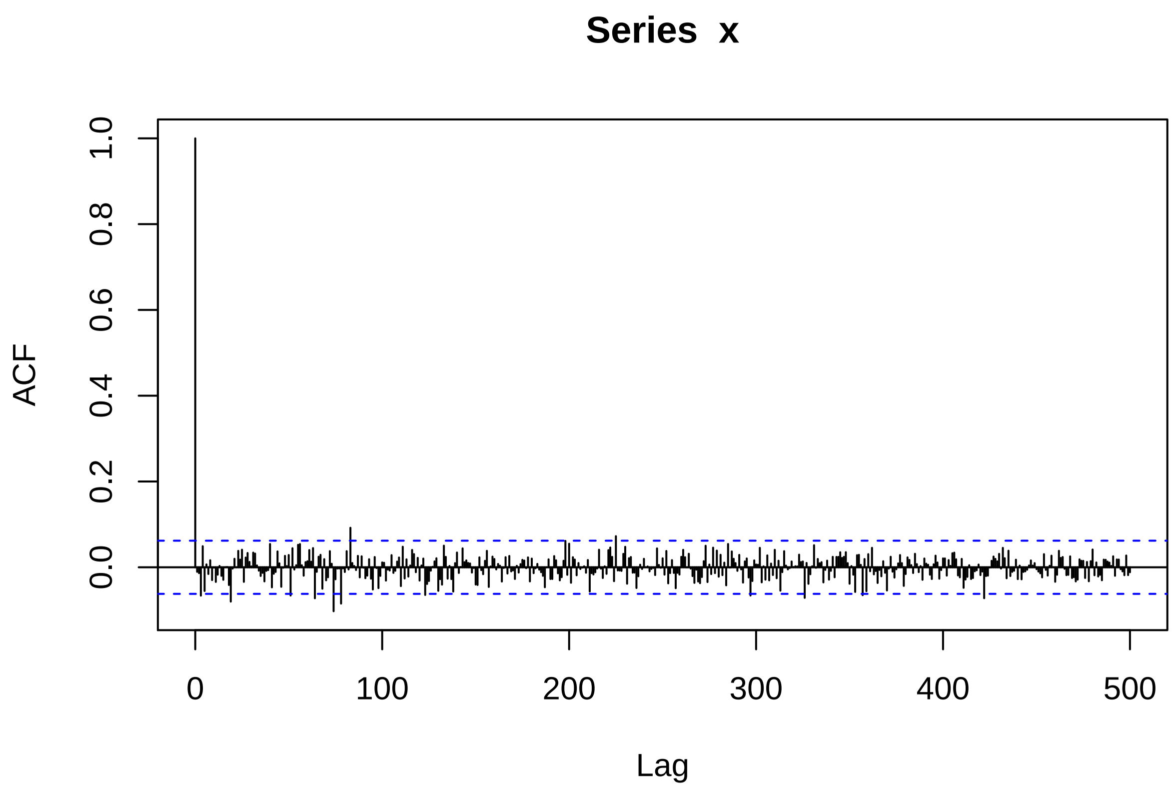
```
#Auto-Covariance
covar <- acf(x, lag.max = 500, type = "covariance")
```




```
# Average Auto-Covariance over all time lags from 0->500
mean(covar$acf)
```

```
## [1] 0.0007833218
```

```
#Auto-Correlation
auto_cor <- acf(x, lag.max = 500, type = "correlation")
```




```
# Average Auto-Correlation over all time lags from 0->500
mean(auto_cor$acf)
```

```
## [1] 0.0007771484
```

Just like the lag plots before, the auto-correlation and auto-covariance plots look almost exactly like the ones previous.

Additionally, the histogram looks exactly the same but has moved to have central tendency around the value of 2 which is in line with the population density of the time series.

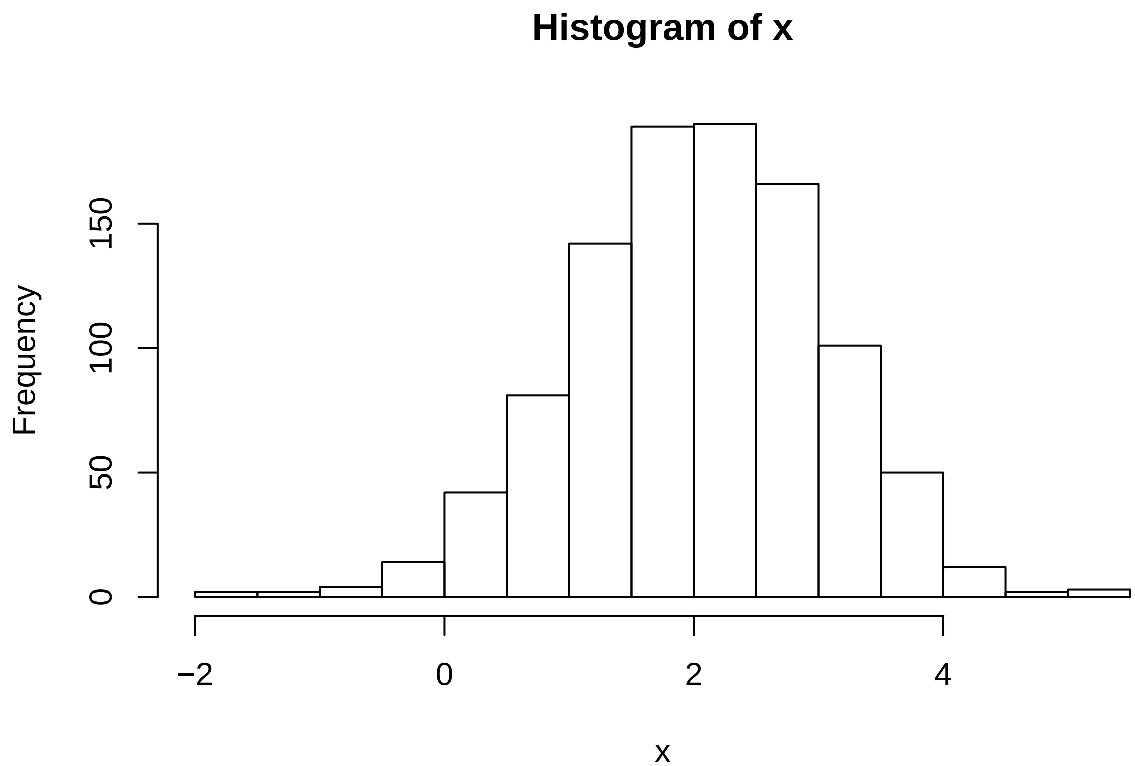
```
min(x)
```

```
## [1] -1.993425
```

```
max(x)
```

```
## [1] 5.37491
```

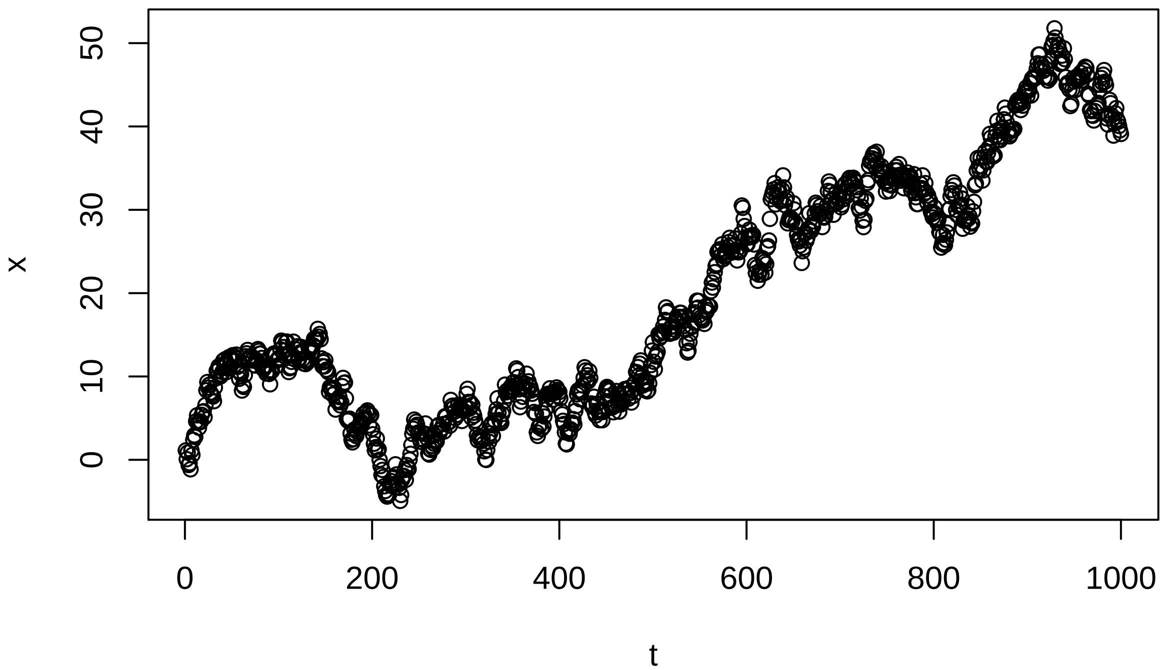
```
hist(x)
```



Part B

The next step in the homework was to repeat number but with a new time series. The time series is generated and plotted across t below. We re-define x and perform all the measures.

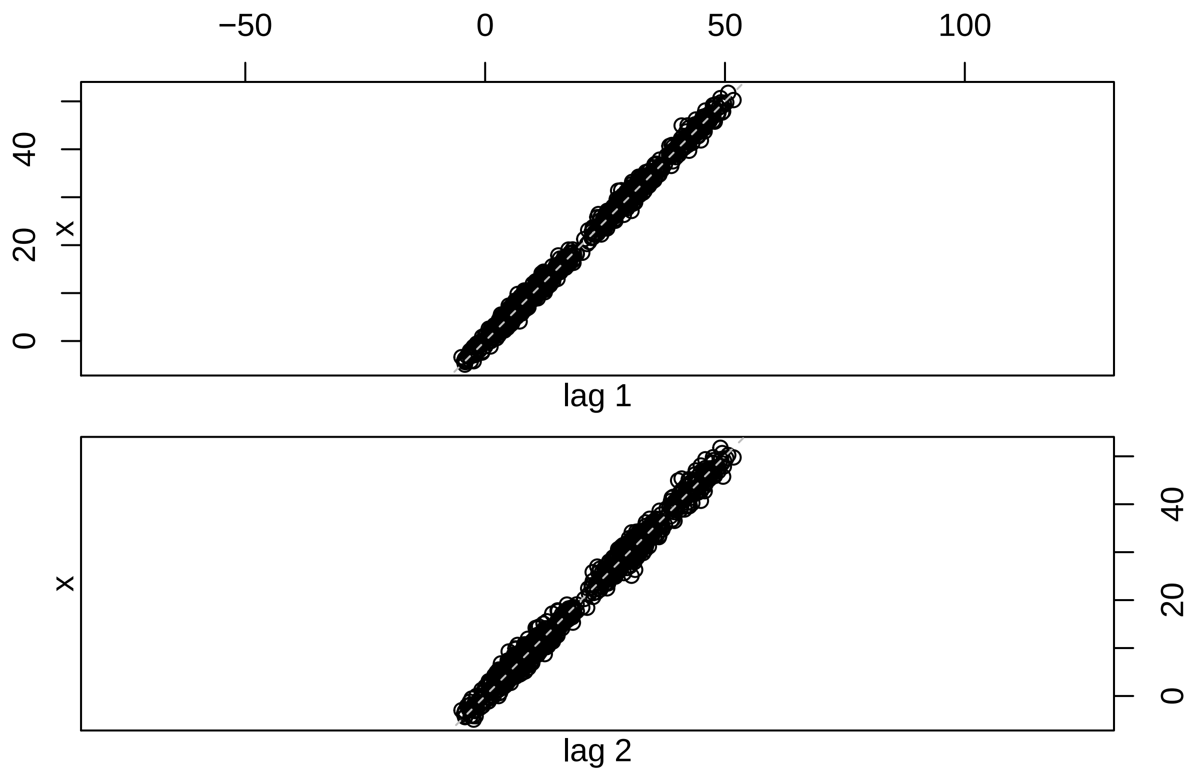
```
x <- cumsum(sig)
plot(t,x)
```



From this plot we can begin to see some trending in the time series. We can see that the function is a Gaussian Random Walk.

Now we plot the lags of the time series against the time series. And come up with the following plots.

```
lag.plot(x, lags=2)
```



From the lag plots we can see that there is trending going on in our time series. Which we would expect to see this because each new value of X is based on the previous values of X. There is going to be a lot of auto-corvariance and auto-correlation in our time series as a result.

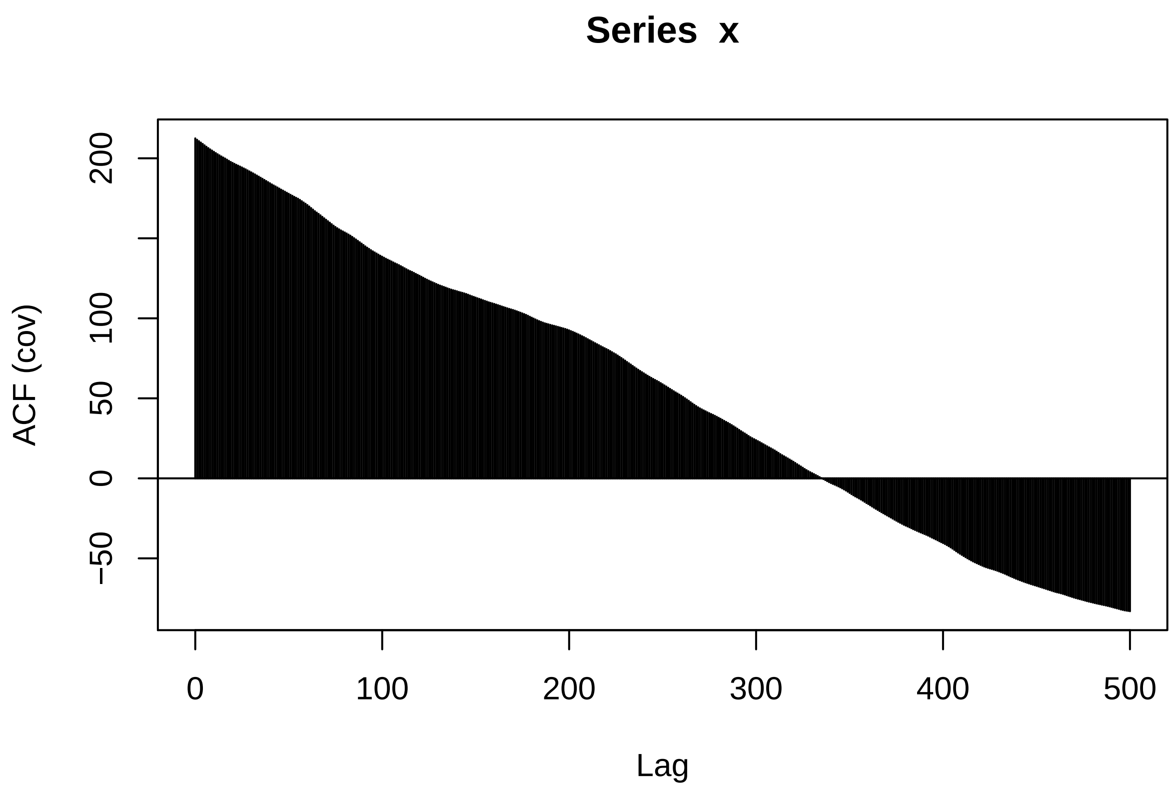
```r
mean(x)
```

```
## [1] 19.53906
```

```r
var(x)
```

```
## [1] 212.6647
```

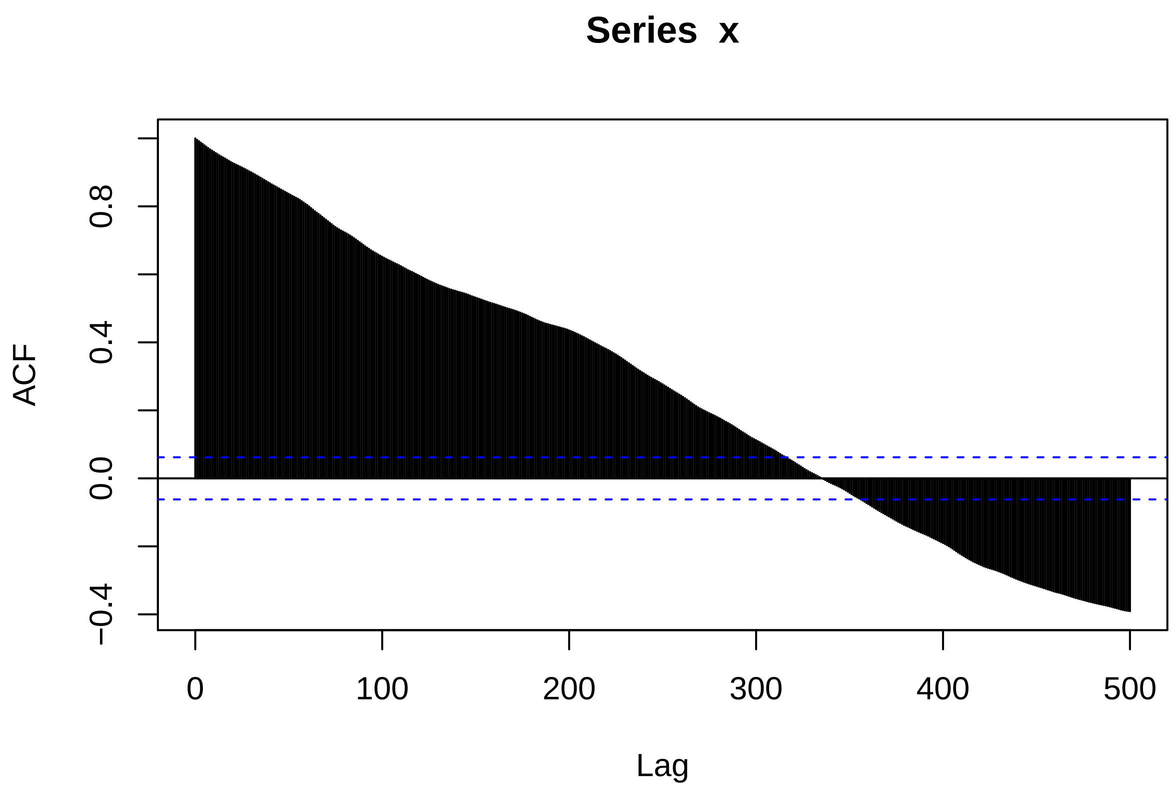
```r
#Auto-Covariance
covar <- acf(x, lag.max = 500, type = "covariance")
```




```
# Average Auto-Covariance over all time lags from 0->500
mean(covar$acf)
```

```
## [1] 54.75417
```

```
#Auto-Correlation
auto_cor <- acf(x, lag.max = 500, type = "correlation")
```




```
# Average Auto-Correlation over all time lags from 0->500
mean(auto_cor$acf)
```

```
## [1] 0.2577249
```

These plots look far different from our previous two time series. We can see how the auto-covariance and auto-correlation changes across different time lags in our data.

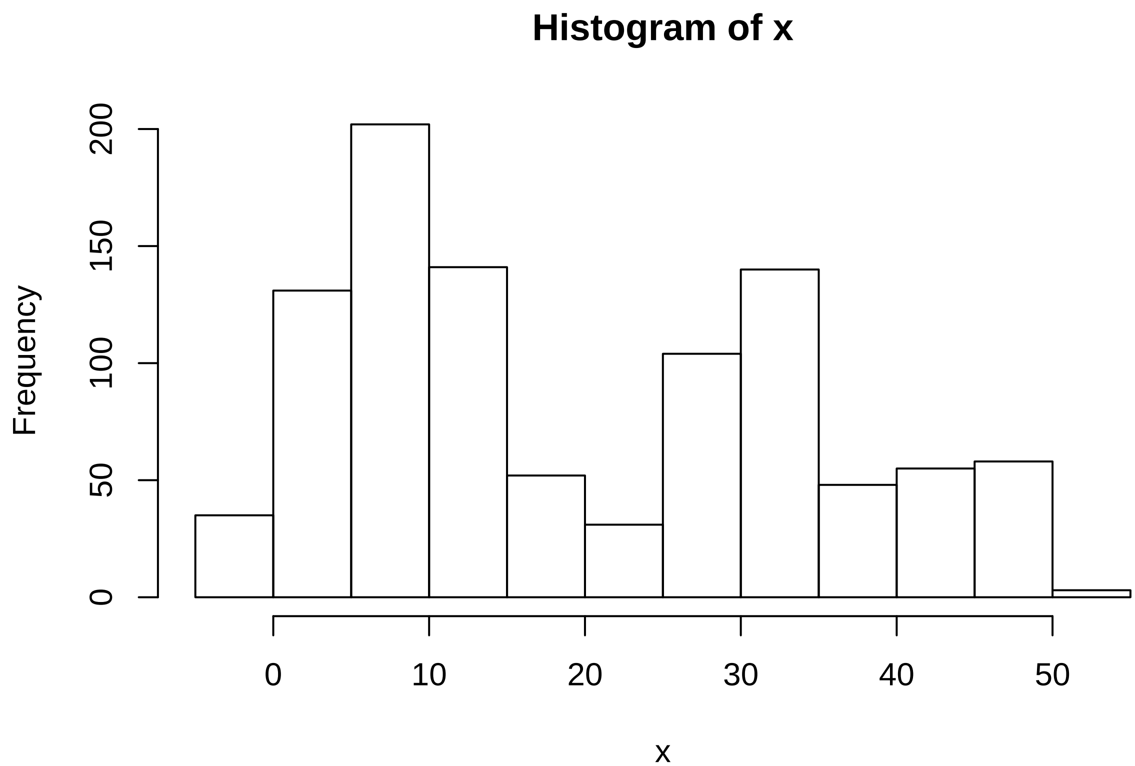
```
min(x)
```

```
## [1] -4.927356
```

```
max(x)
```

```
## [1] 51.777
```

```
hist(x)
```



Our data in this case is appears to no longer be normally distributed as can be seen from the histogram.

Number 4

1. Let xt be a weakly stationary process with $\gamma(k)$ as its auto-correlation function
2. Let $a = (a_1, a_2, ...a_n)^T$ contained R^n
3. Let $z_i = x_i - E(x_t), 1 \leq i \leq n$, and, $z = (z_1, z_2, ...z_n)^T$ contained R^n
4. Then: $0 \leq Var(a^t z) = E(a^t z)^2 = E[a^t z z^t a] = a^T \Gamma a = \sum a_i \Gamma(i,j) a_j$, where, $\Gamma = \Gamma(i,j) = E(z_i z_j)$ is in the n x n covariance matrix.
5. Since $\Gamma(i,j) = \gamma(|i-j|)$, then the auto-covariance function is non-negative definite.