Partial autocorrelation

Partial autocorrelation for exchange rate between GBP and NZD

Read in the quarterly GBP to NZD exchange rate, and save it as a vector called `exchange_data`:

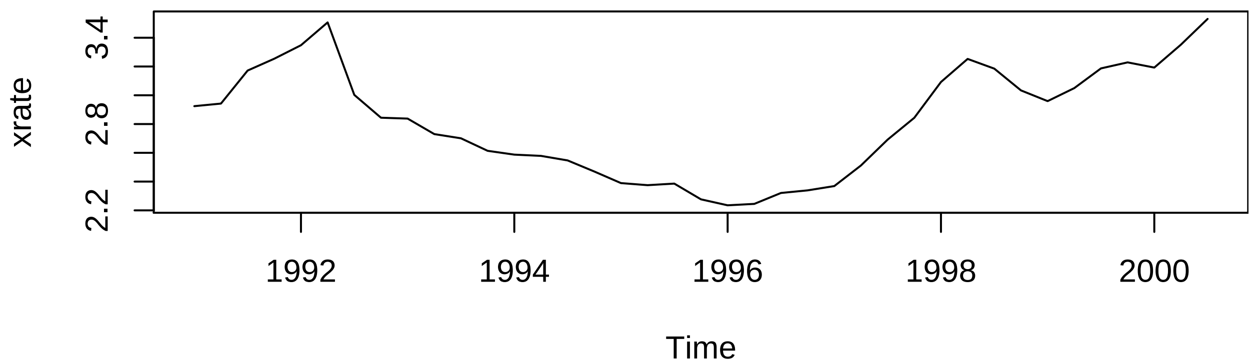
```
www <- "http://asta.math.aau.dk/eng/static/datasets?file=pounds_nz.dat"
exchange_data <- read.table(www, header = TRUE)
```

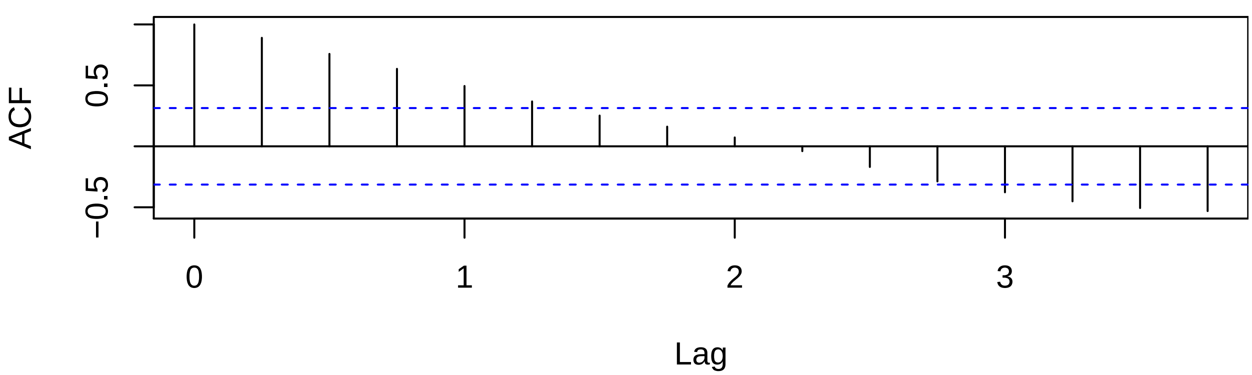
Now convert it to a time series object (`ts`) with the correct starting date (First quarter 1991) and frequency and call it `exchange`:

```
exchange <- ts(exchange_data, start = 1991, freq = 4)
```

- Plot the series and its autocorrelation function.

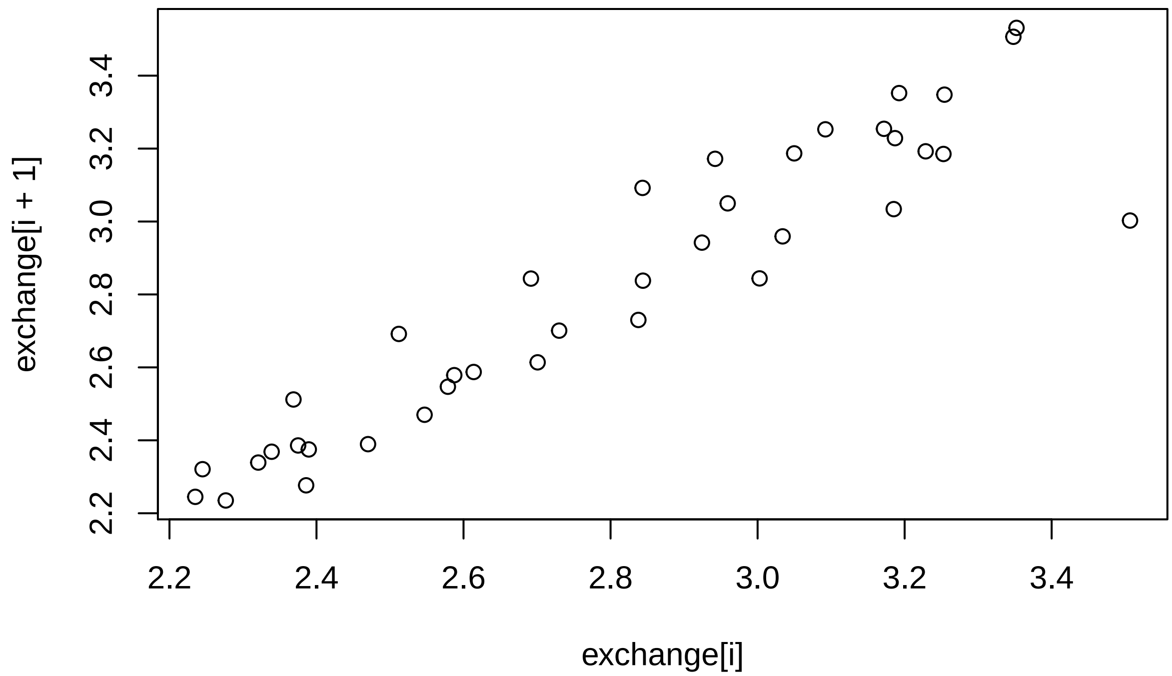
```
par(mfrow = c(2,1), mar = c(5,4,1,0))
plot(exchange)
acf(exchange)
```





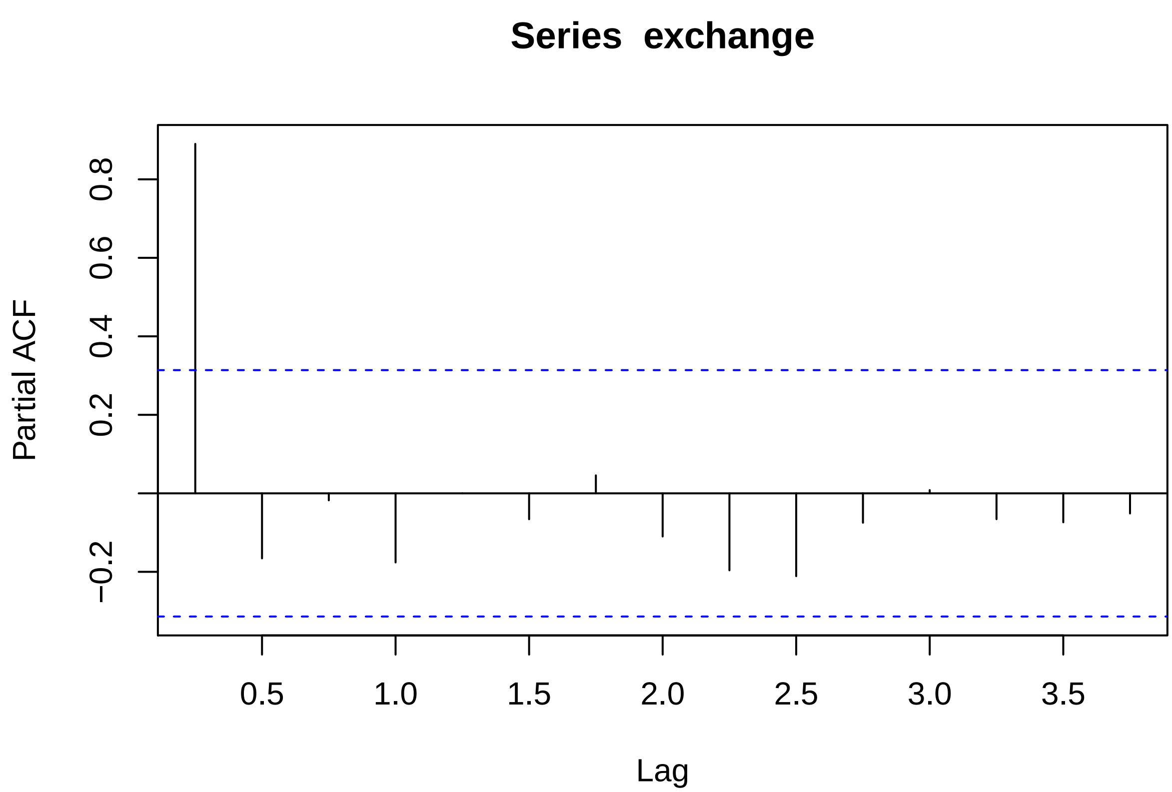
- Make a scatterplot of the series and its lag 1 values.

```r
i <- 1:(length(exchange) - 1)
plot(exchange[i], exchange[i+1])
```



- Plot the partial autocorrelation function.

```r
pacf(exchange)
```

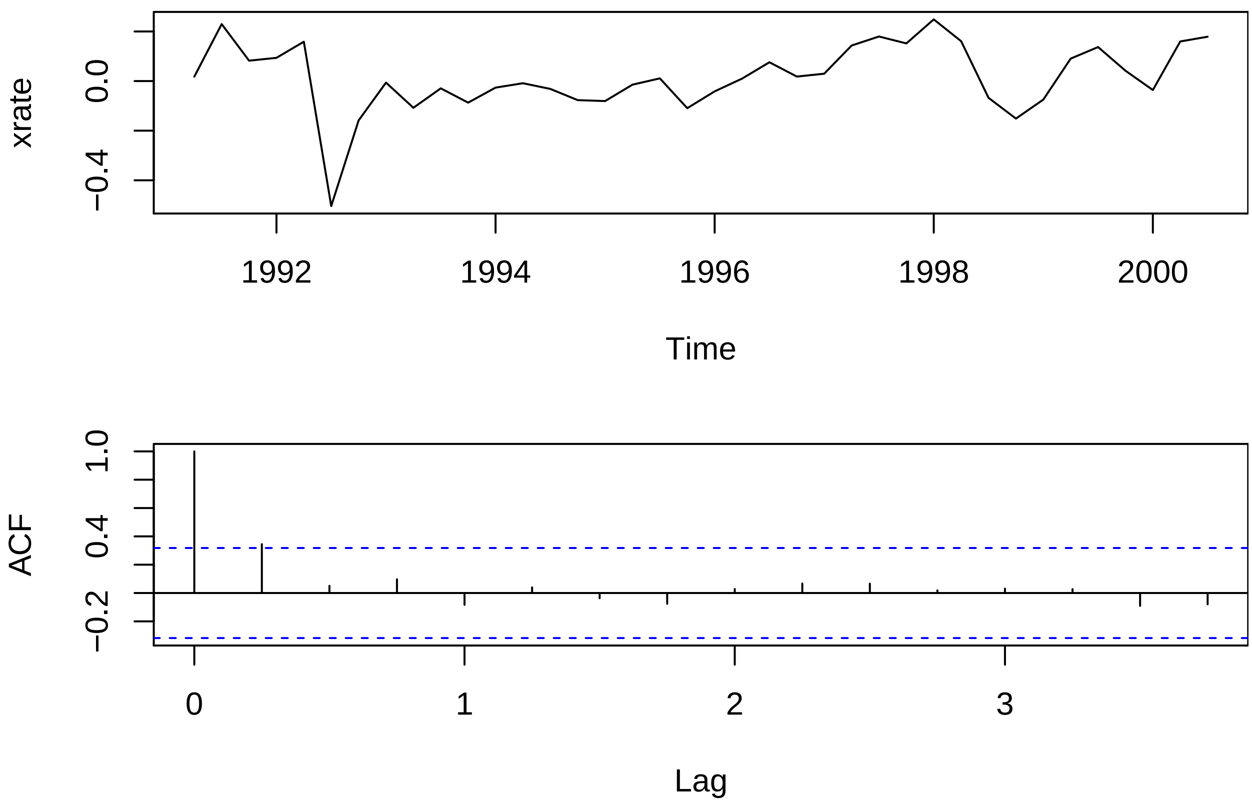


Consider the series of first order differences $y_t = x_{t+1} - x_t$ for $t = 1, \ldots, 38$:

```
dexchange <- diff(exchange)
```

- Plot the series of differences and its autocorrelation function and comment on it.

```
par(mfrow = c(2,1), mar = c(5,4,1,0))
plot(dexchange)
acf(dexchange)
```



Random walk

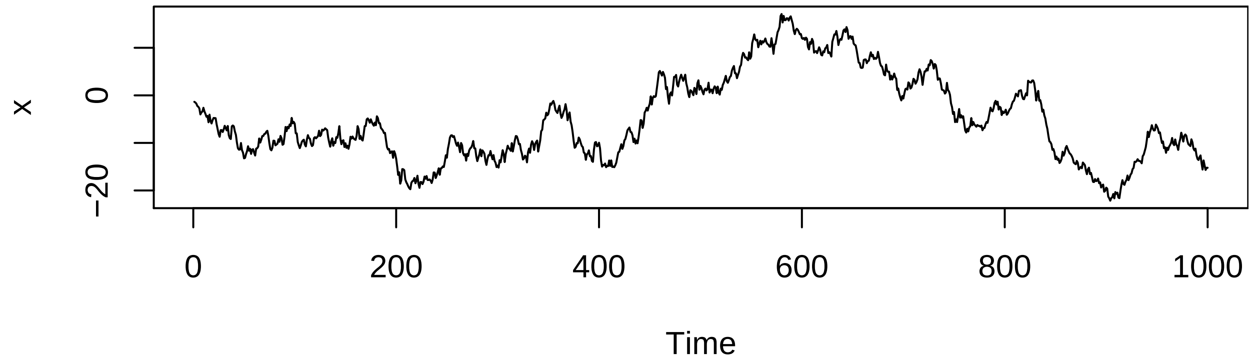
The method of removing trend by differencing can be really effective as seen in the following where we redo the exercise above for an artificial dataset `x`:

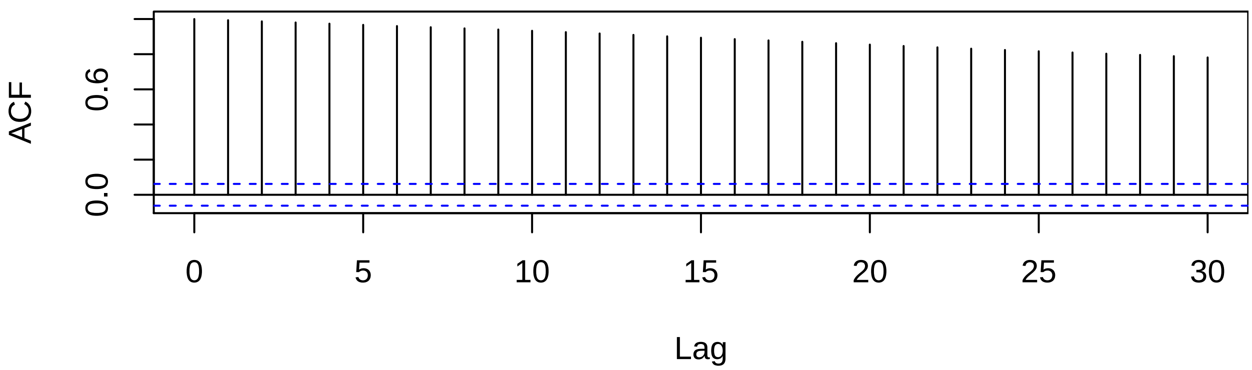
```
x <- cumsum(rnorm(1000))
```

This is called a random walk and we will discuss it more in the next lecture.

- Plot the series and its autocorrelation function.

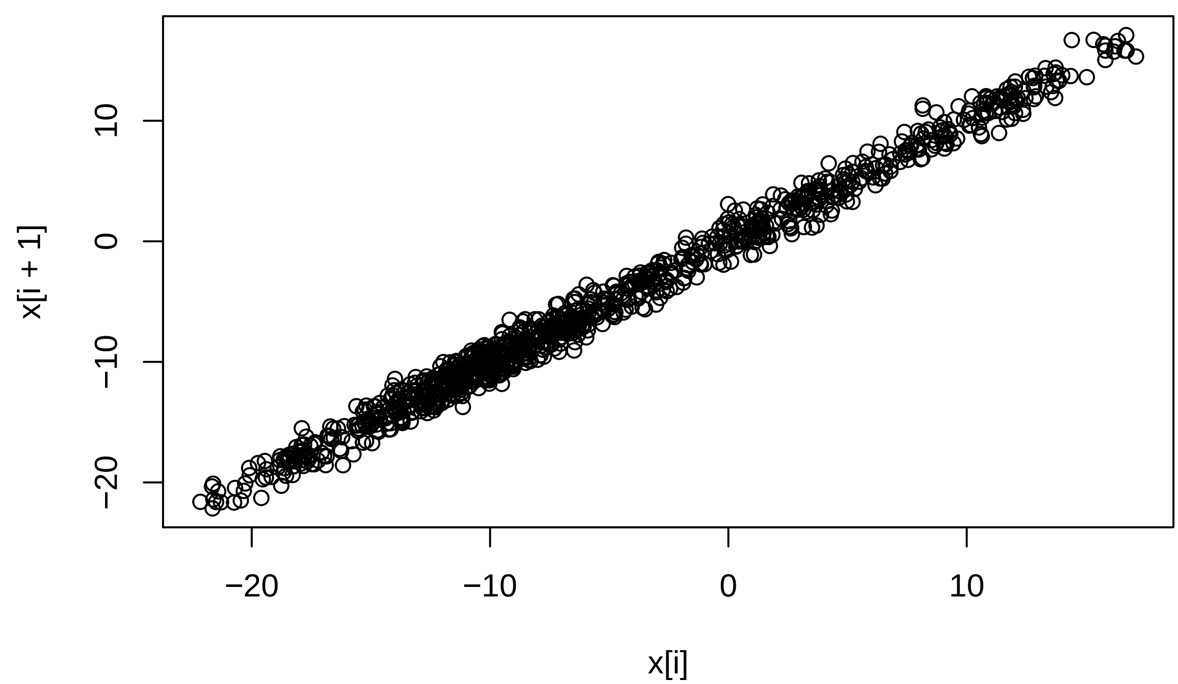
```
par(mfrow = c(2,1), mar = c(5,4,1,0))
ts.plot(x)
acf(x)
```





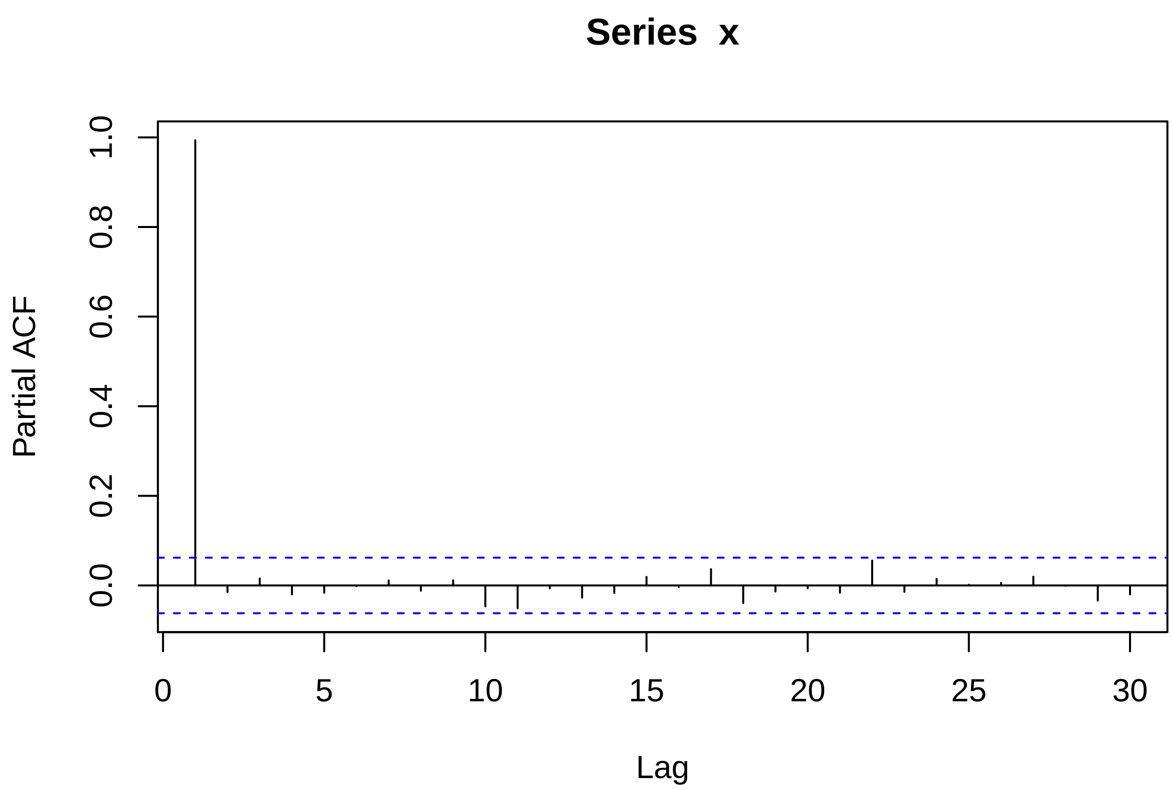
- Make a scatterplot of the series and its lag 1 values.

```
i <- 1:(length(x) - 1)
plot(x[i], x[i+1])
```



- Plot the partial autocorrelation function.

```r
pacf(x)
```

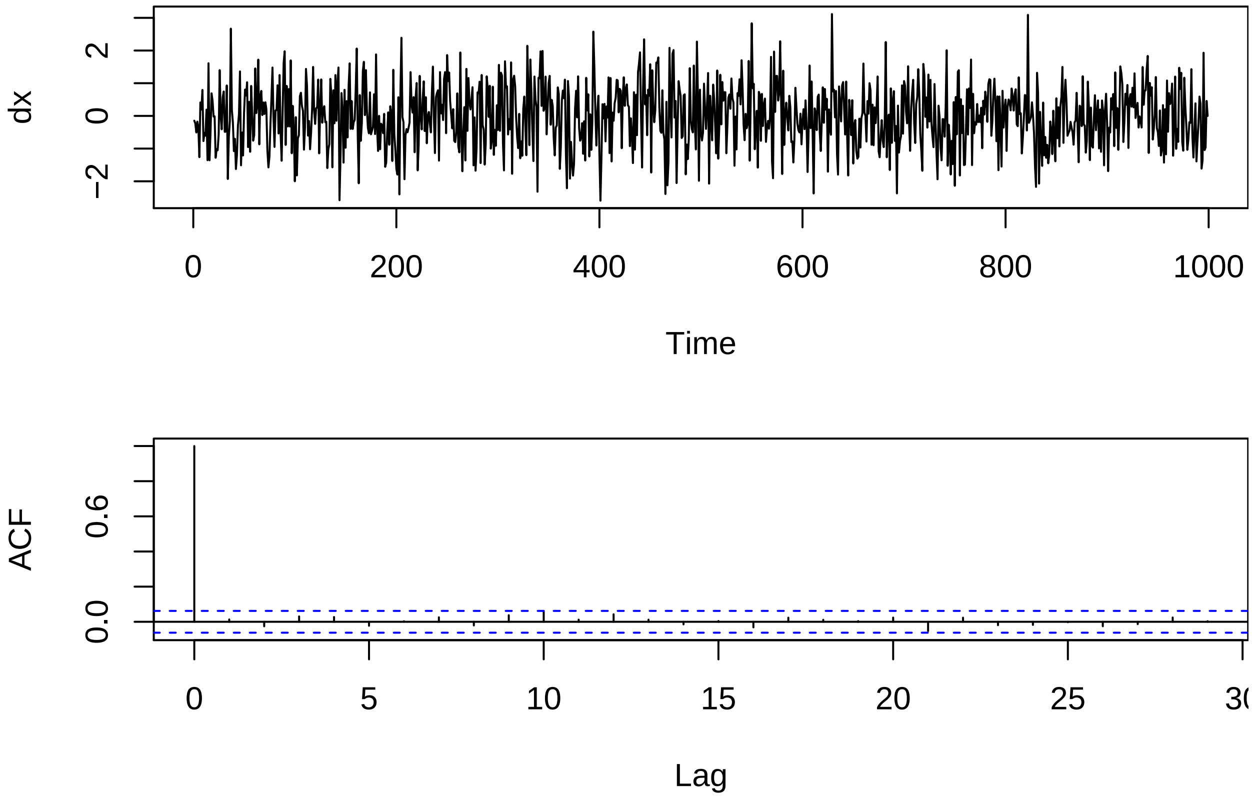


Consider the series of first order differences $y_t = x_{t+1} - x_t$ for $t = 1, \ldots, 999$:

```r
dx <- diff(x)
```

- Plot the series and its autocorrelation function.

```r
par(mfrow = c(2,1), mar = c(5,4,1,0))
ts.plot(dx)
acf(dx)
```


dx
2
0
−2
0
200
400
600
800
1000
Time
ACF
0.6
0.0
0
5
10
15
20
25
Lag