

MATH 545: INTRODUCTION TO TIME SERIES ANALYSIS

WINTER 2025

NOTATION AND SIMULATION FOR TIME SERIES MODELS

We can define time series models using simple constructions. Consider first the sequence of random variables $\{Z_t\}_{t \geq 1}$, where $Z_t \sim \text{Normal}(0, 1)$ with the random variables (mutually) independent. Using these variables, we may define the simple time series $\{X_t\}_{t \geq 1}$. For example, suppose

$$X_t = Z_t - 5Z_{t-1} + 4Z_{t-2}.$$

We define the backshift operator, B , essentially, as the operator that shifts time back by one unit; in particular, we have

$$BZ_t = Z_{t-1}.$$

This operator exhibits standard arithmetic and algebraic properties: for example

- $B^2 Z_t = B(BZ_t) = B(Z_{t-1}) = Z_{t-2}$
- $(c_1 + c_2 B)Z_t = c_1 Z_t + c_2 Z_{t-1}$
- $(1 - B)^2 Z_t = (1 - 2B + B^2)Z_t = Z_t - 2Z_{t-1} + Z_{t-2}$

and so on. Therefore we may write

$$X_t = Z_t - 5Z_{t-1} + 4Z_{t-2} = (1 - 5B + 4B^2) Z_t.$$

For simulation, we generate Z_{-1} and Z_{-2} from the same $\text{Normal}(0, 1)$ distribution.

```
set.seed(2101)           #Set the random number generator seed value
n<-200                   #Set the sample size
Z<-rnorm(n)              #Generate Z[1],...Z[n]
Z.1<-rnorm(1)            #Generate Z[-1]
Z.2<-rnorm(1)            #Generate Z[-2]
Z.extra<-c(Z.2,Z.1,Z)    #Concatenate
X<-Z.extra[3:(n+2)]-5*Z.extra[2:(n+1)]+4*Z.extra[1:n]  #Define the time series X
```

An alternate way to do this is to use the filter function:

```
X.f<-filter(Z.extra,c(1,-5,4))[2:(n+1)]
sum(X.f-X)                #X and X.f are identical
+ [1] 0
```

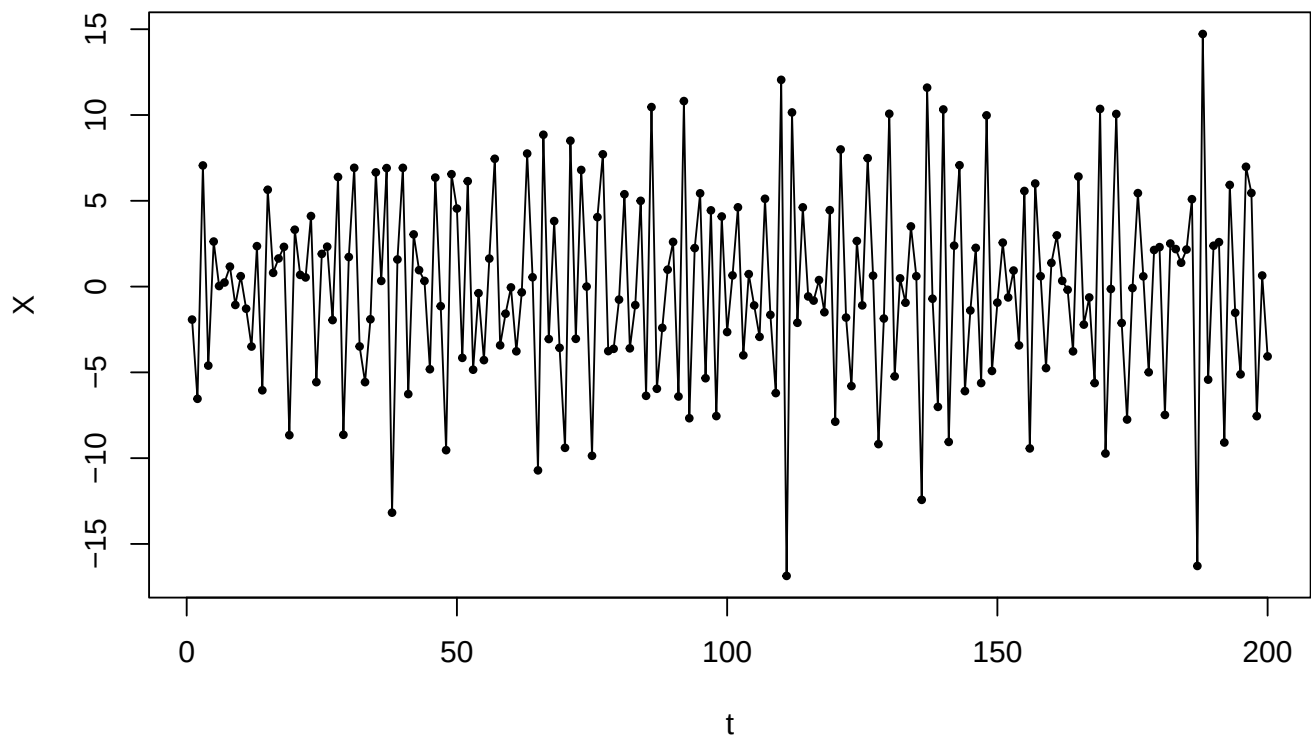
We can also create a special time series object in R using the `ts` function: for example, if we wanted the observations to represent quarterly data, starting from 2010, 3rd quarter, we would write

```
X.t<-ts(X, frequency = 4, start = c(2010, 3))
```

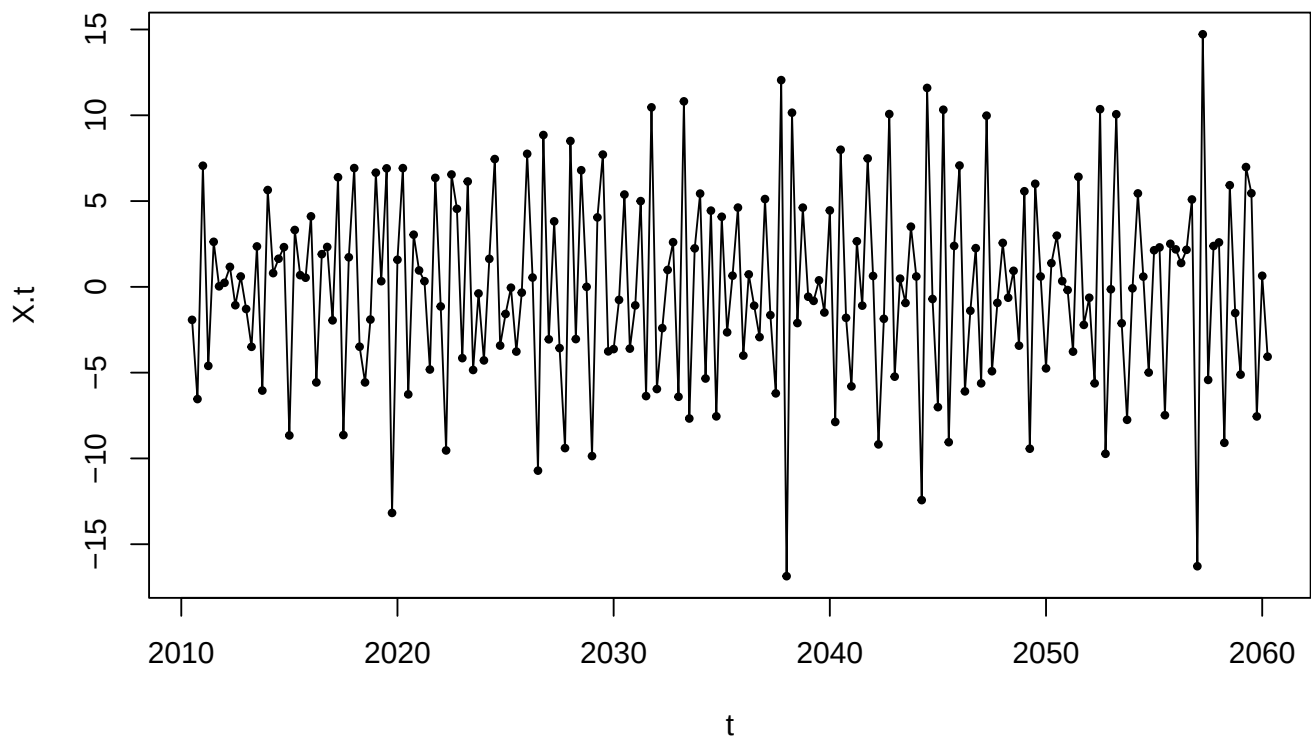
With 200 observations, this amounts to 40 years worth of data.

The plotting outcomes are slightly different:

```
par(mar=c(4,4,1,0))      #Set up the plotting margins
plot(1:n,X,xlab='t',type='l') #Construct a line plot
points(1:n,X,pch=19,cex=0.5) #Add the points on top of the line
```



```
plot(X.t,xlab='t',type='l')      #Construct a line plot
points(X.t,pch=19,cex=0.5)      #Add the points on top of the line
```



Note that the time axis is now plotted in years.

We can use elementary functions in R to summarize the simulated data:

```
mean(X)                                #Mean
+ [1] -0.03171071

var(X)                                  #Variance
+ [1] 31.6681
```

In time series, we are interested in covariance and correlation at different *lags*, $h = 0, 1, 2, \dots$; recall that for a stationary process

$$\gamma_X(h) = \text{Cov}[X_t, X_{t+h}] \quad h = 0, \pm 1, \pm 2, \dots$$

with $\gamma_X(-h) = \gamma_X(h)$, and

$$\rho_X(h) = \text{Corr}[X_t, X_{t+h}] = \frac{\gamma_X(h)}{\gamma_X(0)} \quad h = 0, \pm 1, \pm 2, \dots$$

define the *autocovariance function* (ACVF) and *autocorrelation function* (ACF). We can compute both using the `acf` function:

```
acf(X, type='covariance', plot=F)      #Covariance

+
+ Autocovariances of series 'X', by lag
+
+      0      1      2      3      4      5      6      7      8      9
+ 31.510 -16.947 -1.359  5.075 -5.954  5.690 -1.890 -1.382  1.978 -1.124
+      10     11     12     13     14     15     16     17     18     19
+ -0.464  2.562 -3.517  2.018  0.339  0.433 -2.690  1.463  0.830 -1.712
+      20     21     22     23
+  0.776  0.664  0.195 -2.416

acf(X, type='correlation', plot=F)     #Correlation

+
+ Autocorrelations of series 'X', by lag
+
+      0      1      2      3      4      5      6      7      8      9     10
+ 1.000 -0.538 -0.043  0.161 -0.189  0.181 -0.060 -0.044  0.063 -0.036 -0.015
+      11     12     13     14     15     16     17     18     19     20     21
+ 0.081 -0.112  0.064  0.011  0.014 -0.085  0.046  0.026 -0.054  0.025  0.021
+      22     23
+ 0.006 -0.077
```

Notice that the variance calculation using `acf` at lag $h = 0$ gives a different answer to that obtained using `var`:

```
(v0<-var(X))                                #Variance using var
+ [1] 31.6681

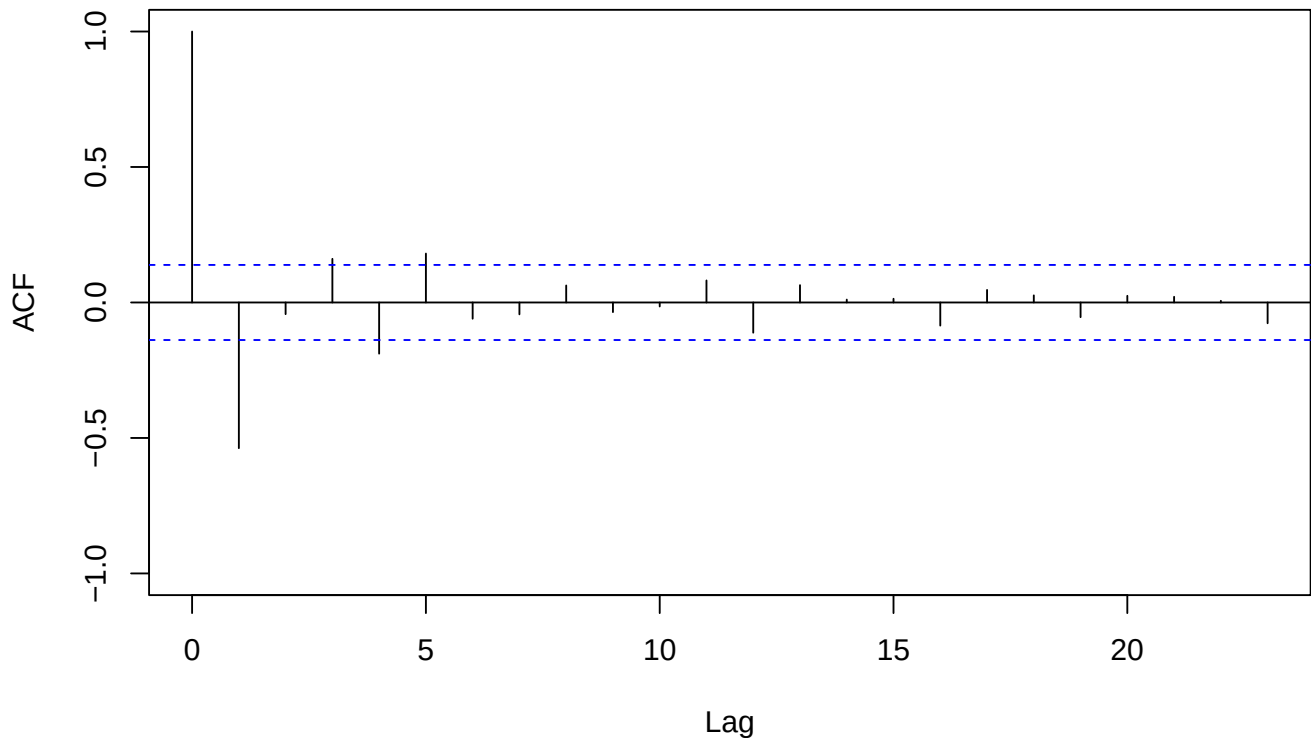
(v1<-acf(X, type='covariance', plot=F)$acf[1])  #Variance using acf
+ [1] 31.50976
```

This is because the `acf` uses the formula

$$\hat{\gamma}_X(h) = \frac{1}{n-h} \sum_{t=1}^{n-h} (x_t - \bar{x})(x_{t+h} - \bar{x}) \quad (1)$$

that is, dividing by n not $n - 1$ for the variance calculation (with $h = 0$). We may plot the ACVF or ACF using the `'plot'` argument to `acf`

```
par(mar=c(4,4,1,0)) #Set up the plotting margins
acf(X,type='correlation',plot=T,ylim=range(-1,1)) #Autocorrelation plot
```



The dashed horizontal lines either side of zero give an indication of statistical significance; if the ACF estimate $\hat{\rho}_X(h)$ exceeds the horizontal dashed line, then the correlation at that lag may be considered significantly different from zero.

Note: Recall that a random walk time series process is formed by setting

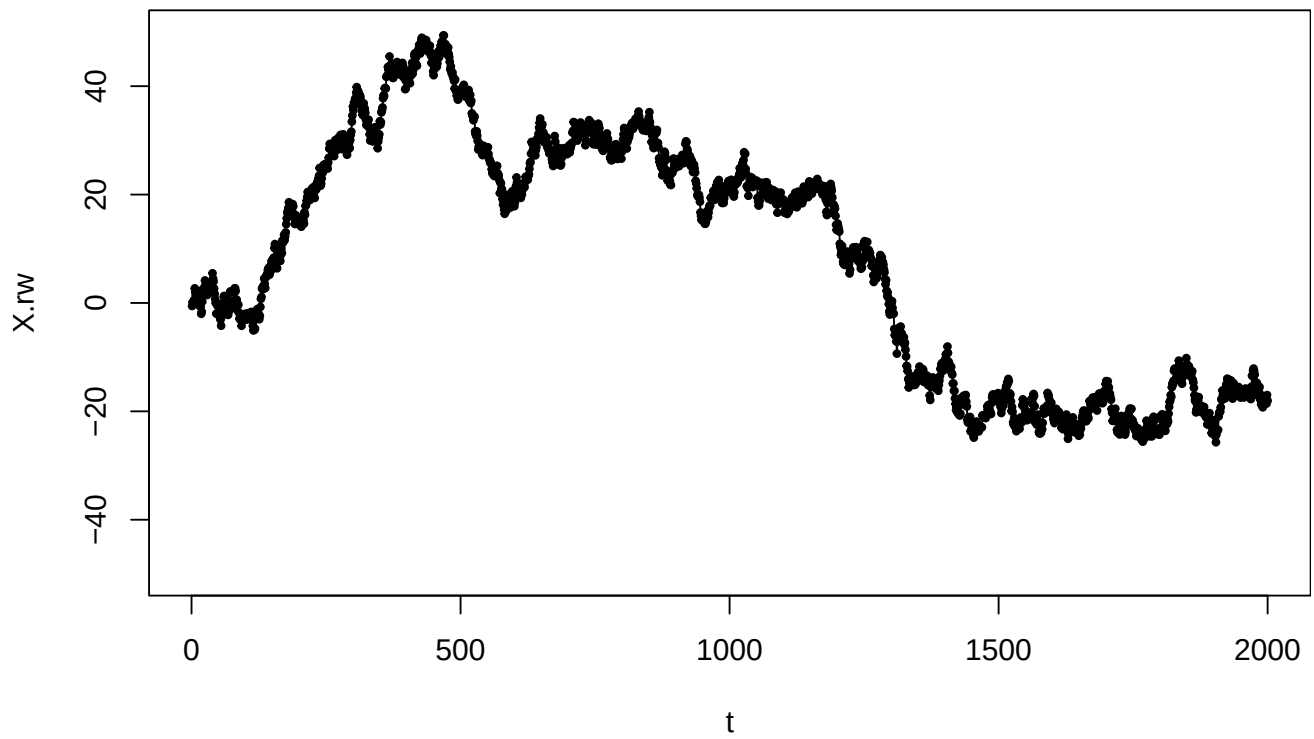
$$X_t = \sum_{i=1}^t Z_i;$$

with $\{Z_t\} \sim \text{WN}(0, \sigma_Z^2)$. Recall that for such a process

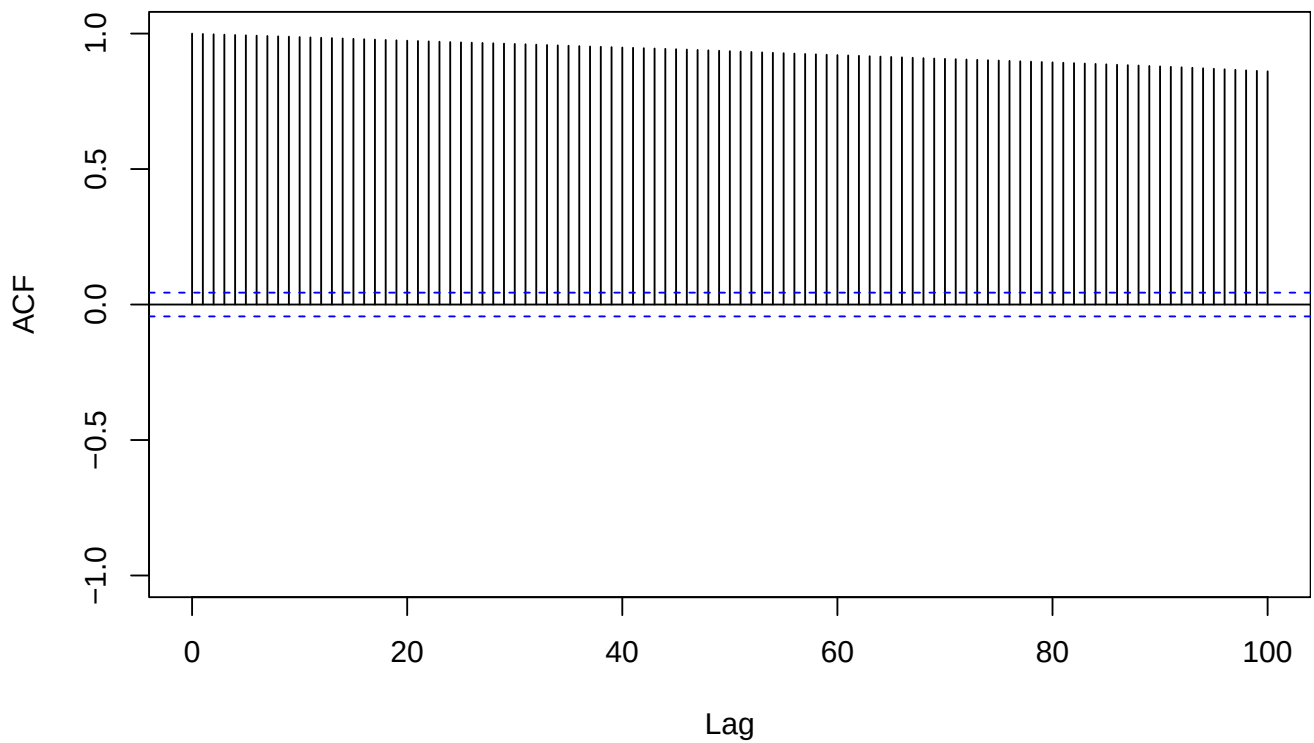
$$\rho_X(t, h) = \text{Corr}[X_t, X_{t+h}] = \frac{t}{\sqrt{t(t+h)}}.$$

This process is non-stationary and because of this, the estimates derived using equation (1) – which is derived assuming stationarity – are inappropriate. Because of this, the random walk has a characteristic sample ACF pattern with very slow decay.

```
n<-2000
Z<-rnorm(n)
X.rw<-cumsum(Z)
par(mar=c(4,4,1,0)) #Form the cumulative sum of the Zs
plot(1:n,X.rw,xlab='t',type='l',ylim=range(-50,50)) #Set up the plotting margins
points(1:n,X.rw,pch=19,cex=0.5) #Construct a line plot
#Add the points on top of the line
```



```
acf(X.rw,type='correlation',plot=T,ylim=range(-1,1),lag.max=100) #Autocorrelation plot
```



An ACF plot such as this is strongly indicative of non-stationarity.