Time Series Simulation

Xinyue Lu

2020/2/24

```
library(dtw)
```

```
## Loading required package: proxy
```

```
##
## Attaching package: 'proxy'
```

```
## The following objects are masked from 'package:stats':
##
##     as.dist, dist
```

```
## The following object is masked from 'package:base':
##
##     as.matrix
```

```
## Loaded dtw v1.21-3. See ?dtw for help, citation("dtw") for use in publication.
```

```
library(TSA)
```

```
## Warning: package 'TSA' was built under R version 3.6.2
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##     acf, arima
```

```
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(astsa)
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.6.2
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Registered S3 methods overwritten by 'forecast':
##   method       from
##   fitted.Arima TSA
##   plot.Arima   TSA
```

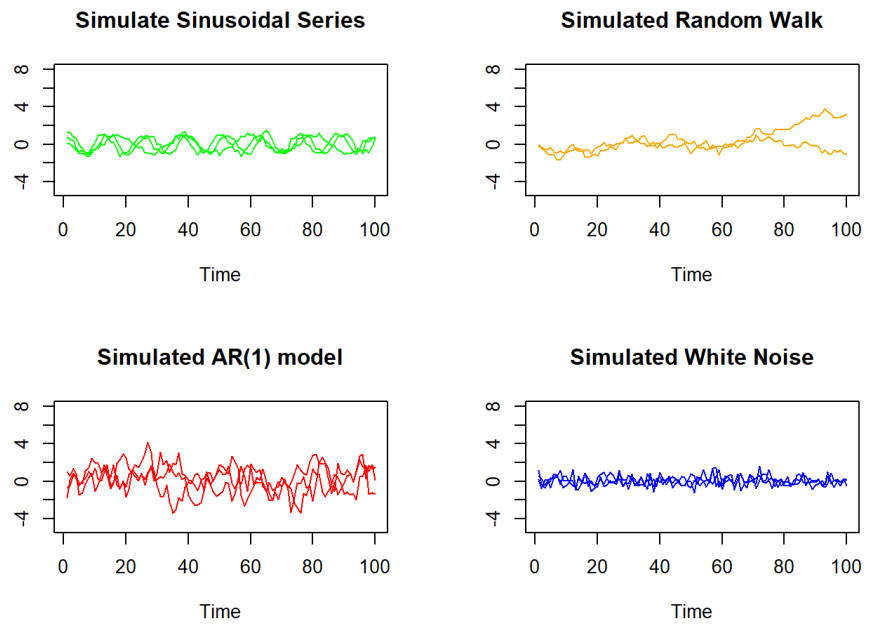
```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:astsa':
##
##     gas
```

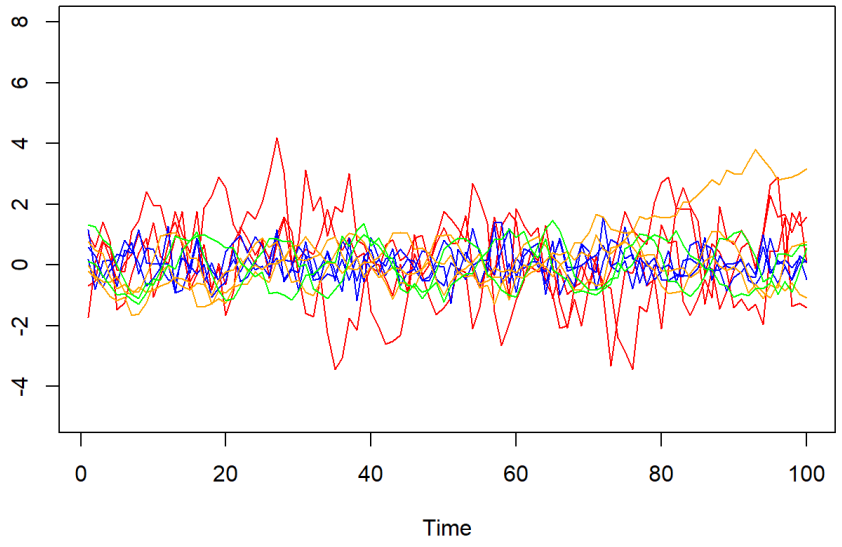
```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 3.6.2
```

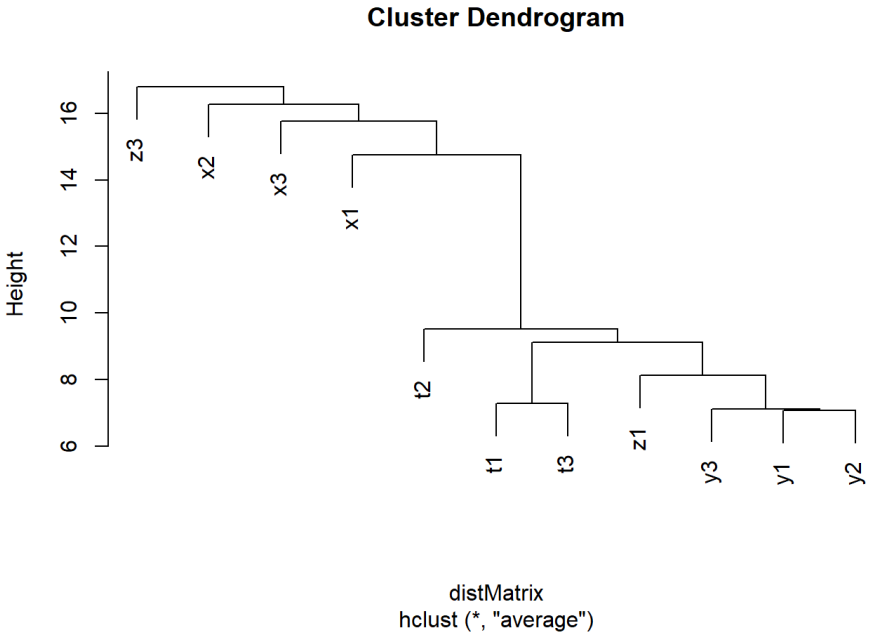
```
# randomly sampled 100 cases from each class
set.seed(123)
n=100
x=c(1:100)

#simulate sinusoidal series
p1=rnorm(n,mean=0,sd=0.2)
p2=rnorm(n,mean=0,sd=0.2)
t2=cos(0.3*x+1)+p2
t1=cos(0.5*x)+p1
p3=rnorm(n,mean=0,sd=0.2)
t3=cos(0.5*x-1)+p3

par(mfrow=c(2,2))
plot(x,t1,type='l',xlab="Time",ylab="",ylim=c(-5,8),col="green",main = "Simulate Sinusoidal Series")
lines(x,t2,col="green")
lines(x,t3,col="green")

##simulate random walk
z1=cumsum(rnorm(n,mean=0,sd=0.3))
z3=cumsum(rnorm(n,mean=0,sd=0.3))

plot(x,z1,type='l',xlab="Time",ylab="",ylim=c(-5,8),col="orange",main = "Simulated Random Walk")
lines(x,z3,col="orange")

#simulate AR(1) model
x1 = arima.sim(model=list(order=c(1,0,0), ar=c(0.7)),n,sd=1)+0.5
x2 = arima.sim(model=list(order=c(1,0,0), ar=c(0.7)),n,sd=1)+0.5
x3 = arima.sim(model=list(order=c(1,0,0), ar=c(0.7)),n,sd=1)

plot(x,x1,type='l',xlab="Time",ylab="",ylim=c(-5,8),col="red",main = "Simulated AR(1) model")
lines(x,x1,col="red")
lines(x,x2,col="red")
lines(x,x3,col="red")

#simulate white noise
y1 = arima.sim(model=list(order=c(0,0,0)),n,sd=0.5)
y2 = arima.sim(model=list(order=c(0,0,0)),n,sd=0.5)
y3 = arima.sim(model=list(order=c(0,0,0)),n,sd=0.5)

plot(x,y1,type='l',xlab="Time",ylab="",ylim=c(-5,8),col="blue",main = "Simulated White Noise")
lines(x,y2,col="blue")
lines(x,y3,col="blue")
```



```
#plot all the simulate data
par(mfrow=c(1,1))
plot(x,x1,type='l',xlab="Time",ylab="",ylim=c(-5,8),col="red")
lines(x,x1,col="red")
lines(x,x2,col="red")
lines(x,x3,col="red")
lines(x,y1,col="blue")
lines(x,y2,col="blue")
lines(x,y3,col="blue")
lines(x,z1,col="orange")
lines(x,z3,col="orange")
lines(x,t1,col="orange")
lines(x,t2,col="green")
lines(x,t3,col="green")
```




```
m=rbind(x1,x2,x3,y1,y2,y3,z1,z3,t1,t2,t3)

#Hierarchical clustering using euclidiean distance
par(mfrow=c(1,1))
distMatrix <- dist(m, method='euclidean')
hc <- hclust(distMatrix, method='average')
plot(hc)
```



```
#Hierarchical clustering using DTW distance

distMatrix <- dist(m, method='DTW')
hc <- hclust(distMatrix, method='average')
plot(hc)
```