NON-STATIONARY TIME SERIES MODEL

**INTRODUCTION**

***Problem Description:*** Here in this problem we are interested

1. To simulate the 3 different non stationary class of model and comment about its characterstics properties.
2. To make the three series into stationary by using most appropriate methods.
3. To choose any real time non seosonal non stationary data set and comment about its type of non stationary pattern.

***Objective:*** The main objective of this problem is to simulate the 3 different non stationary class of model and comment about its characterstics properties and make them into stationary by using most appropriate method. Finally we want to choose any real time non seosonal non stationary data set and comment about its type of non stationary pattern.

***Pure random walk process(Zt = Zt-1 + at ):*** Random walk predicts that the value at time “t” will be equal to the last period value plus a stochastic (non-systematic) component that is a white noise, which means at is independent and identically distributed with mean “0” and variance “σ².” Random walk can also be named a process integrated of some order, a process with a unit root or a process with a stochastic trend. It is a non-mean-reverting process that can move away from the mean either in a positive or negative direction. Another characteristic of a random walk is that the variance evolves over time and goes to infinity as time goes to infinity; therefore, a random walk cannot be predicted.

***Pure random walk process with a drift(Zt = α + Zt-1 + at ):*** If the random walk model predicts that the value at time “t” will equal the last period’s value plus a constant, or drift (α), and a white noise term (at), then the process is random walk with a drift. It also does not revert to a long-run mean and has variance dependent on time.

***Deterministic trend(Zt = α + βt + at )***: Often a random walk with a drift is confused for a deterministic trend. Both include a drift and a white noise component, but the value at time “t” in the case of a random walk is regressed on the last period’s value (Zt-1), while in the case of a deterministic trend it is regressed on a time trend (βt). A non-stationary process with a deterministic trend has a mean that grows around a fixed trend, which is constant and independent of time.

*#Setting and getting the current working directory.*  
**setwd**("E:/M.Sc/SEM III/TIME\_SERIES\_ANALYSIS(MST371)/Practical Labs")  
**getwd**()

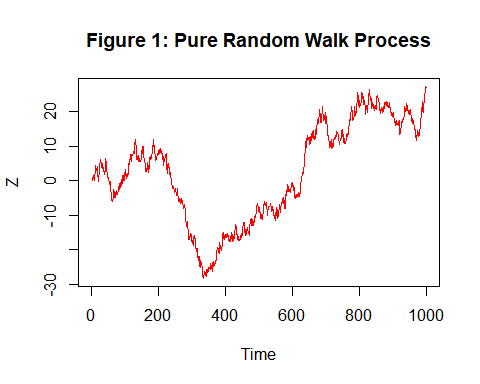
## [1] "E:/M.Sc/SEM III/TIME\_SERIES\_ANALYSIS(MST371)/Practical Labs"

**ANALYSIS**

**1. Simulate the 3 different non stationary class of model and comment about its characterstics properties.**

***1.1 Pure random walk process***

*#Setting the seed value.*  
**set.seed**(2)  
  
*#Obtaining the white noise sequence.*  
z <- a <- **rnorm**(1000)  
  
*#Simulating random walk process with 1000 observations*  
**for**(t **in** 2**:**1000){  
 a=**rnorm**(1000)  
 z[1]=0  
 z[t] = z[t-1] **+**a[t]  
}  
  
*#Plotting the random walk time series process.*   
**ts.plot**(z, main = "Figure 1: Pure Random Walk Process", xlab = "Time", ylab = "Z", col = "red")

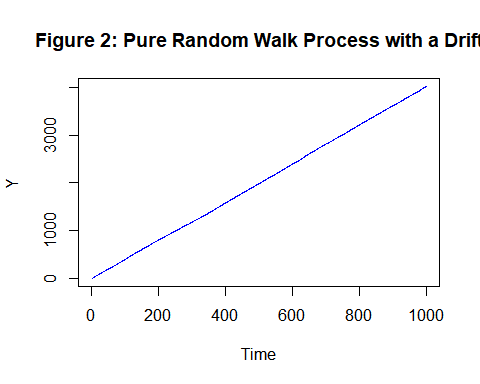


Thus, a pure random walk process has been simulated above.

Properties - The pure random walk process is contant in mean and it is non-stationary in variance i.e. the variance of a pure random process depends on time.

***1.2 Pure random walk process with a drift***

*#Setting the seed value.*  
**set.seed**(2)  
  
*#Obtaining the white noise sequence.*  
Y <- a <- **rnorm**(1000)  
  
*#Value of a drift component*  
A <- 4  
  
*#Simulating random walk process with 1000 observations*  
**for**(t **in** 2**:**1000){  
 a=**rnorm**(1000)  
 Y[1]=0  
 Y[t] = A **+** Y[t-1] **+**a[t]  
}  
  
*#Plotting the random walk time series process.*   
**ts.plot**(Y, main = "Figure 2: Pure Random Walk Process with a Drift", xlab = "Time", ylab = "Y", col = "blue")

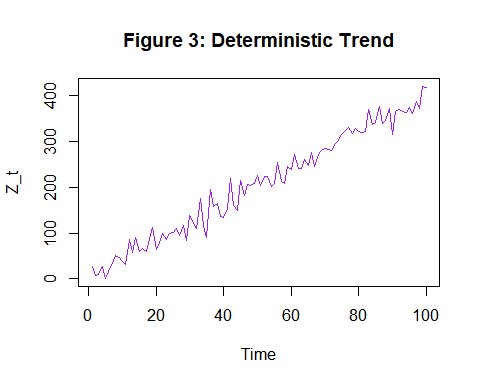


Thus, a pure random walk process with a drift has been simulated above.

Properties - The pure random walk process with a drift is non-stationary in both mean and variance i.e. both mean and variance of a pure random process with a drift depends on time.

***1.3 Deterministic trend***

*#Generating white noise sequence.*  
a\_t=**rnorm**(100, 0, 20)  
  
*#Trend component of deterministic process.*  
trend = 4 **+** 4**\***(1**:**100)  
  
*#Simulating deterministic trend.*  
z\_t = trend **+**a\_t  
  
*#Obtaining the time series plot of deterministic trend.*  
**ts.plot**(z\_t, main = "Figure 3: Deterministic Trend", xlab = "Time", ylab = "Z\_t", col = "purple")



Thus, a deterministic trend has been simulated above.

Properties - The deterministic is non-stationary in mean and stationary in variance i.e. both mean of a deterministic depends on time.

1. **To make the three series in to stationary by using most appropriate methods.**

***2.1 To make Pure random walk process stationary.***

Since the pure random walk process we simulated in 1.1 is a not stationary process we make them statianary by method of differencing.

*#Performing the first order differencing.*  
data=**diff**(z)  
  
*#Loading the package 'tseries'.*  
**library**(tseries)

## Warning: package 'tseries' was built under R version 4.0.5

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

*#Performing augmented Dickey Fuller test to check for the stationarity*  
**adf.test**(data)

## Warning in adf.test(data): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: data  
## Dickey-Fuller = -9.6755, Lag order = 9, p-value = 0.01  
## alternative hypothesis: stationary

***Interpretation:*** Thus, from adf test we observe that the p value is less than 0.01 < 0.05 therefore we reject the null hypothesis and conclude that the pure random walk process simulated in 1.1 has been converted into a stationary time series.

***2.2 To make Pure random walk process with a drift stationary.***

Since the pure random walk process with a drift we simulated in 1.2 is a not stationary process we make them statianary by method of differencing.

*#Performing the first order differencing.*  
data1=**diff**(Y)  
  
*#Loading the package 'tseries'.*  
**library**(tseries)  
  
*#Performing augmented Dickey Fuller test to check for the stationarity*  
**adf.test**(data1)

## Warning in adf.test(data1): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: data1  
## Dickey-Fuller = -9.6755, Lag order = 9, p-value = 0.01  
## alternative hypothesis: stationary

***Interpretation:*** Thus, from adf test we observe that the p value is less than 0.01 < 0.05 therefore we reject the null hypothesis and conclude that the pure random walk process with a a drift simulated in 1.2 has been converted into a stationary time series.

***2.3 To make deterministic trend stationary.***

Since the deterministic trend we simulated in 1.3 is a not stationary process we make them statianary by method of moving average.

In case of deterministic trend if we use first order differencing we loose information thus to avoid this we go for either OLS or moving average method of detrending.

*#Loading the package 'forecast'.*  
**library**(forecast)

## Warning: package 'forecast' was built under R version 4.0.5

*#Performing the first order moving average smoothing.*  
ma1=**ma**(z\_t,order=3)  
  
*#Obtaining the detrended series.*  
res1=z\_t **-** ma1  
  
*#Loading the package 'tseries'.*  
**library**(tseries)  
  
*#Performing augmented Dickey Fuller test to check for the stationarity*  
**adf.test**(res1[2**:**99])

## Warning in adf.test(res1[2:99]): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: res1[2:99]  
## Dickey-Fuller = -11.898, Lag order = 4, p-value = 0.01  
## alternative hypothesis: stationary

***Interpretation:*** Thus, from adf test we observe that the p value is less than 0.01 < 0.05 therefore we reject the null hypothesis and conclude that the deterministic trend simulated in 1.3 has been converted into a stationary time series.

1. **To choose any real time non seasonal non stationary data set and comment about its type of non stationary pattern.**

*#Loading the package 'readxl'.*  
**library**(readxl)  
  
*#Loading the non stationary monthly milk production dataset*  
milkProduction <- **read\_excel**("E:/M.Sc/SEM III/TIME\_SERIES\_ANALYSIS(MST371)/milkProduction.xlsx")  
  
*#Obtaining the first few records of milk production data.*  
**head**(milkProduction)

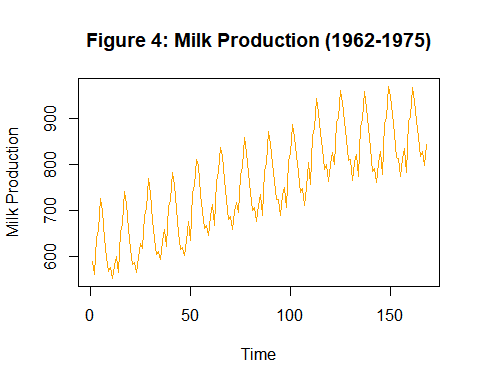
## # A tibble: 6 x 2  
## Month Milk\_Production  
## <dttm> <dbl>  
## 1 1962-01-01 00:00:00 589  
## 2 1962-02-01 00:00:00 561  
## 3 1962-03-01 00:00:00 640  
## 4 1962-04-01 00:00:00 656  
## 5 1962-05-01 00:00:00 727  
## 6 1962-06-01 00:00:00 697

*#Extracing the data for the production variable which we are interested in.*  
production <- milkProduction**$**Milk\_Production  
  
*#Now converting the it into a time series data.*  
production <- **ts**(production)   
  
*#Here. we are checking if the dataset has been converted into a timeseries plot.*  
**class**(production)

## [1] "ts"

Hence, now the dataset we are interested in is a timeseries data.

*#Obtaining timeseries plot for milk production data.*   
**ts.plot**(production, main = "Figure 4: Milk Production (1962-1975)", xlab = "Time", ylab = "Milk Production", col = "orange")



***Interpretation:*** We observe that the above data is a pure random walk with a drift since the points rarely reverts to a trend line one would place through the data.

**CONCLUSION**

Thus, in the above analysis we have simulated random walk time series process of size , random walk time series process with a drift of size , deterministic trend time series process of size and futher converted them into stationary process by 1st order differencing and first order moving average smoothing respectively.

Later we have also taken a monthly milk production data over the period of years 1962 - 1975 we observed that the time series data belongs to the class of pure random walk process with a drift since the points rarely reverts to a trend line one would place through the data.