Forecasting\_Project

VISHAL PATNAIK DAMODARAPATRUNI - s3811521

03/10/2021

Table of Contents

[Introduction 2](#_Toc85919975)

[Task 1: 2](#_Toc85919976)

[Task 2: 2](#_Toc85919977)

[Task 3: 2](#_Toc85919978)

[Part (a): 3](#_Toc85919979)

[Part (b): 3](#_Toc85919980)

[Method 3](#_Toc85919981)

[Task 1 3](#_Toc85919982)

[Data 3](#_Toc85919983)

[Descriptive Analysis 5](#_Toc85919984)

[Checking for Stationary in the series 12](#_Toc85919985)

[Impact of components on each time series. 18](#_Toc85919986)

[Suitable distributed lag models (Multivariate Analysis). 23](#_Toc85919987)

[Exponential Smoothing 70](#_Toc85919988)

[State Space Model Variations 74](#_Toc85919989)

[Forecasting 86](#_Toc85919990)

[Task 2 88](#_Toc85919991)

[Data 88](#_Toc85919992)

[Descriptive Analysis 90](#_Toc85919993)

[Checking for Stationary in the series 97](#_Toc85919994)

[Suitable Distributed lag models (Univariate analysis). 103](#_Toc85919995)

[Exponential Smoothing 131](#_Toc85919996)

[State Space Model Variations 139](#_Toc85919997)

[Forecasting 151](#_Toc85919998)

[Task 3 154](#_Toc85919999)

[Part (a) 154](#_Toc85920000)

[Part (b) 190](#_Toc85920001)

[Conclusion 208](#_Toc85920002)

[Task 1: 208](#_Toc85920003)

[Task 2: 209](#_Toc85920004)

[Task 3: 210](#_Toc85920005)

[Part (a): 210](#_Toc85920006)

[Part (b): 211](#_Toc85920007)

[References 212](#_Toc85920008)

# Introduction

This analysis has three parts: Task 1: Forecasting Mortality rate data. Task 2: Forecasting First Flowering day data. Task 3: Forecasting Rank-based Order similarity metric data.

## Task 1:

Here, we will forecast the Mortality rate data for the next four weeks using the best fit model. To get this best model we have three approaches.  
1. Suitable Distributed Lag and Dynamic LM 2. Smoothing methods 3. State Space models

The best model is the one that has the best MASE score as well as gives the best residual analysis.

## Task 2:

Here, we will forecast the First Flowering day data for the next four years using the best fit model. To get this best model we have three approaches.  
1. Suitable Distributed Lag and Dynamic LM (With and without slope) 2. Smoothing methods 3. State Space models

The best model is the one that has the best MASE score as well as gives the best residual analysis.

## Task 3:

This again has 2 parts.

Part (a)

Part (b)

### Part (a):

Here, we will forecast the Rank-based Order similarity metric data for the next three years using the best fit model among Distributed Lag and Dynamic LM The best model is the one that has the best MASE score as well as gives the best residual analysis.

### Part (b):

Here, we will forecast the Rank-based Order similarity metric data during the Millennium drought period (1997 - 2009) for the next three years using the best fit model among Distributed Lag and Dynamic LM. The best model is the one that has the best MASE score as well as gives the best residual analysis.

# Method

## Task 1

The following packages are for all the three parts.

library(dplyr)  
library(forecast) # Forecasting Functions for Time Series and Linear Models. [1] - https://cran.r-project.org/web/packages/forecast/index.html  
library(dLagM) # Distributed lag model.  
library(lmtest) # Testing Linear Regression Models. [2] - https://cran.r-project.org/web/packages/lmtest/index.html  
library(tidyr)  
library(tseries) # Time Series Analysis and Computational Finance.[3] - https://cran.r-project.org/web/packages/tseries/index.html  
library(fUnitRoots) # To analyze trends and unit roots in financial time series. [4] - https://cran.r-project.org/web/packages/fUnitRoots/index.html  
library(expsmooth) # Forecasting with Exponential Smoothing. [5] - https://cran.r-project.org/web/packages/expsmooth/index.html  
library(TSA) # Time Series Analysis.  
library(urca) # Unit Root and Cointegration Tests. [6] - https://cran.r-project.org/web/packages/urca/index.html  
library(readr)  
library(xts)

### Data

The data here used is the weekly averages of potential effects of both climate and pollution on disease specific mortality between the years 2010-2020.

v\_Mortality\_data <- read.csv("mort.csv", header = TRUE)  
head(v\_Mortality\_data)

## ï.. mortality temp chem1 chem2 particle.size  
## 1 1 97.85 72.38 11.51 3.37 72.72  
## 2 2 104.64 67.19 8.92 2.59 49.60  
## 3 3 94.36 62.94 9.48 3.29 55.68  
## 4 4 98.05 72.49 10.28 3.04 55.16  
## 5 5 95.85 74.25 10.57 3.39 66.02  
## 6 6 95.98 67.88 7.99 2.57 44.01

# Using str() to check the type of each column.  
str(v\_Mortality\_data)

## 'data.frame': 508 obs. of 6 variables:  
## $ ï.. : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ mortality : num 97.8 104.6 94.4 98 95.8 ...  
## $ temp : num 72.4 67.2 62.9 72.5 74.2 ...  
## $ chem1 : num 11.51 8.92 9.48 10.28 10.57 ...  
## $ chem2 : num 3.37 2.59 3.29 3.04 3.39 2.57 2.35 3.38 1.5 2.56 ...  
## $ particle.size: num 72.7 49.6 55.7 55.2 66 ...

Checking for Missing values.

colSums(is.na(v\_Mortality\_data))

## ï.. mortality temp chem1 chem2   
## 0 0 0 0 0   
## particle.size   
## 0

There are no missing values in the data.

Checking the class of v\_Mortality\_data (It should be a data frame.)

class(v\_Mortality\_data)

## [1] "data.frame"

Setting frequency = 365.27/7. Since weekly data.

v\_Mortality\_data\_TS <- ts(v\_Mortality\_data$mortality, start = c(2010, 1), frequency = (365.27/7))  
v\_Mortality\_temp\_data\_TS <- ts(v\_Mortality\_data$temp, start = c(2010, 1), frequency = (365.27/7))  
v\_Mortality\_chem1\_data\_TS <- ts(v\_Mortality\_data$chem1, start = c(2010, 1), frequency = (365.27/7))  
v\_Mortality\_chem2\_data\_TS <- ts(v\_Mortality\_data$chem2, start = c(2010, 1), frequency = (365.27/7))  
v\_Mortality\_particle.size\_data\_TS <- ts(v\_Mortality\_data$particle.size, start = c(2010, 1), frequency = (365.27/7))

Confirming the class of each time series object.

class(v\_Mortality\_data\_TS)

## [1] "ts"

class(v\_Mortality\_temp\_data\_TS)

## [1] "ts"

class(v\_Mortality\_chem1\_data\_TS)

## [1] "ts"

class(v\_Mortality\_chem2\_data\_TS)

## [1] "ts"

class(v\_Mortality\_particle.size\_data\_TS)

## [1] "ts"

Now let us perform descriptive analysis on each time series object.

### Descriptive Analysis

#### Mortality Rate

plot(v\_Mortality\_data\_TS, type = "b", xlab = "weeks", ylab = "Mortality rate", main = "Time series plot for mortality rate from 2010 to 2020 (508 weeks)", pch = 1)  
legend("topright", inset = .03, title = "Rate", legend = "Mortality rate series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart

Description automatically generated with low confidence

Fig 1.1: Mortality Rate - Time series plot.

McLeod.Li.test(y = v\_Mortality\_data\_TS, main = "McLeod-Li Test Statistics for Mortality Rate.")

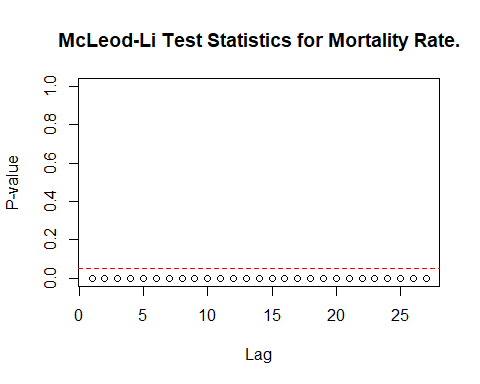


Fig 1.2: McLeod-Li Test Statistics for Mortality Rate.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention at multiple points in the series.
3. From the series plot, we can conclude that there is seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. Therefore, there is no change Autoregressive and moving average behaviour.
6. Also, we cannot see change in variance.

#### Temperature

plot(v\_Mortality\_temp\_data\_TS, type = "b", xlab = "weeks", ylab = "Temperature", main = "Time series plot for temperature from 2010 to 2020 (508 weeks)", pch = 1)  
legend("top", inset = .03, title = "Temperature", legend = "Temperature series", horiz = TRUE, cex = 0.6, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

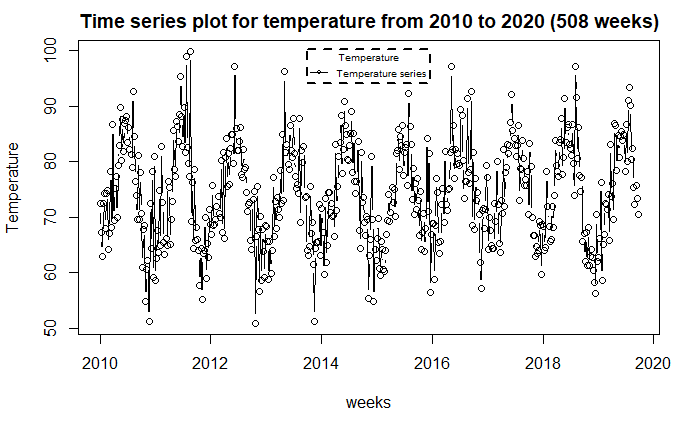


Fig 1.3: Temperature - Time series plot.

McLeod.Li.test(y = v\_Mortality\_temp\_data\_TS, main = "McLeod-Li Test Statistics for Temperature")

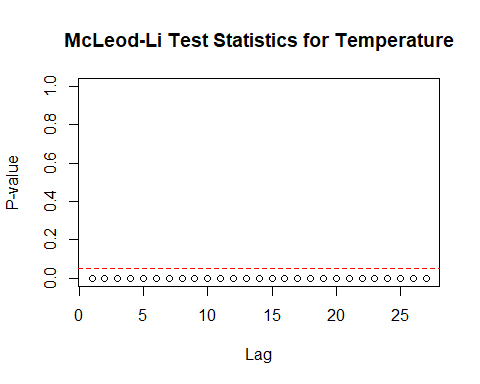


Fig 1.4: McLeod-Li Test Statistics for Precipitation.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention at multiple points in the series.
3. From the series plot, we can conclude that there is seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. Therefore, there is no change Autoregressive and moving average behaviour.
6. Also, we cannot see change in variance.

#### Chemical Emission 1

plot(v\_Mortality\_chem1\_data\_TS, type = "b", xlab = "weeks", ylab = "Chemical Emission 1", main = "Time series plot for Chemical Emission 1 from 2010 to 2020 (508 weeks)", pch = 1)  
legend("topright", inset = .03, title = "Chemical Emission 1", legend = "Chemical Emission 1 series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

A picture containing diagram

Description automatically generated

Fig 1.3: Chemical Emission 1 - Time series plot.

McLeod.Li.test(y = v\_Mortality\_chem1\_data\_TS, main = "McLeod-Li Test Statistics for Chemical Emission 1.")

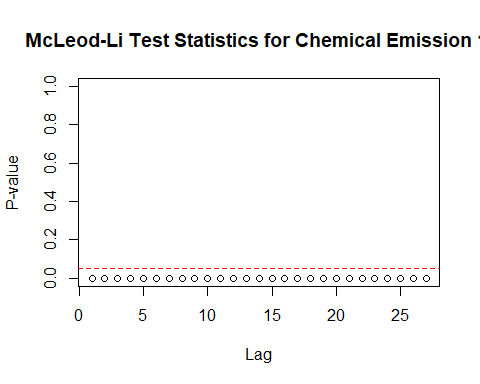


Fig 1.4: McLeod-Li Test Statistics for Chemical Emission 1.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention at multiple points in the series.
3. From the series plot, we can conclude that there is seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. Therefore, there is no change Autoregressive and moving average behaviour.
6. Also, we cannot see change in variance.

#### Chemical Emission 2

plot(v\_Mortality\_chem2\_data\_TS, type = "b", xlab = "weeks", ylab = "Chemical Emission 2", main = "Time series plot for Chemical Emission 2 from 2010 to 2020 (508 weeks)", pch = 1)  
legend("topright", inset = .03, title = "Chemical Emission 2", legend = "Chemical Emission 2 series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

A picture containing scatter chart

Description automatically generated

Fig 1.3: Chemical Emission 2 - Time series plot.

McLeod.Li.test(y = v\_Mortality\_chem2\_data\_TS, main = "McLeod-Li Test Statistics for Chemical Emission 2.")

Graphical user interface, application

Description automatically generated

Fig 1.4: McLeod-Li Test Statistics for Chemical Emission 2.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data. But it seems like a downward trend.
2. There is an intervention at multiple points in the series.
3. From the series plot, we can conclude that there is seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. Therefore, there is no change Autoregressive and moving average behaviour.
6. Also, we cannot see change in variance.

#### Partical size

plot(v\_Mortality\_particle.size\_data\_TS, type = "b", xlab = "weeks", ylab = "Partical size", main = "Time series plot for Particle size from 2010 to 2020 (508 weeks)", pch = 1)  
legend("topleft", inset = .03, title = "Partical size", legend = "Partical size", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

A picture containing chart

Description automatically generated

Fig 1.3: Partical size - Time series plot.

McLeod.Li.test(y = v\_Mortality\_particle.size\_data\_TS, main = "McLeod-Li Test Statistics for Partical size")

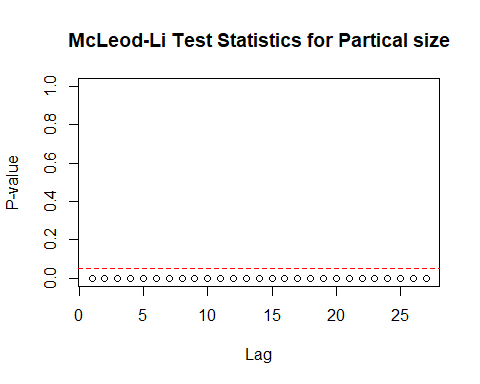


Fig 1.4: McLeod-Li Test Statistics for Partical size.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention at multiple points in the series.
3. From the series plot, we can conclude that there is seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. Therefore, there is no change Autoregressive and moving average behaviour.
6. Also, we cannot see change in variance.

### Checking for Stationary in the series

# Function to check Stationary on the series.   
Stationary\_Check <- function(x, m1, m2) {  
   
 # Analysing trends by plotting ACF and PACF.  
 par(mfrow = c(1,2))  
 acf(x, main = m1)  
 pacf(x, main = m2)  
   
 # Lag for ADF test  
 d = ar(x)$order  
   
 # Conducting Augmented Dickey-Fuller test.  
 adf.test(x, k = d)  
}

Checking for Stationary on Mortality Rate series.

Stationary\_Check(v\_Mortality\_data\_TS, "Mortality Rate - ACF plot", "Mortality Rate - PACF plot")

## Warning in adf.test(x, k = d): p-value smaller than printed p-value

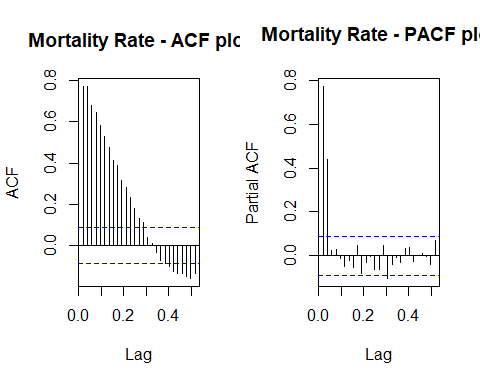


Fig 1.5: Mortality Rate - ACF

Fig 1.6: Mortality Rate - ACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -5.161, Lag order = 2, p-value = 0.01  
## alternative hypothesis: stationary

The seasonal pattern in the significant lags suggests that there is no trend in the series.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Mortality rate series is Stationary.

Checking for Stationary on Temperature series.

Stationary\_Check(v\_Mortality\_temp\_data\_TS, "Temperature - ACF plot", "Temperature - PACF plot")

## Warning in adf.test(x, k = d): p-value smaller than printed p-value

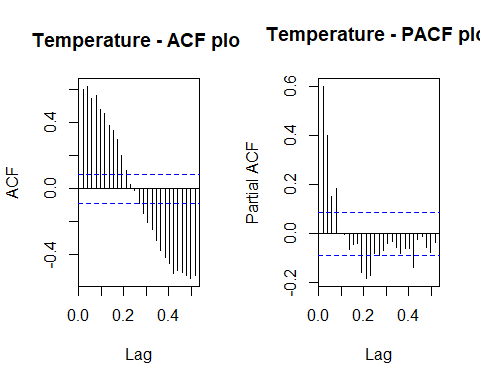


Fig 1.7: Temperature - ACF

Fig 1.8: Temperature - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -8.2554, Lag order = 22, p-value = 0.01  
## alternative hypothesis: stationary

Fig 1.7: Temperature - ACF Fig 1.8: Temperature - PACF

The seasonal pattern in the significant lags suggests that there is no trend in the series.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Temperature series is Stationary.

Checking for Stationary on Chemical Emission 1 series.

Stationary\_Check(v\_Mortality\_chem1\_data\_TS, "Chemical Emission 1 - ACF plot", "Chemical Emission 1 - PACF plot")

## Warning in adf.test(x, k = d): p-value smaller than printed p-value

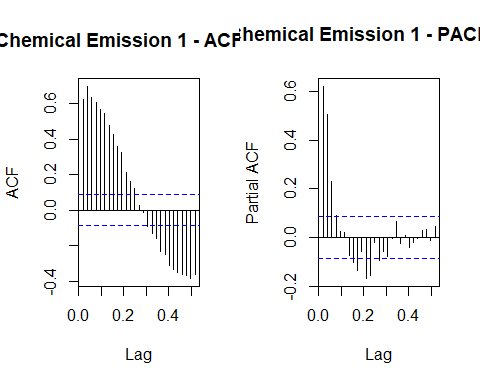


Fig 1.9: Chemical Emission 1 - ACF

Fig 1.10: Chemical Emission 1 - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -8.1588, Lag order = 16, p-value = 0.01  
## alternative hypothesis: stationary

The seasonal pattern in the significant lags suggests that there is no trend in the series.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value : ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Chemical Emission 1 series is Stationary.

Checking for Stationary on Chemical Emission 2 series.

Stationary\_Check(v\_Mortality\_chem2\_data\_TS, "Chemical Emission 2 - ACF plot", "Chemical Emission 2 - PACF plot")

## Warning in adf.test(x, k = d): p-value smaller than printed p-value

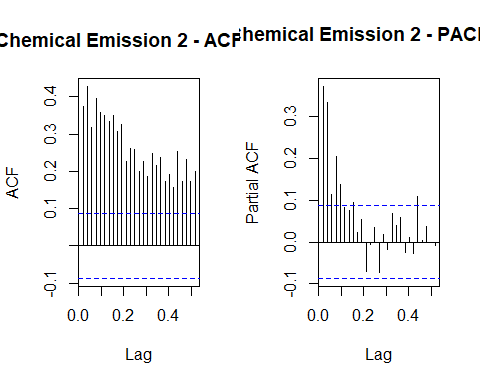


Fig 1.11: Chemical Emission 2 - ACF

Fig 1.12: Chemical Emission 2 - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -5.3362, Lag order = 8, p-value = 0.01  
## alternative hypothesis: stationary

The decrease in the ACF plot and a high peak in the PACF plot in the beginning, suggests that there is some pattern in the Chemical Emission 2 series.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Chemical Emission 2 series is Stationary.

Checking for Stationary on Particle Size data.

Stationary\_Check(v\_Mortality\_particle.size\_data\_TS, "Particle Size - ACF plot", "Particle Size - PACF plot")

## Warning in adf.test(x, k = d): p-value smaller than printed p-value

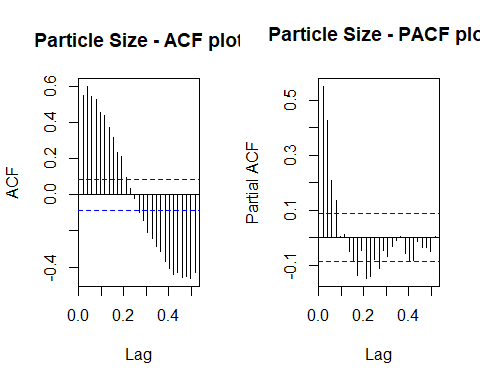


Fig 1.13: Particle size - ACF

Fig 1.14: Particle Size - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -7.2956, Lag order = 14, p-value = 0.01  
## alternative hypothesis: stationary

The seasonal pattern in the significant lags suggests that there is no trend in the series.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value : ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Particle size series is Stationary.

Therefore, no differentiation is required. As the two series are stationary.

### Impact of components on each time series.

The components of a series are usually,

1. Seasonality
2. Trend
3. Remainder

We should decompose the time series into the above components as we can see the impact of these components on the series data.

For this STL decomposition is used, as there is intervention in some of the series. This intervention is might be due to outliers and STL decomposition is robust in the case of outliers.

Decomposing Mortality series into components.

v\_Mortality\_stl\_decomp <- stl(v\_Mortality\_data\_TS, t.window = 15, s.window = "periodic", robust = TRUE)   
plot(v\_Mortality\_stl\_decomp, main = "Decomposing Mortality Series into components")

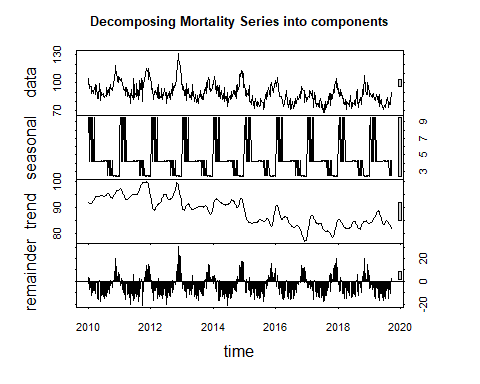


Fig 1.15: Decomposing Mortality series into components - stl decomposition.

1. The seasonality component shows peaks at the same points suggesting some seasonality.
2. The trend in the series data is not shown by the trend component.
3. Remainder component shows a high intervention point around 2013.

Decomposing Temperature series into components.

v\_temp\_stl\_decomp <- stl(v\_Mortality\_temp\_data\_TS, t.window = 15, s.window = "periodic", robust = TRUE)   
plot(v\_temp\_stl\_decomp, main = "Decomposing Temperature Series into components")

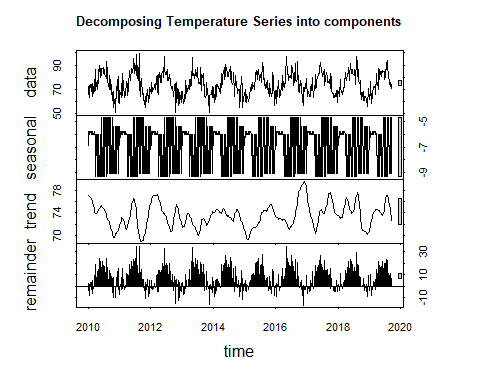


Fig 1.16: Decomposing Temperature series into components - stl decomposition.

1. The seasonal adjusted series is not meaningful in this case as it much deviated from the original series. Suggesting that there is no seasonality in the series. But we observed seasonality in the series. This makes that there is no sense in the seasonality components.
2. The trend in the series data is not shown by the trend component.
3. Remainder component shows no high intervention points.

Decomposing Chemical Emission 1 series into components.

v\_Chem1\_stl\_decomp <- stl(v\_Mortality\_chem1\_data\_TS, t.window = 15, s.window = "periodic", robust = TRUE)   
plot(v\_Chem1\_stl\_decomp, main = "Decomposing Chemical Emission 1 Series into components")

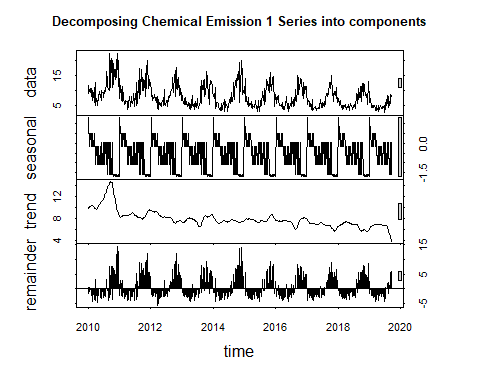


Fig 1.17: Decomposing Chemical Emission 1 series into components - stl decomposition.

1. The seasonal adjusted series is not meaningful in this case as it much deviated from the original series. Suggesting that there is no seasonality in the series. But we observed seasonality in the series. This makes that there is no sense in the seasonality components.
2. The trend in the series data is not shown by the trend component.
3. Remainder component shows a high intervention point at 2011 and 2015.

Decomposing Chemical Emission 2 series into components.

v\_Chem2\_stl\_decomp <- stl(v\_Mortality\_chem2\_data\_TS, t.window = 15, s.window = "periodic", robust = TRUE)   
plot(v\_Chem2\_stl\_decomp, main = "Decomposing Chemical Emission 2 Series into components")

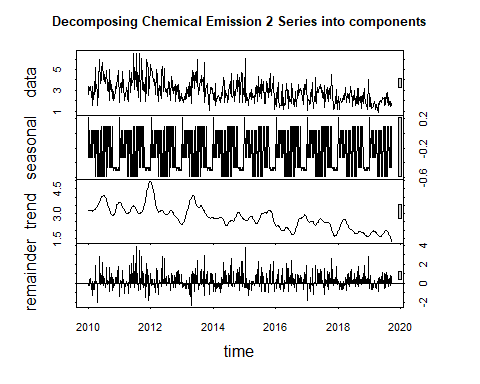


Fig 1.17: Decomposing Chemical Emission 2 series into components - stl decomposition.

1. The seasonal adjusted series is not meaningful in this case as it much deviated from the original series. Suggesting that there is no seasonality in the series. But we observed seasonality in the series. This makes that there is no sense in the seasonality components.
2. The trend in the series data is shown exactly by the trend component.
3. Remainder component shows a high intervention point at multiple points.

Decomposing COPPER price series into components.

v\_Part\_stl\_decomp <- stl(v\_Mortality\_particle.size\_data\_TS, t.window = 15, s.window = "periodic", robust = TRUE)   
plot(v\_Part\_stl\_decomp, main = "Decomposing Particle price Series into components")

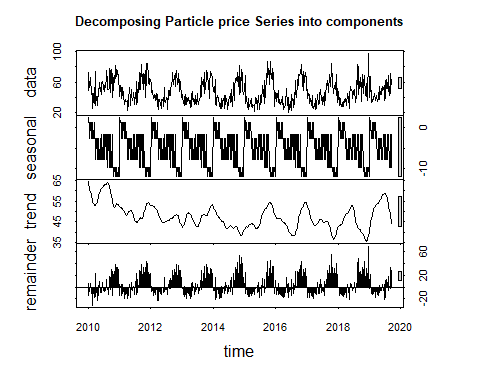


Fig 1.18: Decomposing Particle price series into components - stl decomposition.

1. The seasonal adjusted series is not meaningful in this case as it much deviated from the original series. Suggesting that there is no seasonality in the series. But we observed seasonality in the series. This makes that there is no sense in the seasonality components.
2. The trend in the series data is shown by the trend component.
3. Remainder component shows a high intervention point at multiple points.

### Suitable distributed lag models (Multivariate Analysis).

Before this let us find the correlation between the two series.

# Calculating the correlation coefficient  
cor(v\_Mortality\_data\_TS, v\_Mortality\_temp\_data\_TS)

## [1] -0.4386396

cor(v\_Mortality\_data\_TS, v\_Mortality\_chem1\_data\_TS)

## [1] 0.5574476

cor(v\_Mortality\_data\_TS, v\_Mortality\_chem2\_data\_TS)

## [1] 0.2569989

cor(v\_Mortality\_data\_TS, v\_Mortality\_particle.size\_data\_TS)

## [1] 0.4438713

This suggests that Mortality rate has a strong correlation with Chemical emission 1 and Particle size.

As we are going to forecast the Mortality Rate data our dependent variable “y” will be Mortality Rate series object and independent variable “x” will be Chemical emission 1 and Particle size. Since multivariate analysis. For this let us convert the entire data set into time series.

v\_data\_TS <- ts(v\_Mortality\_data, start = c(2010, 1), frequency = (365.27/7))  
cor(v\_data\_TS)

## ï.. mortality temp chem1 chem2  
## ï.. 1.00000000 -0.4587450 0.05951337 -0.32983451 -0.4999335  
## mortality -0.45874498 1.0000000 -0.43863962 0.55744759 0.2569989  
## temp 0.05951337 -0.4386396 1.00000000 -0.09785582 0.4043740  
## chem1 -0.32983451 0.5574476 -0.09785582 1.00000000 0.5130047  
## chem2 -0.49993345 0.2569989 0.40437401 0.51300467 1.0000000  
## particle.size -0.07664951 0.4438713 -0.01723095 0.86611747 0.4679340  
## particle.size  
## ï.. -0.07664951  
## mortality 0.44387133  
## temp -0.01723095  
## chem1 0.86611747  
## chem2 0.46793404  
## particle.size 1.00000000

Convert column names.

colnames(v\_data\_TS)<-c("n", "y", "x1", "x2", "x3", "x4")  
  
## Where y -> mortality  
## x1 -> temp  
## x2 -> chem1  
## x3 -> chem2  
## x4 -> particle.size

#### Finite distributed lag model

Getting q values for finite distributed lag model based on MASE values.

for ( i in 1:10){  
 model\_1 = dlm(formula=y ~ x2 + x4, data = data.frame(v\_data\_TS), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_1$model), "BIC = ", BIC(model\_1$model), "MASE =", MASE(model\_1)$MASE, "\n")  
}

## q = 1 AIC = 3543.48 BIC = 3568.851 MASE = 1.145141   
## q = 2 AIC = 3490.679 BIC = 3524.491 MASE = 1.087568   
## q = 3 AIC = 3464.306 BIC = 3506.551 MASE = 1.054802   
## q = 4 AIC = 3404.234 BIC = 3454.905 MASE = 1.002905   
## q = 5 AIC = 3381.397 BIC = 3440.485 MASE = 0.9804769   
## q = 6 AIC = 3353.635 BIC = 3421.132 MASE = 0.9585366   
## q = 7 AIC = 3321.442 BIC = 3397.341 MASE = 0.9219601   
## q = 8 AIC = 3305.874 BIC = 3390.166 MASE = 0.9009314   
## q = 9 AIC = 3297.186 BIC = 3389.864 MASE = 0.8920576   
## q = 10 AIC = 3293.419 BIC = 3394.473 MASE = 0.8897851

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC-BIC  
  
finite\_dlm\_mort = dlm(formula=y ~ x2 + x4, data = data.frame(v\_data\_TS), q = 10)  
summary(finite\_dlm\_mort)

##   
## Call:  
## lm(formula = as.formula(model.formula), data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.065 -3.835 -0.260 3.552 33.056   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 70.062958 1.493864 46.901 < 2e-16 \*\*\*  
## x2.t -0.121965 0.220995 -0.552 0.58128   
## x2.1 0.210913 0.221677 0.951 0.34186   
## x2.2 0.406731 0.234094 1.737 0.08295 .   
## x2.3 0.360535 0.235634 1.530 0.12667   
## x2.4 0.647814 0.242696 2.669 0.00786 \*\*   
## x2.5 0.004949 0.244396 0.020 0.98385   
## x2.6 0.483837 0.243557 1.987 0.04755 \*   
## x2.7 0.381800 0.239767 1.592 0.11197   
## x2.8 0.201629 0.238500 0.845 0.39831   
## x2.9 0.175025 0.226108 0.774 0.43927   
## x2.10 0.010672 0.225713 0.047 0.96231   
## x4.t 0.154497 0.050470 3.061 0.00233 \*\*   
## x4.1 -0.080595 0.050770 -1.587 0.11307   
## x4.2 -0.061847 0.052011 -1.189 0.23499   
## x4.3 -0.096085 0.052257 -1.839 0.06658 .   
## x4.4 -0.059814 0.053976 -1.108 0.26836   
## x4.5 0.020664 0.054232 0.381 0.70335   
## x4.6 -0.057614 0.054019 -1.067 0.28672   
## x4.7 0.017066 0.052546 0.325 0.74548   
## x4.8 0.034731 0.052547 0.661 0.50896   
## x4.9 0.031218 0.050799 0.615 0.53915   
## x4.10 0.027749 0.050399 0.551 0.58218   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.444 on 475 degrees of freedom  
## Multiple R-squared: 0.6063, Adjusted R-squared: 0.5881   
## F-statistic: 33.25 on 22 and 475 DF, p-value: < 2.2e-16  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 3293.419 3394.473

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model.**

**HA: The data fits the Finite distributed lag model.**

**Interpretations:**

F - statistic is 33.25

R - squared is 0.6063

Adjusted R - squared is 0.5881

Degrees of freedom - DF are (22, 475)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Finite distributed lag model.

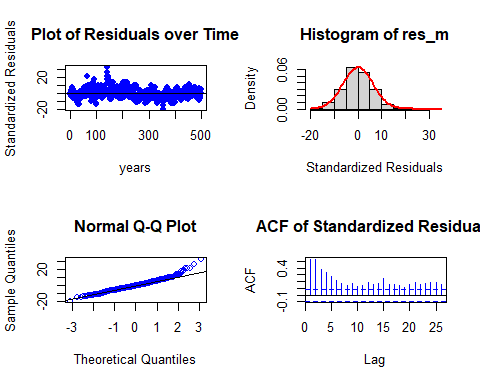
This model suggests that there is only 58.81% of data variance. Suggesting that the model explains only 58.81% of the trend. Which implies that the model shows some trend.

Now let us check the residual analysis.

**Residual analysis**

# Function for residual analysis.  
  
res\_analysis <- function(res\_m) {  
   
 par(mfrow = c(2, 2))  
 # Scatter plot for model residuals  
 plot(res\_m, type = "b", pch = 19, col = "blue", xlab = "years", ylab = "Standardized Residuals", main = "Plot of Residuals over Time")  
  
 abline(h = 0)  
   
 # Standard distribution  
 hist(res\_m, xlab = 'Standardized Residuals', freq = FALSE)  
 curve(dnorm(x, mean = mean(res\_m), sd = sd(res\_m)), col = "red", lwd = 2, add = TRUE, yaxt = "n")  
   
 # QQplot for model residuals  
 qqnorm(res\_m, col = c("blue"))  
 qqline(res\_m)  
   
 # Auto-Correlation Plot  
 acf(res\_m, main = "ACF of Standardized Residuals",col=c("blue"))  
   
 # Shapiro Wilk test  
 shapiro.test(res\_m)  
   
}

res\_analysis(residuals(finite\_dlm\_mort$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.97054, p-value = 1.86e-08

Residual Analysis for Finite DLM:

1. The data points are below the line at the start and above the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Therefore, Further analysis is needed by adding polynomial to the lag model.

#### Polynomial distributed lag model

Polynomial distributed lag model makes predictors with only one variable. There fore, let us fit models on the mortality rate separately.

y = v\_Mortality\_data\_TS # Independent variable  
x1 = v\_Mortality\_chem1\_data\_TS # Dependent variable  
x2 = v\_Mortality\_particle.size\_data\_TS # Dependent variable

##### Polynomial distributed lag model with Chemical emission 1

for (i in 1:3){  
 model\_2 <- polyDlm(x = as.vector(x1) , y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_2$model), "BIC = ", BIC(model\_2$model), "MASE =", MASE(model\_2)$MASE, "\n")  
}

## q = 1 k = 1 AIC = 3554.683 BIC = 3571.597 MASE = 1.171233   
## q = 2 k = 2 AIC = 3505.901 BIC = 3527.034 MASE = 1.122964   
## q = 3 k = 3 AIC = 3478.836 BIC = 3504.183 MASE = 1.104185

Let us fit a polynomial model of order 3. Since least AIC, BIC and MASE scores.

# Ploynomial DLM  
  
PolyDLM\_model\_mort\_chem1 = polyDlm(x = as.vector(x1), y = as.vector(y), q = 3, k = 3, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 0.410 0.135 3.03 2.56e-03  
## beta.1 0.190 0.134 1.42 1.57e-01  
## beta.2 0.764 0.134 5.70 2.07e-08  
## beta.3 0.654 0.135 4.84 1.72e-06

summary(PolyDLM\_model\_mort\_chem1)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -17.8615 -5.3630 -0.4056 4.4528 31.0281   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 72.6825 0.8845 82.170 < 2e-16 \*\*\*  
## z.t0 0.4103 0.1353 3.032 0.00256 \*\*   
## z.t1 -1.1101 0.5367 -2.069 0.03910 \*   
## z.t2 1.1363 0.4719 2.408 0.01640 \*   
## z.t3 -0.2464 0.1033 -2.384 0.01750 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.527 on 500 degrees of freedom  
## Multiple R-squared: 0.4369, Adjusted R-squared: 0.4324   
## F-statistic: 97 on 4 and 500 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 97

R - squared is 0.4369

Adjusted R - squared is 0.4324

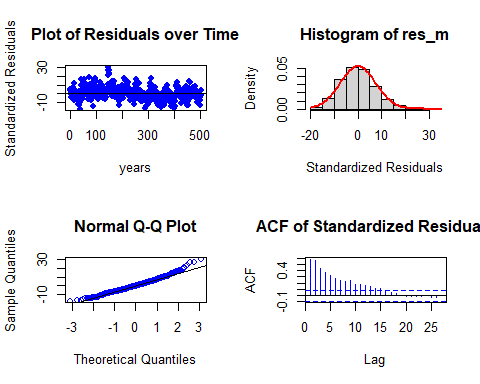
Degrees of freedom - DF are (4, 500)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Polynomial distributed lag model.

This model suggests that there is only 43.24% of data variance. Suggesting that the model explains only 43.24% of the trend. Which implies that the model shows some trend.

**Residual analysis**

res\_analysis(residuals(PolyDLM\_model\_mort\_chem1$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.98453, p-value = 3.338e-05

Residual Analysis for Polynomial DLM with Chem1:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

##### Polynomial distributed lag model with Particle size

for (i in 1:3){  
 model\_2 <- polyDlm(x = as.vector(x2) , y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_2$model), "BIC = ", BIC(model\_2$model), "MASE =", MASE(model\_2)$MASE, "\n")  
}

## q = 1 k = 1 AIC = 3653.402 BIC = 3670.316 MASE = 1.306209   
## q = 2 k = 2 AIC = 3623.85 BIC = 3644.983 MASE = 1.264552   
## q = 3 k = 3 AIC = 3606.089 BIC = 3631.437 MASE = 1.260364

Let us fit a polynomial model of order 3. Since least AIC, BIC and MASE scores.

# Ploynomial DLM  
  
PolyDLM\_model\_mort\_part = polyDlm(x = as.vector(x2), y = as.vector(y), q = 3, k = 3, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 0.1260 0.0342 3.670 2.65e-04  
## beta.1 0.0272 0.0343 0.793 4.28e-01  
## beta.2 0.1360 0.0343 3.970 8.33e-05  
## beta.3 0.1260 0.0342 3.700 2.42e-04

summary(PolyDLM\_model\_mort\_part)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -21.287 -5.938 -0.093 5.186 32.825   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 68.97944 1.49483 46.145 < 2e-16 \*\*\*  
## z.t0 0.12568 0.03421 3.673 0.000265 \*\*\*  
## z.t1 -0.31087 0.14126 -2.201 0.028218 \*   
## z.t2 0.26679 0.12440 2.145 0.032469 \*   
## z.t3 -0.05436 0.02728 -1.993 0.046819 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.538 on 500 degrees of freedom  
## Multiple R-squared: 0.2756, Adjusted R-squared: 0.2698   
## F-statistic: 47.55 on 4 and 500 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 47.55

R - squared is 0.2756

Adjusted R - squared is 0.2698

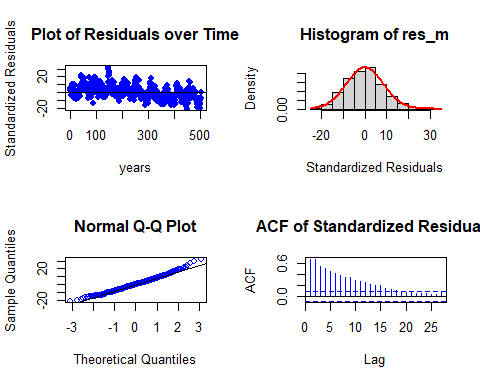
Degrees of freedom - DF are (4, 500)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Polynomial distributed lag model.

This model suggests that there is only 26.98% of data variance. Suggesting that the model explains only 26.98% of the trend. Which implies that the model shows some trend.

**Residual analysis**

res\_analysis(residuals(PolyDLM\_model\_mort\_part$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99092, p-value = 0.003401

Residual Analysis for Polynomial DLM with part:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (0.0034) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Now let us fit Koyck model in the similar war. Since it does not take multiple predictors.

#### Koyck model

##### Koyck with Chemical Emission 1

# Koyk DLM  
  
Koyck\_DLM\_mort\_chem1 = koyckDlm(x = as.vector(x1) , y = as.vector(y))  
summary(Koyck\_DLM\_mort\_chem1)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -20.19302 -4.18513 -0.06397 3.61514 23.08021   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 24.50952 2.55334 9.599 < 2e-16 \*\*\*  
## Y.1 0.66674 0.03637 18.331 < 2e-16 \*\*\*  
## X.t 0.63624 0.15311 4.155 3.81e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.891 on 504 degrees of freedom  
## Multiple R-Squared: 0.6544, Adjusted R-squared: 0.653   
## Wald test: 443.5 on 2 and 504 DF, p-value: < 2.2e-16   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: 73.5455 0.6362403 0.6667435

**Hypotheses:**

**H0: The data doesn′t fit the Koyck distributed lag model.**

**HA: The data fits the Koyck distributed lag model.**

**Interpretations:**

Wald test statistic is 443.5

R - squared is 0.6544

Adjusted R - squared is 0.653

Degrees of freedom - DF are (2, 504)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Koyck distributed lag model.

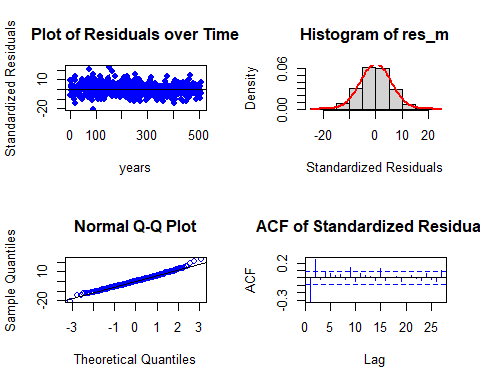
This model suggests that there is only 65.3% of data variance. Suggesting that the model explains only 65.3% of the trend. Which implies that the model performs better on the series data when compared to the former model.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(Koyck\_DLM\_mort\_chem1))

## 2 3 4 5 6 7   
## 9.21436814 -5.94911486 4.08601619 -0.75877704 2.47955863 -6.56044356   
## 8 9 10 11 12 13   
## 0.98640224 2.77128778 -2.16622184 4.08377675 1.52909262 3.51642996   
## 14 15 16 17 18 19   
## -5.29039308 9.93977670 -10.21228532 3.39046555 3.47145935 -7.18732006   
## 20 21 22 23 24 25   
## 13.24074678 -14.36074231 9.18623077 -4.05857344 0.70471267 -2.85735853   
## 26 27 28 29 30 31   
## 0.01006012 -2.83477705 -3.71123174 -1.22092691 -0.58089444 -4.99072530   
## 32 33 34 35 36 37   
## 2.98715823 -5.36269163 -4.83234266 3.09378729 -5.42633541 -3.96770828   
## 38 39 40 41 42 43   
## 4.63532748 -7.94465483 -6.43482517 -1.74418872 -5.31989236 -1.26076872   
## 44 45 46 47 48 49   
## -0.04826078 6.72177431 7.49834316 10.47897808 -8.58794435 6.37878395   
## 50 51 52 53 54 55   
## -10.53019539 0.49850679 2.23672839 4.49484958 5.52965692 0.92413154   
## 56 57 58 59 60 61   
## 7.45431750 2.57005145 1.11210238 3.52883875 -2.31295006 1.26692156   
## 62 63 64 65 66 67   
## 2.59136926 5.46410994 -5.07511947 8.86458121 -6.69363093 9.02403867   
## 68 69 70 71 72 73   
## -0.54029672 0.18853669 -3.69523140 5.30254314 -8.72577242 11.16667340   
## 74 75 76 77 78 79   
## -1.75557251 -0.43020017 -4.62684179 20.30375745 -4.48044675 -6.27224218   
## 80 81 82 83 84 85   
## 1.94635955 -0.86993070 9.73816496 -6.46917249 1.27158653 -8.38623496   
## 86 87 88 89 90 91   
## 12.67203413 -10.88441297 6.20214645 0.27617429 6.33465357 -20.19301935   
## 92 93 94 95 96 97   
## 15.27755383 -3.56906818 -0.64673653 10.43273428 1.55126385 10.21599291   
## 98 99 100 101 102 103   
## 7.88600284 4.40408609 -1.68417403 9.61282047 -6.18617881 8.08995074   
## 104 105 106 107 108 109   
## 5.41982559 -3.80902337 1.72085437 -2.07633999 -2.59912956 4.75341892   
## 110 111 112 113 114 115   
## -9.89154749 7.02970654 -0.06396549 0.58221005 2.60149084 -1.10412680   
## 116 117 118 119 120 121   
## -1.66134672 8.86040698 -6.42151375 13.45642292 -10.48127004 2.22904050   
## 122 123 124 125 126 127   
## -0.17392788 0.50334402 9.69557219 -8.78412369 6.20391232 0.22999553   
## 128 129 130 131 132 133   
## 3.52683353 3.98950449 -3.43583854 2.87433532 -10.07046325 5.40984341   
## 134 135 136 137 138 139   
## -1.34891977 2.39411683 2.49238235 -4.90404611 2.17042746 -6.57881427   
## 140 141 142 143 144 145   
## 6.53628267 -6.18532303 8.06740880 -1.15422684 -2.55585628 2.60782266   
## 146 147 148 149 150 151   
## -2.19737923 2.64092242 6.09244630 5.26801341 11.59842533 23.08021265   
## 152 153 154 155 156 157   
## 5.36269639 6.77839941 6.59716609 7.58198344 -5.17687203 -0.21532920   
## 158 159 160 161 162 163   
## 2.97990029 -1.64457916 6.21461245 -6.29815320 5.37233309 3.96109295   
## 164 165 166 167 168 169   
## -0.35962969 -7.59977170 17.99346771 -5.03055188 6.68192300 2.53784819   
## 170 171 172 173 174 175   
## -5.20164898 2.20593202 2.16612035 -4.21440384 0.47922625 13.38453581   
## 176 177 178 179 180 181   
## -6.13495064 5.20930299 -7.24124350 0.08737708 0.09317980 4.48289276   
## 182 183 184 185 186 187   
## -6.02932828 4.71832306 -4.06759907 -3.69109617 0.52895072 6.56011624   
## 188 189 190 191 192 193   
## -1.28062509 3.52954344 -5.38983704 -5.49070739 -4.40979854 1.27307287   
## 194 195 196 197 198 199   
## -8.30175940 5.41189141 -2.60995857 5.52381885 -6.77074254 6.11563176   
## 200 201 202 203 204 205   
## -4.68822660 3.83277676 9.23362436 5.90994747 -6.12453642 -5.26689311   
## 206 207 208 209 210 211   
## 0.37697139 -3.33724939 2.46937101 0.66583618 9.39759933 0.53755343   
## 212 213 214 215 216 217   
## 12.35287789 -3.63972400 3.62637172 -3.02850472 8.71042063 5.06109414   
## 218 219 220 221 222 223   
## -7.86421741 10.59843815 -4.59452514 13.66133639 -8.25448186 7.86748537   
## 224 225 226 227 228 229   
## 1.23296754 0.49559914 -3.09769121 1.03482794 10.63065267 -9.56192989   
## 230 231 232 233 234 235   
## 4.32221432 -0.77715881 -2.54420907 7.43599229 -11.40720760 4.55657243   
## 236 237 238 239 240 241   
## -0.10916408 1.68650697 0.88520556 -4.27492649 6.23686923 -5.74485818   
## 242 243 244 245 246 247   
## 2.01819706 2.12620483 -0.74273636 2.18821130 -2.92419531 -6.78964454   
## 248 249 250 251 252 253   
## 6.41655674 -12.40378409 8.70351174 -7.79607543 0.55367669 -2.24649524   
## 254 255 256 257 258 259   
## 15.38311436 1.98767554 4.58592644 4.91348081 -4.22151580 11.49436091   
## 260 261 262 263 264 265   
## 4.92130789 -8.53354312 3.92365564 -8.09108762 -1.81812185 4.05721807   
## 266 267 268 269 270 271   
## -4.73019543 3.30970972 -1.08868766 3.23616461 9.73740950 -12.19364495   
## 272 273 274 275 276 277   
## -0.57746220 -5.79522153 0.77177722 6.67748312 -2.81116486 -3.43283395   
## 278 279 280 281 282 283   
## -7.09565987 1.97361077 3.32470140 -1.70710190 -6.03036195 -2.80350073   
## 284 285 286 287 288 289   
## 2.42259806 -3.83661212 -0.39429129 -4.46919649 -2.28004099 3.90607855   
## 290 291 292 293 294 295   
## -3.25029286 -1.17991629 -5.33751597 -2.06589411 -11.18166568 -1.73461286   
## 296 297 298 299 300 301   
## -0.75627228 3.60391001 0.30833984 -9.51384167 -0.90877293 -0.45974005   
## 302 303 304 305 306 307   
## 4.05415866 -7.04779346 0.02260254 -9.09204272 7.29299529 5.02668568   
## 308 309 310 311 312 313   
## -6.81885254 -1.84522233 -4.15585879 -1.54668677 8.74369458 7.79436201   
## 314 315 316 317 318 319   
## 3.98658380 0.02996274 3.13853249 12.14491965 -5.75841356 1.18702346   
## 320 321 322 323 324 325   
## -1.13656849 -1.11844216 1.04816434 -0.24883784 1.52371716 -4.47136154   
## 326 327 328 329 330 331   
## -0.58370197 -6.09860902 -0.15481282 2.00973954 2.11218027 -0.16137822   
## 332 333 334 335 336 337   
## 6.45872202 -9.42300381 -3.74032481 -1.01686762 -5.29017701 2.60225594   
## 338 339 340 341 342 343   
## 3.01026708 -11.40711264 1.88216372 -8.46207571 3.13138148 -9.79266910   
## 344 345 346 347 348 349   
## 0.30855991 -4.03895378 0.45054246 0.08306385 -6.28355785 -2.97016989   
## 350 351 352 353 354 355   
## 2.96336250 -5.70576602 -4.77803025 -4.64103654 -2.53406695 0.68086691   
## 356 357 358 359 360 361   
## -10.90987870 3.24760686 2.51231548 -1.68091628 1.20545778 -2.18568399   
## 362 363 364 365 366 367   
## 0.16681095 -12.88004954 3.81822477 -8.41339114 -5.84305424 2.22409958   
## 368 369 370 371 372 373   
## -0.63439180 1.28055255 -1.21353735 8.03093143 1.48047518 1.46263451   
## 374 375 376 377 378 379   
## -0.41049564 5.11136490 -4.32764483 0.70392214 -8.18572724 13.49954853   
## 380 381 382 383 384 385   
## -3.15506704 -3.41508445 -5.25858517 1.03767018 -0.80624352 -4.08321395   
## 386 387 388 389 390 391   
## 0.70736924 -5.21018615 2.73994472 1.20067960 -2.96870485 -7.72162536   
## 392 393 394 395 396 397   
## -0.51382765 -4.33730999 -8.68973234 2.72115534 -9.96392776 -0.90317195   
## 398 399 400 401 402 403   
## 2.48664073 -10.16638390 1.27035459 -5.64433417 -3.84086646 -0.41901828   
## 404 405 406 407 408 409   
## 1.17694384 -14.60584958 7.20686947 -10.72238200 -0.67207197 -6.40194430   
## 410 411 412 413 414 415   
## 9.04246574 0.26102055 -2.02508939 6.94010340 5.27223449 7.17268252   
## 416 417 418 419 420 421   
## -4.33683821 1.47202359 7.57303491 -4.96442319 -2.10130094 1.10402469   
## 422 423 424 425 426 427   
## 6.46947760 -3.54957633 4.82555968 -4.29277539 3.77907364 -3.50824187   
## 428 429 430 431 432 433   
## -3.88408912 -2.36946240 3.14130488 -0.81558563 -10.49551878 6.02608745   
## 434 435 436 437 438 439   
## -2.33269972 -3.11494969 -1.33024307 -4.04568286 -0.22383858 7.09499497   
## 440 441 442 443 444 445   
## -11.95974424 -0.01786730 -2.80577560 2.25968916 -2.03623411 -2.75700759   
## 446 447 448 449 450 451   
## 2.67177156 -6.59728831 -0.43059361 7.05021364 3.98232649 -12.33203681   
## 452 453 454 455 456 457   
## -4.27357184 -7.83258203 3.78725639 -5.59080408 1.12823954 -4.47126614   
## 458 459 460 461 462 463   
## -7.74576605 9.50824906 -3.50956903 5.10584301 4.02668685 9.72615624   
## 464 465 466 467 468 469   
## -8.13850940 -4.52153623 -0.04699402 -0.18077296 14.32283276 -4.39234179   
## 470 471 472 473 474 475   
## -1.35784046 -3.35397483 2.35411994 0.73064339 -0.63459247 2.52481794   
## 476 477 478 479 480 481   
## -1.48085344 5.97062258 0.55544744 -4.00595554 2.99288716 -1.87899038   
## 482 483 484 485 486 487   
## 5.57697345 1.64689153 1.39794362 -4.55425635 3.22803865 4.95551027   
## 488 489 490 491 492 493   
## -10.47248794 2.68261110 1.25720734 -2.95710796 -5.19288949 -2.65960270   
## 494 495 496 497 498 499   
## -5.16559836 0.56529598 -1.82200653 -0.93755364 -4.68563497 3.87796566   
## 500 501 502 503 504 505   
## -0.04441795 -2.29514986 -2.44379437 -9.14650575 2.52572676 -4.30699969   
## 506 507 508   
## -2.41162959 7.32516359 -3.74267284



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99486, p-value = 0.08876

Residual Analysis for Koyck DLM:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.08876) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis cannot be rejected.

##### Koyck with Particle size

# Koyk DLM  
  
Koyck\_DLM\_mort\_part = koyckDlm(x = as.vector(x2) , y = as.vector(y))  
summary(Koyck\_DLM\_mort\_part)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -20.8011 -4.3973 -0.1927 3.7232 21.6263   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.97832 2.44335 8.177 2.39e-15 \*\*\*  
## Y.1 0.74335 0.03337 22.273 < 2e-16 \*\*\*  
## X.t 0.05835 0.03945 1.479 0.14   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.147 on 504 degrees of freedom  
## Multiple R-Squared: 0.6236, Adjusted R-squared: 0.6221   
## Wald test: 400.4 on 2 and 504 DF, p-value: < 2.2e-16   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: 77.84216 0.05834985 0.7433483

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

Wald test statistic is 400.4

R - squared is 0.6236 Adjusted

R - squared is 0.6221

Degrees of freedom - DF are (2, 504)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Koyck distributed lag model.

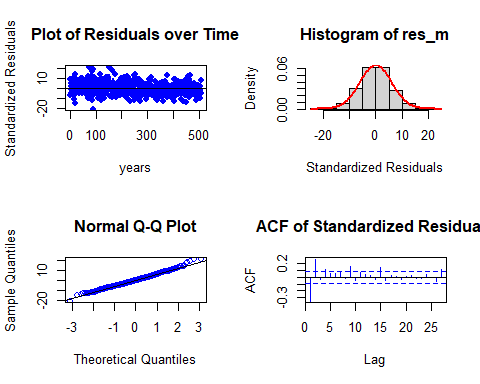
This model suggests that there is only 62.21% of data variance. Suggesting that the model explains only 62.21% of the trend. Which implies that the model performs better on the series data when compared to the former model.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(Koyck\_DLM\_mort\_part))

## 2 3 4 5 6 7   
## 9.03089412 -6.65120775 4.71075441 -0.86588011 2.18376634 -5.48576537   
## 8 9 10 11 12 13   
## 2.44466431 3.09032190 -2.01888424 5.61940825 0.67856245 4.71688805   
## 14 15 16 17 18 19   
## -6.96057045 9.87645373 -11.56371584 3.02637841 3.35428691 -6.94651296   
## 20 21 22 23 24 25   
## 14.01660468 -15.42978498 9.97202733 -4.96316374 1.32737661 -2.81533091   
## 26 27 28 29 30 31   
## 0.43391490 -3.25479757 -2.69755962 -0.24063861 0.45340819 -3.01845276   
## 32 33 34 35 36 37   
## 6.99499610 -3.01445503 -2.78311022 4.47607368 -4.76151925 3.47122208   
## 38 39 40 41 42 43   
## 6.35263577 -2.90900812 -0.52965168 0.14886345 0.72488175 4.21667368   
## 44 45 46 47 48 49   
## 1.74406375 10.51612296 9.43360242 10.28121202 -4.52362801 7.66719020   
## 50 51 52 53 54 55   
## -4.71735694 0.44417895 7.67930042 3.62811786 4.28952685 1.48304677   
## 56 57 58 59 60 61   
## 6.09197239 2.39136049 -0.78373601 4.50622521 -3.05428058 0.16153101   
## 62 63 64 65 66 67   
## 1.83001806 4.48107222 -5.94768448 8.68733252 -7.53716377 7.84970714   
## 68 69 70 71 72 73   
## -1.99704145 -0.86700784 -3.87331350 4.51486714 -9.75658403 11.86420012   
## 74 75 76 77 78 79   
## -2.82648206 -1.31032529 -4.79357587 21.44318943 -6.08816718 -7.83648388   
## 80 81 82 83 84 85   
## 3.07941476 -1.75201482 12.07366256 -7.89895693 2.24583678 -7.66143465   
## 86 87 88 89 90 91   
## 17.53765613 -12.57755952 8.51146185 -0.19938789 9.45160455 -20.80112197   
## 92 93 94 95 96 97   
## 18.97538762 -2.50781620 -0.36487578 11.03512597 3.57329579 9.23961449   
## 98 99 100 101 102 103   
## 8.11495470 6.04039134 1.57073916 8.27233998 -5.47975920 8.58227759   
## 104 105 106 107 108 109   
## 6.47547562 -3.85998483 1.72083453 -3.05775829 -2.96492291 3.63063824   
## 110 111 112 113 114 115   
## -10.04503809 7.74028213 0.31356606 0.23906348 2.67546905 -1.48773432   
## 116 117 118 119 120 121   
## -3.16566537 9.38757483 -8.24081104 13.84486046 -12.36760388 2.05037168   
## 122 123 124 125 126 127   
## -0.78941167 0.02520035 8.99153675 -9.50818847 5.91209775 -0.55524759   
## 128 129 130 131 132 133   
## 4.70470139 2.83523774 -5.03316369 2.63005435 -10.57668494 5.85125070   
## 134 135 136 137 138 139   
## -2.04336594 2.56830595 2.82914935 -5.08293345 2.89467687 -5.98551964   
## 140 141 142 143 144 145   
## 7.81955485 -5.61200633 10.82512188 -1.82516129 -1.93243690 4.93403124   
## 146 147 148 149 150 151   
## 1.27246875 1.60664544 6.92041283 8.67367011 10.82354955 21.62627580   
## 152 153 154 155 156 157   
## 4.74714968 3.96748715 5.57321240 6.05323918 -7.81841863 -0.39557090   
## 158 159 160 161 162 163   
## 2.22084814 -2.17192253 5.20612659 -7.06071646 4.87051456 3.01115518   
## 164 165 166 167 168 169   
## -0.69344137 -8.90312899 18.14322473 -7.34203611 5.73787401 0.94076853   
## 170 171 172 173 174 175   
## -7.12120833 1.78102185 0.85498945 -4.55980786 -0.19597211 14.04402169   
## 176 177 178 179 180 181   
## -8.20839541 4.23479374 -9.26757281 -0.15352684 -0.11366897 3.73321131   
## 182 183 184 185 186 187   
## -7.24003252 5.18553462 -4.68395027 -4.75255358 -0.19272562 5.94822438   
## 188 189 190 191 192 193   
## -1.95450783 5.33483946 -6.76166517 -2.84355383 -2.95517074 4.35575318   
## 194 195 196 197 198 199   
## -7.18525219 7.97684011 -2.86713825 5.76294566 -5.54741540 9.69751184   
## 200 201 202 203 204 205   
## -3.11070388 4.86718477 8.17672900 3.28841782 -5.67934388 -4.40353635   
## 206 207 208 209 210 211   
## 2.03963845 -1.92293123 3.29570042 0.72829335 9.06774260 -0.38414660   
## 212 213 214 215 216 217   
## 11.15788691 -4.03384428 1.26861684 -4.70326749 7.87292623 3.55534228   
## 218 219 220 221 222 223   
## -9.52648666 9.14247231 -6.70784763 12.23949424 -10.62397455 7.23060393   
## 224 225 226 227 228 229   
## -0.58738153 -1.20184515 -4.24344112 0.56914679 10.58858853 -11.97545396   
## 230 231 232 233 234 235   
## 3.95411496 -1.38094966 -3.57617393 6.83215212 -13.15671514 4.55341536   
## 236 237 238 239 240 241   
## -0.99106993 1.84711306 1.13779888 -4.91714354 7.45066387 -6.74455422   
## 242 243 244 245 246 247   
## 2.57874796 2.70285220 1.52288289 1.96776762 -2.35619155 -4.75330435   
## 248 249 250 251 252 253   
## 9.43261762 -11.96410233 12.98459559 -6.68438460 3.70299616 2.66774274   
## 254 255 256 257 258 259   
## 13.64004074 2.63216671 4.70951380 9.31341813 -2.82765455 10.49531762   
## 260 261 262 263 264 265   
## 4.71536057 -9.28586979 3.98364891 -6.27754971 -2.17959828 4.92329968   
## 266 267 268 269 270 271   
## -4.92701550 3.00984255 -0.72954475 2.48224629 9.42494011 -13.63616315   
## 272 273 274 275 276 277   
## -0.94572191 -5.62721394 0.42539392 6.84804700 -4.18334588 -3.90164834   
## 278 279 280 281 282 283   
## -7.70630032 2.42106166 3.71327840 -2.00592325 -6.84956919 -3.64740807   
## 284 285 286 287 288 289   
## 2.06065749 -4.57758030 -0.05870030 -5.63894867 -2.03359509 4.25928241   
## 290 291 292 293 294 295   
## -3.70545646 -1.53321817 -4.07710277 -1.82435583 -10.87958174 -1.10270179   
## 296 297 298 299 300 301   
## 1.30796253 3.62808410 2.72750134 -8.58007976 1.73482870 0.56860847   
## 302 303 304 305 306 307   
## 3.80940477 -5.19091479 0.16133518 -6.31820121 8.74344042 4.62153663   
## 308 309 310 311 312 313   
## -5.69722586 0.71410372 -4.00003675 1.23137221 7.48972663 7.58886096   
## 314 315 316 317 318 319   
## 2.92478000 -1.02066889 1.70139318 11.53008412 -6.91913624 0.37304088   
## 320 321 322 323 324 325   
## -2.18293010 -2.44587840 0.38643847 -1.17841419 0.67949770 -5.75666227   
## 326 327 328 329 330 331   
## -0.85056946 -6.87381908 -0.67910340 2.12181650 1.76876218 -0.73508324   
## 332 333 334 335 336 337   
## 6.69870865 -11.13852481 -3.96717278 -1.68985478 -5.46990380 2.43073217   
## 338 339 340 341 342 343   
## 2.98625112 -11.70396589 2.32969554 -7.50714799 3.66495704 -9.20240839   
## 344 345 346 347 348 349   
## 0.75965618 -3.96551734 0.59366541 1.59847973 -6.56989306 -2.70629448   
## 350 351 352 353 354 355   
## 4.60235065 -1.91080817 -4.46972818 -1.39980947 -1.84311292 3.37154977   
## 356 357 358 359 360 361   
## -8.82846745 5.90279110 3.39693857 -2.11055314 0.40197699 -1.12391942   
## 362 363 364 365 366 367   
## 0.27002210 -13.14591052 5.15110339 -7.20886580 -4.67997090 1.76272055   
## 368 369 370 371 372 373   
## -0.71981806 0.85638225 -1.92216305 7.14172983 0.23743741 0.05718406   
## 374 375 376 377 378 379   
## -2.10984999 4.88553204 -6.20189285 -0.36458150 -9.05723551 13.17022454   
## 380 381 382 383 384 385   
## -4.54618896 -4.64562412 -6.21733393 0.72332951 -2.40145315 -4.84243334   
## 386 387 388 389 390 391   
## 0.13003599 -6.10235044 2.13112772 1.00978577 -4.21125274 -8.31959469   
## 392 393 394 395 396 397   
## -0.19349006 -4.88341803 -8.77903255 3.22354375 -10.09833025 -0.41737393   
## 398 399 400 401 402 403   
## 3.25680152 -9.39197642 1.45661207 -5.30734018 -2.23697010 0.89500205   
## 404 405 406 407 408 409   
## 3.27619303 -13.55510750 9.44508252 -8.13371254 1.18242780 -5.64354364   
## 410 411 412 413 414 415   
## 8.59150617 -0.02267756 -1.66977938 7.00800974 4.68942241 7.02184611   
## 416 417 418 419 420 421   
## -5.39174061 0.39319352 6.82033030 -5.99166364 -3.05431620 0.52563856   
## 422 423 424 425 426 427   
## 6.27201021 -4.39100023 3.77220205 -5.42342826 2.54017873 -4.42608199   
## 428 429 430 431 432 433   
## -4.78648019 -3.06835451 2.90184433 -1.90722004 -11.24101125 5.72994668   
## 434 435 436 437 438 439   
## -3.72413409 -3.69508065 -1.81989654 -4.90496776 -1.33163526 6.07721366   
## 440 441 442 443 444 445   
## -13.64569406 -0.89426770 -4.29316133 1.01997444 -3.40493882 -3.84499262   
## 446 447 448 449 450 451   
## 1.41417863 -7.31891034 -1.19294637 8.86723324 2.35626132 -14.69335989   
## 452 453 454 455 456 457   
## -3.64889574 -7.45979819 4.08408684 -4.02948422 1.12972286 -2.63495933   
## 458 459 460 461 462 463   
## -7.62918697 11.88795314 -3.51941672 7.31557310 4.78935507 9.03198719   
## 464 465 466 467 468 469   
## -8.66725869 -5.70139827 0.83031216 -0.27088578 13.22647082 -5.11004863   
## 470 471 472 473 474 475   
## -2.19911876 -4.47627117 1.49348856 -0.85797945 -2.17980717 1.52739458   
## 476 477 478 479 480 481   
## -3.00611223 4.38354032 -0.84753214 -5.77031917 1.76043490 -3.98653633   
## 482 483 484 485 486 487   
## 4.53539425 -0.99463705 -0.30151442 -7.00189183 2.47432558 2.67404850   
## 488 489 490 491 492 493   
## -12.57464683 1.83261231 -0.47286006 -4.90810033 -7.02386678 -4.15895257   
## 494 495 496 497 498 499   
## -6.54236420 -0.50356746 -3.57244490 -1.95263110 -6.10663490 2.74789134   
## 500 501 502 503 504 505   
## 0.23722897 -3.20016967 -4.21576849 -9.82722478 1.98469789 -4.82530848   
## 506 507 508   
## -2.49557004 7.72418440 -4.61924500



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99482, p-value = 0.08584

Residual Analysis for Koyck DLM:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.08584) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis cannot be rejected.

So far Koyck with Chemical emission 1 is the best model but let us fit ardlDlm model to check whether it fits better than Koyck model or not.

#### Autoregressive distributed lag model

This again takes multiple predictors.

Getting p and q values for finite distributed lag model based on MASE values.

for (i in 1:5){  
 for(j in 1:5){  
 model\_4 = ardlDlm(formula = y ~ x2 + x4, data = data.frame(v\_data\_TS), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_4$model), "BIC = ", BIC(model\_4$model), "MASE =", MASE(model\_4)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = 3232.773 BIC = 3262.373 MASE = 0.8626722   
## p = 1 q = 2 AIC = 3117.872 BIC = 3151.685 MASE = 0.771343   
## p = 1 q = 3 AIC = 3113.495 BIC = 3151.516 MASE = 0.7702783   
## p = 1 q = 4 AIC = 3108.353 BIC = 3150.579 MASE = 0.769892   
## p = 1 q = 5 AIC = 3104.428 BIC = 3150.855 MASE = 0.7663933   
## p = 2 q = 1 AIC = 3209.942 BIC = 3247.981 MASE = 0.8431273   
## p = 2 q = 2 AIC = 3115.262 BIC = 3157.527 MASE = 0.7656794   
## p = 2 q = 3 AIC = 3110.805 BIC = 3157.275 MASE = 0.7642383   
## p = 2 q = 4 AIC = 3106.131 BIC = 3156.801 MASE = 0.7641585   
## p = 2 q = 5 AIC = 3102.548 BIC = 3157.416 MASE = 0.7620808   
## p = 3 q = 1 AIC = 3206.975 BIC = 3253.445 MASE = 0.8416035   
## p = 3 q = 2 AIC = 3113.362 BIC = 3164.057 MASE = 0.7654371   
## p = 3 q = 3 AIC = 3114.739 BIC = 3169.658 MASE = 0.7644454   
## p = 3 q = 4 AIC = 3110.085 BIC = 3169.202 MASE = 0.7643283   
## p = 3 q = 5 AIC = 3106.499 BIC = 3169.807 MASE = 0.7621898   
## p = 4 q = 1 AIC = 3171.64 BIC = 3226.534 MASE = 0.8083768   
## p = 4 q = 2 AIC = 3093.839 BIC = 3152.955 MASE = 0.7477227   
## p = 4 q = 3 AIC = 3095.396 BIC = 3158.735 MASE = 0.7478399   
## p = 4 q = 4 AIC = 3097.377 BIC = 3164.938 MASE = 0.747751   
## p = 4 q = 5 AIC = 3093.631 BIC = 3165.381 MASE = 0.7456829   
## p = 5 q = 1 AIC = 3167.579 BIC = 3230.887 MASE = 0.8043549   
## p = 5 q = 2 AIC = 3087.036 BIC = 3154.565 MASE = 0.7391588   
## p = 5 q = 3 AIC = 3088.377 BIC = 3160.128 MASE = 0.7385057   
## p = 5 q = 4 AIC = 3090.327 BIC = 3166.297 MASE = 0.7386083   
## p = 5 q = 5 AIC = 3092.255 BIC = 3172.446 MASE = 0.7383519

(p, q) = (5, 2); (5, 3) has the least AIC, BIC and MASE scores.

Let us fit (5, 2)

# ARDLM model  
AR\_DLM\_mort\_52 = ardlDlm(formula = y ~ x2 + x4, data = data.frame(v\_data\_TS), p = 5, q = 2)  
summary(AR\_DLM\_mort\_52)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 508  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -14.2494 -3.5098 -0.2163 3.1941 23.3525   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.6211638 2.9621009 6.624 9.25e-11 \*\*\*  
## x2.t 0.0006556 0.1745611 0.004 0.997005   
## x2.1 0.4018107 0.1739867 2.309 0.021336 \*   
## x2.2 0.3500628 0.1810555 1.933 0.053758 .   
## x2.3 0.0946475 0.1829440 0.517 0.605141   
## x2.4 0.3725505 0.1809623 2.059 0.040051 \*   
## x2.5 -0.3662787 0.1813384 -2.020 0.043944 \*   
## x4.t 0.1445148 0.0395899 3.650 0.000290 \*\*\*  
## x4.1 -0.1515187 0.0398071 -3.806 0.000159 \*\*\*  
## x4.2 -0.0853990 0.0401870 -2.125 0.034085 \*   
## x4.3 -0.0327749 0.0399252 -0.821 0.412100   
## x4.4 0.0092831 0.0398760 0.233 0.816016   
## x4.5 0.0918108 0.0395403 2.322 0.020646 \*   
## y.1 0.3438820 0.0418985 8.208 2.03e-15 \*\*\*  
## y.2 0.3712665 0.0397978 9.329 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.119 on 488 degrees of freedom  
## Multiple R-squared: 0.7451, Adjusted R-squared: 0.7378   
## F-statistic: 101.9 on 14 and 488 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model.**

**HA: The data fits the Autoregressive distributed lag model.**

**Interpretations:**

F - statistic is 101.9

R - squared is 0.7451

Adjusted R - squared is 0.7378

Degrees of freedom - DF are (14, 488)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Autoregressive distributed lag model.

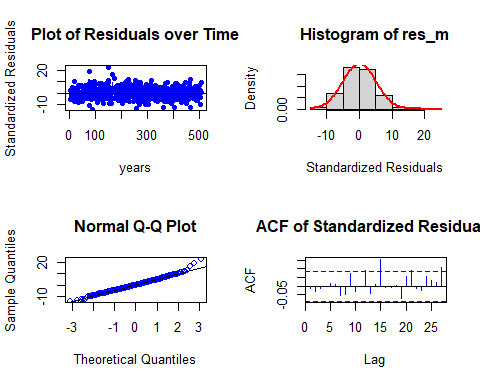
This model suggests that there is only 73.78% of data variance. Suggesting that the model explains only 73.78% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(AR\_DLM\_mort\_52))

## Time Series:  
## Start = 6   
## End = 508   
## Frequency = 1   
## 6 7 8 9 10 11   
## 2.18125931 -5.60652325 -2.02357065 2.57044010 -5.69639970 0.56169951   
## 12 13 14 15 16 17   
## 2.91124606 1.37212879 -0.28115197 4.98076918 -6.27270419 -2.96589274   
## 18 19 20 21 22 23   
## 3.26696985 -3.98272547 9.95729405 -7.21933582 3.82166102 0.44677960   
## 24 25 26 27 28 29   
## -2.07014149 -2.09317167 0.28955780 -1.15526010 -4.30462518 -1.05369246   
## 30 31 32 33 34 35   
## -1.37842835 -3.77410679 3.47223279 -0.13611484 -5.40130159 -0.58149309   
## 36 37 38 39 40 41   
## -8.18736331 -3.99646189 6.43987433 -6.51537241 -6.51340863 -7.07637805   
## 42 43 44 45 46 47   
## -5.01935138 -3.04393017 -4.39097492 5.57987175 6.98208940 6.49161606   
## 48 49 50 51 52 53   
## -0.86308154 0.13486443 -7.27878136 -7.87860344 -1.83479258 4.73260532   
## 54 55 56 57 58 59   
## -3.03173517 0.88911087 0.50811098 3.64932169 -0.06235321 1.01850161   
## 60 61 62 63 64 65   
## -0.68973972 -2.24955085 1.62055876 2.00674587 -4.90469401 6.66668837   
## 66 67 68 69 70 71   
## -3.38859084 6.08425164 0.44382899 -1.89671137 -5.07649730 7.03260151   
## 72 73 74 75 76 77   
## -4.60166649 7.59324626 3.15362205 -1.34087084 -4.30044518 19.40455530   
## 78 79 80 81 82 83   
## 3.63608147 -10.95775397 -1.76935404 0.07126750 8.92549264 0.14900717   
## 84 85 86 87 88 89   
## -4.33350771 -5.50252118 9.59620055 -3.69120983 -0.22690715 5.42376765   
## 90 91 92 93 94 95   
## 3.04598498 -14.24942214 6.38706743 3.96596228 -5.12990834 7.73178338   
## 96 97 98 99 100 101   
## 4.05975520 10.64704635 6.25552423 4.83118092 -1.99216672 7.60720104   
## 102 103 104 105 106 107   
## -4.43635996 0.33404171 2.19612166 -4.29273004 -4.77608120 -3.19672750   
## 108 109 110 111 112 113   
## -7.20939569 0.93920764 -8.19692757 3.14855403 3.13993253 0.21395298   
## 114 115 116 117 118 119   
## 0.72164789 -1.33246798 -3.12789278 7.64672479 -2.20327618 8.80678240   
## 120 121 122 123 124 125   
## -5.95122144 -3.51869312 2.01000803 0.85471058 8.76209996 -5.37102054   
## 126 127 128 129 130 131   
## 2.96503616 3.91619612 3.08121846 5.56348217 -3.17309372 1.22401988   
## 132 133 134 135 136 137   
## -9.57872490 2.10519629 3.16664916 -0.51043691 4.56498004 -2.67173352   
## 138 139 140 141 142 143   
## 2.34964395 -5.14122957 3.78157058 -2.50069237 7.76472788 4.11574158   
## 144 145 146 147 148 149   
## -3.58110760 1.97731501 -0.87142051 0.74810308 3.24247978 9.51927061   
## 150 151 152 153 154 155   
## 10.95927945 23.35246733 9.71457935 0.43051569 3.18439889 4.75178097   
## 156 157 158 159 160 161   
## -9.36956127 -5.93086357 -0.61693381 -3.84391463 4.01093380 -7.32911311   
## 162 163 164 165 166 167   
## 2.50257631 4.35719428 -0.42123707 -8.97945107 16.33031088 1.35168903   
## 168 169 170 171 172 173   
## 2.21384488 4.48473146 -5.11916968 0.57113729 3.26670539 -5.16254832   
## 174 175 176 177 178 179   
## 0.81793608 13.38835606 0.20829419 0.82155257 -5.05278487 -3.50088383   
## 180 181 182 183 184 185   
## 0.80987895 5.57545938 -3.31916528 2.09284392 -0.37235296 -4.97468962   
## 186 187 188 189 190 191   
## -0.25569138 6.90410489 1.81476257 5.40750255 -1.99022771 -6.80417384   
## 192 193 194 195 196 197   
## -2.53223447 1.27943959 -3.85542693 0.05675061 2.12110175 1.59339620   
## 198 199 200 201 202 203   
## -4.55464700 4.35239686 0.24360862 2.65988782 7.55099977 5.30533567   
## 204 205 206 207 208 209   
## -8.34144350 -9.33491188 -1.14093274 -0.07020637 1.38002064 2.06170738   
## 210 211 212 213 214 215   
## 7.08162741 1.18597614 8.62107028 -2.60638078 1.16366276 -4.76411300   
## 216 217 218 219 220 221   
## 4.99433012 4.41864239 -6.99181838 6.16227405 0.14366735 10.56176745   
## 222 223 224 225 226 227   
## -4.20740544 2.28980098 4.84651262 -0.21631732 -3.66431338 0.18035953   
## 228 229 230 231 232 233   
## 12.36007825 -3.87840130 -0.65625413 0.91991437 -3.01611909 8.73611868   
## 234 235 236 237 238 239   
## -7.88439169 -1.07004588 2.02010545 2.87711851 3.90347372 -1.84517149   
## 240 241 242 243 244 245   
## 5.15195215 -2.60827541 1.06408057 2.01611644 1.19803207 4.30334649   
## 246 247 248 249 250 251   
## -3.82862197 -7.32261446 4.43779423 -7.37388872 7.49991624 -5.52474629   
## 252 253 254 255 256 257   
## -1.45770756 -0.86140742 11.59820205 3.86626363 3.36064737 2.88582073   
## 258 259 260 261 262 263   
## -0.88032456 3.85131549 6.90804754 -14.00475307 -4.68216787 -7.75985826   
## 264 265 266 267 268 269   
## -3.99136468 3.41680548 -4.56453174 -1.89364977 -0.66733720 2.62303766   
## 270 271 272 273 274 275   
## 9.91838770 -9.88236492 -8.26515700 -4.53734794 0.87575511 9.59557442   
## 276 277 278 279 280 281   
## -0.30029553 -5.44580761 -7.91328190 1.88680522 5.54158714 1.16785221   
## 282 283 284 285 286 287   
## -6.83170899 -3.16980736 2.09477623 -1.04509023 -0.20075464 -1.31735134   
## 288 289 290 291 292 293   
## -2.87652392 4.74124607 -0.76989376 0.46943031 -4.64566268 -1.44110458   
## 294 295 296 297 298 299   
## -10.07702278 -1.44401676 -0.72916282 7.34561808 1.94505223 -5.92158058   
## 300 301 302 303 304 305   
## -2.21408896 0.70951536 5.04598377 -2.77257062 -1.36446604 -9.79110572   
## 306 307 308 309 310 311   
## 5.36253843 6.58899928 -4.28620161 -3.84162373 -5.47742876 -3.87173748   
## 312 313 314 315 316 317   
## 8.12325153 5.20773339 4.57899728 -4.47632181 2.56922245 10.88306678   
## 318 319 320 321 322 323   
## -3.59958304 -2.54187011 -1.65270768 -4.17824372 -0.22283688 -0.64058921   
## 324 325 326 327 328 329   
## 1.34974866 -3.08343994 -2.73251235 -3.15595662 -2.07362650 4.01481713   
## 330 331 332 333 334 335   
## 2.94441388 1.04347460 6.52416990 -4.92659506 -7.02002802 1.05115922   
## 336 337 338 339 340 341   
## -7.03280029 3.36457336 4.73179503 -7.92983866 -0.92750387 -6.98710301   
## 342 343 344 345 346 347   
## 2.42686243 -5.98531907 0.05583307 -3.11337634 0.16949626 0.26213548   
## 348 349 350 351 352 353   
## -1.89430755 -1.85009800 4.82987307 -0.24483711 -5.71752471 -4.48768719   
## 354 355 356 357 358 359   
## -2.95288753 0.52967914 -5.78297329 0.35869976 4.66948306 -0.98228587   
## 360 361 362 363 364 365   
## -0.62089762 -4.41306583 -0.94691460 -13.11568139 -0.18062502 -3.84985370   
## 366 367 368 369 370 371   
## -7.59587973 1.82874552 -1.74821414 -0.03252803 -1.85180450 9.02806984   
## 372 373 374 375 376 377   
## 3.25242948 0.20935248 -0.66797186 4.12749835 -1.56466132 -2.19537883   
## 378 379 380 381 382 383   
## -7.65022216 9.08990976 3.74814010 -6.52311128 -4.47541220 0.56277604   
## 384 385 386 387 388 389   
## -0.85714579 -1.05737901 1.14565683 -2.06728123 1.83373265 3.61125975   
## 390 391 392 393 394 395   
## 1.05376156 -7.42824531 -0.25808481 -2.98161975 -9.21390459 0.67294358   
## 396 397 398 399 400 401   
## -4.84096238 -2.83606975 5.77373457 -7.28466188 -0.91770881 0.24061769   
## 402 403 404 405 406 407   
## -1.55892882 2.66229428 1.90064716 -12.73007530 3.77449939 -5.86060873   
## 408 409 410 411 412 413   
## -4.15935391 -3.09557836 8.36132616 2.42181956 -4.59132532 4.18142128   
## 414 415 416 417 418 419   
## 8.20231753 6.46566379 -1.58449546 -0.92990411 4.71748183 -4.47516332   
## 420 421 422 423 424 425   
## -5.97093638 -1.18874227 7.07120624 -1.96766655 2.75520313 -3.72481649   
## 426 427 428 429 430 431   
## 0.52139622 -2.71463525 -3.99365885 -3.80502414 4.47466951 2.02152076   
## 432 433 434 435 436 437   
## -8.73314891 3.92415636 -0.69249786 -1.66948390 -0.49783841 -0.55900947   
## 438 439 440 441 442 443   
## -2.00408885 6.61038385 -6.57196669 -1.80344373 1.43315441 4.09410742   
## 444 445 446 447 448 449   
## -0.54215882 -2.14216109 3.71961574 -1.95830112 -1.11945360 9.30559155   
## 450 451 452 453 454 455   
## 2.59138555 -9.36631547 -6.11654218 -0.95384781 3.49178118 -2.35329323   
## 456 457 458 459 460 461   
## -0.76632112 -2.21054921 -5.52973374 3.20379100 3.68868693 2.26376554   
## 462 463 464 465 466 467   
## 7.14885809 8.70263919 -2.86038276 -12.91976116 -0.71331752 -2.14883513   
## 468 469 470 471 472 473   
## 12.57365600 -4.14082913 1.19756887 -2.78615093 0.78190724 -2.52391531   
## 474 475 476 477 478 479   
## 0.95426030 3.98245230 -0.80617398 2.35228029 2.08539788 -4.62503380   
## 480 481 482 483 484 485   
## 3.13361869 -0.01398302 4.70232614 3.78357674 2.87337093 -3.82409649   
## 486 487 488 489 490 491   
## 1.94898780 5.71460524 -6.73675540 0.43798225 4.43915638 -1.56628160   
## 492 493 494 495 496 497   
## -6.18320648 -1.65615910 -2.88525671 1.54833641 2.89787632 -1.92649252   
## 498 499 500 501 502 503   
## -0.48262528 1.50262756 6.98569419 0.31160167 -2.19689011 -10.28253328   
## 504 505 506 507 508   
## 2.13032081 -0.47552892 -3.61238431 9.91230599 0.68775341



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.98986, p-value = 0.001518

Residual Analysis for AR\_DLM\_mort\_52:

1. The data points are below the line at both the start and end of the trend. Randomness is not seen here.
2. From normal distribution curve, the distribution is almost symmetric. This suggests a good fit with a very few data falling outside the normal curve indicating Kurtosis.
3. The data at the tails is deviated but is normal for most part of the line suggesting normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.001518) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Let us fit (5, 3)

# ARDLM model  
AR\_DLM\_mort\_53 = ardlDlm(formula = y ~ x2 + x4, data = data.frame(v\_data\_TS), p = 5, q = 3)  
summary(AR\_DLM\_mort\_53)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 508  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -14.158 -3.441 -0.156 3.125 23.548   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.040440 3.051143 6.240 9.49e-10 \*\*\*  
## x2.t -0.009554 0.175093 -0.055 0.956506   
## x2.1 0.393798 0.174340 2.259 0.024338 \*   
## x2.2 0.365912 0.182207 2.008 0.045171 \*   
## x2.3 0.102061 0.183247 0.557 0.577812   
## x2.4 0.366179 0.181205 2.021 0.043847 \*   
## x2.5 -0.379162 0.182122 -2.082 0.037872 \*   
## x4.t 0.147885 0.039829 3.713 0.000228 \*\*\*  
## x4.1 -0.146565 0.040302 -3.637 0.000306 \*\*\*  
## x4.2 -0.087017 0.040253 -2.162 0.031124 \*   
## x4.3 -0.039041 0.040704 -0.959 0.337958   
## x4.4 0.010466 0.039918 0.262 0.793295   
## x4.5 0.092896 0.039578 2.347 0.019318 \*   
## y.1 0.329671 0.045536 7.240 1.76e-12 \*\*\*  
## y.2 0.357184 0.043544 8.203 2.11e-15 \*\*\*  
## y.3 0.034651 0.043395 0.799 0.424967   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.121 on 487 degrees of freedom  
## Multiple R-squared: 0.7454, Adjusted R-squared: 0.7376   
## F-statistic: 95.07 on 15 and 487 DF, p-value: < 2.2e-16

Hypotheses : H0 : The data doesn′t fit the Autoregressive distributed lag model. HA : The data fits the Autoregressive distributed lag model.

Interpretations: F - statistic is 95.07 R - squared is 0.7454 Adjusted R - squared is 0.7376 Degrees of freedom - DF are (15, 487) p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Autoregressive distributed lag model.

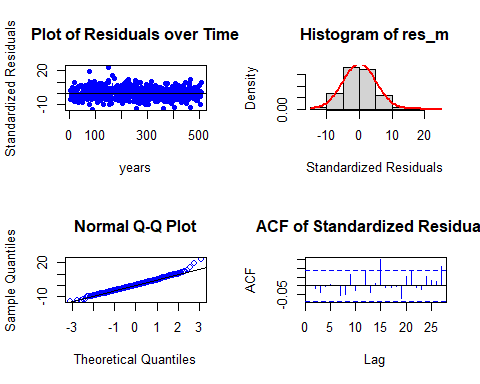
This model suggests that there is only 73.76% of data variance. Suggesting that the model explains only 73.76% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(AR\_DLM\_mort\_53))

## Time Series:  
## Start = 6   
## End = 508   
## Frequency = 1   
## 6 7 8 9 10   
## 2.198622783 -5.653078976 -2.010309452 2.399898622 -5.525722896   
## 11 12 13 14 15   
## 0.551294106 2.818799923 1.498689813 -0.328021501 4.934136829   
## 16 17 18 19 20   
## -6.271450543 -2.928841557 2.928533180 -3.819219609 9.891707083   
## 21 22 23 24 25   
## -7.274078949 3.979806659 0.099804594 -1.773896169 -2.300881659   
## 26 27 28 29 30   
## 0.244475440 -1.183561747 -4.272985999 -1.116122186 -1.377794599   
## 31 32 33 34 35   
## -3.705925636 3.444585061 -0.076230973 -5.158293922 -0.672579390   
## 36 37 38 39 40   
## -8.160598381 -3.836930892 6.391070909 -6.323248761 -6.382996017   
## 41 42 43 44 45   
## -7.324956277 -4.880574352 -2.959941923 -4.341105232 5.776024437   
## 46 47 48 49 50   
## 7.086674376 6.758227699 -0.677922035 0.074928971 -7.579441309   
## 51 52 53 54 55   
## -7.995910653 -2.299767717 4.859438148 -2.869905368 0.971491826   
## 56 57 58 59 60   
## 0.414446844 3.602722888 -0.086371716 0.776741857 -0.797807728   
## 61 62 63 64 65   
## -2.352391416 1.448207524 1.978076613 -4.862081083 6.610297414   
## 66 67 68 69 70   
## -3.527355507 6.218143399 0.257105113 -1.693751881 -5.311340617   
## 71 72 73 74 75   
## 6.869240978 -4.634490328 7.699048321 2.987953720 -0.984676826   
## 76 77 78 79 80   
## -4.417594271 19.291876662 3.824166321 -10.415954981 -2.282491911   
## 81 82 83 84 85   
## -0.183576907 9.019721965 0.264740125 -4.180360892 -5.761315565   
## 86 87 88 89 90   
## 9.480882616 -3.678600008 0.003836582 5.083561655 3.301452301   
## 91 92 93 94 95   
## -14.157485289 6.180459808 3.626369034 -4.590968611 7.526010043   
## 96 97 98 99 100   
## 4.124844471 10.915954432 6.306170924 4.913956079 -2.168832813   
## 101 102 103 104 105   
## 7.406918574 -4.666877969 0.290975844 1.889725007 -4.169278025   
## 106 107 108 109 110   
## -5.016611754 -3.539019990 -7.330167620 0.694235690 -8.276854610   
## 111 112 113 114 115   
## 3.167416805 2.953833931 0.499160754 0.692449767 -1.324678069   
## 116 117 118 119 120   
## -3.134850551 7.548052861 -2.113836611 8.975800310 -6.051350067   
## 121 122 123 124 125   
## -3.386391241 1.604436373 0.996771393 8.778691263 -5.245014112   
## 126 127 128 129 130   
## 3.057728760 3.688257590 3.303026719 5.520129608 -3.080840524   
## 131 132 133 134 135   
## 1.208944986 -9.763159703 2.052558297 2.979121081 -0.244282953   
## 136 137 138 139 140   
## 4.522561869 -2.589186185 2.407203530 -5.259067699 3.770963222   
## 141 142 143 144 145   
## -2.606604803 7.933307126 4.047413538 -3.284926431 1.847049149   
## 146 147 148 149 150   
## -0.917341027 0.832096809 3.109408702 9.583486509 11.194666771   
## 151 152 153 154 155   
## 23.548338814 9.940559525 0.673620262 2.827311803 4.351993215   
## 156 157 158 159 160   
## -9.674850411 -6.226717901 -1.210602699 -4.038763668 3.926164179   
## 161 162 163 164 165   
## -7.442737578 2.489217795 4.133902996 -0.212492730 -9.057348650   
## 166 167 168 169 170   
## 16.086161115 1.394979539 2.639604822 4.142474655 -4.995561525   
## 171 172 173 174 175   
## 0.416196192 3.047796641 -5.079831815 0.759786308 13.239959329   
## 176 177 178 179 180   
## 0.484837331 1.034829724 -5.317523930 -3.494865792 0.556460809   
## 181 182 183 184 185   
## 5.621730453 -3.192381170 2.167780897 -0.516898397 -4.815006296   
## 186 187 188 189 190   
## -0.394638560 6.859169757 2.025359670 5.622415569 -2.027908951   
## 191 192 193 194 195   
## -6.805863159 -2.815315132 1.177802522 -3.705446503 0.113710505   
## 196 197 198 199 200   
## 2.082106093 1.905763405 -4.482899742 4.370474399 0.136951037   
## 201 202 203 204 205   
## 2.834141866 7.384106306 5.524003140 -8.085983004 -9.512968188   
## 206 207 208 209 210   
## -1.708382863 -0.296347754 1.365829738 2.079687282 7.240011858   
## 211 212 213 214 215   
## 1.340973175 8.810093580 -2.658428962 1.213701492 -5.079734561   
## 216 217 218 219 220   
## 4.930403233 4.214579170 -6.800231645 5.942675045 -0.122362418   
## 221 222 223 224 225   
## 10.770418773 -4.263865547 2.448307370 4.472617611 -0.013690963   
## 226 227 228 229 230   
## -3.757275075 -0.009400148 12.232933585 -3.656895237 -0.511201750   
## 231 232 233 234 235   
## 0.567261766 -2.908604796 8.675565582 -7.782165234 -0.961056347   
## 236 237 238 239 240   
## 1.689810506 3.103474473 3.968717804 -1.735081895 5.132157321   
## 241 242 243 244 245   
## -2.637089980 1.238434767 1.925431754 1.305106185 4.353051676   
## 246 247 248 249 250   
## -3.749723499 -7.355306306 4.162917031 -7.362846191 7.651251008   
## 251 252 253 254 255   
## -5.694003238 -1.118512893 -1.031053044 11.738283247 4.023097276   
## 256 257 258 259 260   
## 3.800726831 2.806395675 -0.824916239 3.567074608 6.722697573   
## 261 262 263 264 265   
## -13.744644689 -4.939357295 -8.292616670 -4.110210695 3.187491443   
## 266 267 268 269 270   
## -4.417708402 -1.841735088 -0.730404424 2.765656550 9.940554830   
## 271 272 273 274 275   
## -9.627346990 -8.294515840 -5.077092869 0.857821111 9.619026009   
## 276 277 278 279 280   
## 0.020357064 -5.242642231 -8.082615401 1.770145727 5.531917183   
## 281 282 283 284 285   
## 1.460731782 -6.706338140 -3.264253663 1.962027112 -0.909473838   
## 286 287 288 289 290   
## -0.062191267 -1.310344377 -2.789493456 4.715323427 -0.661985080   
## 291 292 293 294 295   
## 0.691730321 -4.684257016 -1.467821786 -10.173284311 -1.435462445   
## 296 297 298 299 300   
## -0.838943110 7.613485906 2.178758158 -5.658273998 -2.338939348   
## 301 302 303 304 305   
## 0.511795459 5.083348829 -2.611995759 -1.169142165 -9.952874881   
## 306 307 308 309 310   
## 5.261949211 6.467369947 -3.810898186 -3.849326625 -5.692754460   
## 311 312 313 314 315   
## -3.903943826 8.022737691 5.471060748 4.997363739 -4.459766304   
## 316 317 318 319 320   
## 2.471681968 10.682218556 -3.523645614 -2.524914726 -2.056323726   
## 321 322 323 324 325   
## -4.262875403 -0.402630564 -0.682384622 1.379993116 -3.116743871   
## 326 327 328 329 330   
## -2.769566852 -3.273999824 -2.051801221 3.998256673 3.130323441   
## 331 332 333 334 335   
## 1.262420477 6.580210836 -4.873427227 -6.967437510 0.747483663   
## 336 337 338 339 340   
## -7.002243999 3.438469600 4.755441686 -7.568849446 -0.910493463   
## 341 342 343 344 345   
## -7.174040871 2.540718655 -6.020526935 0.319174558 -3.187131260   
## 346 347 348 349 350   
## 0.395260599 0.278756748 -1.744066598 -1.804738969 4.799345091   
## 351 352 353 354 355   
## -0.096698888 -5.501423498 -4.610022115 -3.072309539 0.537980851   
## 356 357 358 359 360   
## -5.599237287 0.408710576 4.541986966 -0.691118432 -0.477369833   
## 361 362 363 364 365   
## -4.378576575 -1.014924050 -13.243499841 -0.351681764 -4.150771526   
## 366 367 368 369 370   
## -7.357658169 1.700532878 -1.656619025 0.223008437 -1.763826525   
## 371 372 373 374 375   
## 9.158410898 3.363896780 0.460355282 -0.761510520 4.068601345   
## 376 377 378 379 380   
## -1.581627565 -2.151585965 -7.812498259 8.981421278 3.756435018   
## 381 382 383 384 385   
## -6.102976939 -4.747830323 0.395411182 -0.895997149 -0.988557493   
## 386 387 388 389 390   
## 1.166479239 -1.995908051 1.884413046 3.555607039 1.272096013   
## 391 392 393 394 395   
## -7.322759959 -0.375831374 -3.073490411 -9.024140668 0.564582742   
## 396 397 398 399 400   
## -4.840289309 -2.566144884 5.694677126 -7.040198605 -0.833119482   
## 401 402 403 404 405   
## 0.048793736 -1.320298645 2.736875596 2.059049402 -12.562645163   
## 406 407 408 409 410   
## 3.716344240 -5.965098358 -3.903013118 -3.369146290 8.567098137   
## 411 412 413 414 415   
## 2.647596911 -4.165612566 4.089447829 8.203284118 6.658444257   
## 416 417 418 419 420   
## -1.542767554 -0.952259281 4.469740811 -4.493982959 -5.969656037   
## 421 422 423 424 425   
## -1.463842029 6.966054399 -1.856129086 2.912140453 -3.857034374   
## 426 427 428 429 430   
## 0.538701463 -2.869439674 -3.921974539 -3.999216938 4.369527228   
## 431 432 433 434 435   
## 2.081176980 -8.511651861 3.825156917 -0.826710697 -1.400866659   
## 436 437 438 439 440   
## -0.563768469 -0.485462569 -1.965179656 6.533314468 -6.458017155   
## 441 442 443 444 445   
## -1.691608762 1.112335009 4.230774435 -0.427436791 -1.996191560   
## 446 447 448 449 450   
## 3.635861772 -1.869440642 -0.956443108 9.250961972 2.742547837   
## 451 452 453 454 455   
## -9.253448286 -6.381159920 -1.290071441 3.557293319 -2.250575274   
## 456 457 458 459 460   
## -0.590343621 -2.158889639 -5.310754821 3.119077724 3.749853420   
## 461 462 463 464 465   
## 2.651499825 7.197074682 8.801647425 -2.705501436 -12.864354797   
## 466 467 468 469 470   
## -1.201056136 -2.320378041 12.669624305 -4.083073366 1.323597056   
## 471 472 473 474 475   
## -3.080204014 0.799191521 -2.832484763 0.974262084 4.073250449   
## 476 477 478 479 480   
## -0.701385311 2.307052100 2.029980245 -4.509184747 2.975121616   
## 481 482 483 484 485   
## -0.156011359 4.783473166 3.734988418 3.024239906 -3.828861410   
## 486 487 488 489 490   
## 1.844496813 5.511701973 -6.615486628 0.374368387 4.157040853   
## 491 492 493 494 495   
## -1.404247181 -6.213368788 -1.860313284 -2.987281502 1.545237257   
## 496 497 498 499 500   
## 2.919507787 -1.733832316 -0.436953133 1.480634032 7.031833502   
## 501 502 503 504 505   
## 0.525917673 -2.108455811 -10.421699860 1.944083012 -0.490849977   
## 506 507 508   
## -3.341879038 9.739872558 0.872414816



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.98974, p-value = 0.001387

Residual Analysis for AR\_DLM\_mort\_53:

1. The data points are below the line at both the start and end of the trend. Randomness is not seen here.
2. From normal distribution curve, the distribution is almost symmetric. This suggests a good fit with a very few data falling outside the normal curve indicating Kurtosis.
3. The data at the tails is deviated but is normal for most part of the line suggesting normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.001518) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

So far AR model with order (5, 2) is better model.

Now let us calculate AIC, BIC and MASE scores and store them in a dataframe to check the better model based on MASE score.

attr(Koyck\_DLM\_mort\_chem1$model, "class") = "lm"  
attr(Koyck\_DLM\_mort\_part$model, "class") = "lm"  
attr(AR\_DLM\_mort\_52$model, "class") = "lm"  
attr(AR\_DLM\_mort\_53$model, "class") = "lm"  
  
v\_model\_name <- c("finite\_dlm\_mort", "PolyDLM\_model\_mort\_chem1", "PolyDLM\_model\_mort\_part", "Koyck\_DLM\_mort\_chem1", "Koyck\_DLM\_mort\_part", "AR\_DLM\_mort\_52", "AR\_DLM\_mort\_53")

MASE <- MASE(finite\_dlm\_mort$model, PolyDLM\_model\_mort\_chem1$model, PolyDLM\_model\_mort\_part$model, Koyck\_DLM\_mort\_chem1$model, Koyck\_DLM\_mort\_part$model, AR\_DLM\_mort\_52$model, AR\_DLM\_mort\_53$model)$MASE  
  
aic <- AIC(finite\_dlm\_mort$model, PolyDLM\_model\_mort\_chem1$model, PolyDLM\_model\_mort\_part$model, Koyck\_DLM\_mort\_chem1$model, Koyck\_DLM\_mort\_part$model, AR\_DLM\_mort\_52$model, AR\_DLM\_mort\_53$model)$AIC  
  
bic <- BIC(finite\_dlm\_mort$model, PolyDLM\_model\_mort\_chem1$model, PolyDLM\_model\_mort\_part$model, Koyck\_DLM\_mort\_chem1$model, Koyck\_DLM\_mort\_part$model, AR\_DLM\_mort\_52$model, AR\_DLM\_mort\_53$model)$BIC

v\_score <- data.frame(v\_model\_name, MASE, aic, bic)  
colnames(v\_score) <- c("MODEL\_NAME", "MASE", "AIC", "BIC")  
v\_score

## MODEL\_NAME MASE AIC BIC  
## 1 finite\_dlm\_mort 0.8897851 3293.419 3394.473  
## 2 PolyDLM\_model\_mort\_chem1 1.1041854 3478.836 3504.183  
## 3 PolyDLM\_model\_mort\_part 1.2603636 3606.089 3631.437  
## 4 Koyck\_DLM\_mort\_chem1 0.8708743 3241.979 3258.893  
## 5 Koyck\_DLM\_mort\_part 0.9116488 3285.238 3302.152  
## 6 AR\_DLM\_mort\_52 0.7391588 3087.036 3154.565  
## 7 AR\_DLM\_mort\_53 0.7385057 3088.377 3160.128

Comparitively, AR\_DLM\_solar\_53 is the better model in terms of MASE, AIC and BIC scores.

Now let us fit dynamic lm model

#### Dynamic model

v\_mort\_dyna <- dynlm(y ~ x2 + x4, data = data.frame(v\_data\_TS))  
summary(v\_mort\_dyna)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 508  
##   
## Call:  
## dynlm(formula = y ~ x2 + x4, data = data.frame(v\_data\_TS))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.164 -5.582 -0.935 4.324 39.903   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 79.0240 1.2897 61.273 <2e-16 \*\*\*  
## x2 1.8404 0.1956 9.410 <2e-16 \*\*\*  
## x4 -0.1029 0.0486 -2.118 0.0346 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.281 on 505 degrees of freedom  
## Multiple R-squared: 0.3168, Adjusted R-squared: 0.3141   
## F-statistic: 117.1 on 2 and 505 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 117.1

R - squared is 0.3168

Adjusted R - squared is 0.3141

Degrees of freedom - DF are (2, 505)

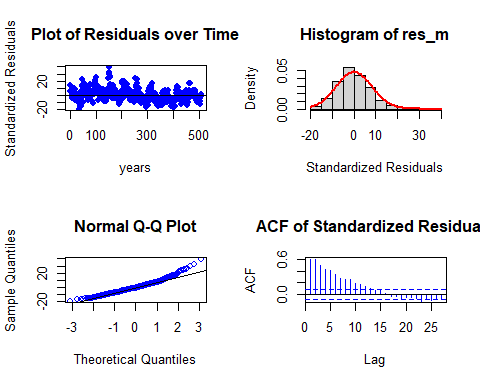
p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Dynamic linear model.

This model suggests that there is only 73.76% of data variance. Suggesting that the model explains only 73.76% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(v\_mort\_dyna))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.96747, p-value = 3.518e-09

Residual Analysis for v\_mort\_dyna:

1. The data points are above the line at both the start and end of the trend. Randomness is not seen here.
2. From normal distribution curve, the distribution is almost symmetric. This suggests a good fit with a very few data falling outside the normal curve indicating Kurtosis.
3. The data at the tails is deviated but is normal for most part of the line suggesting normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern
5. p - value (0.001518) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

### Exponential Smoothing

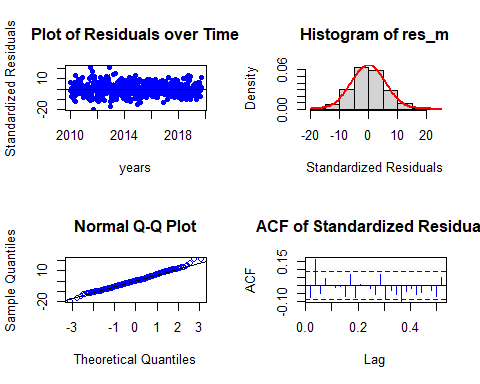
Since the Seasonality component is week we cannot get additive and multiplicative seasonality. So let us fit with simple seasonality. Since, we need next 4 weeks point forecasts as well as confidence intervals, we used h = 4 (frequency).

v\_mort\_ses <- ses(v\_Mortality\_data\_TS, seasonal = "simple", h = 4)  
summary(v\_mort\_ses)

##   
## Forecast method: Simple exponential smoothing  
##   
## Model Information:  
## Simple exponential smoothing   
##   
## Call:  
## ses(y = v\_Mortality\_data\_TS, h = 4, seasonal = "simple")   
##   
## Smoothing parameters:  
## alpha = 0.5111   
##   
## Initial states:  
## l = 98.9373   
##   
## sigma: 5.8932  
##   
## AIC AICc BIC   
## 4971.256 4971.303 4983.947   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.05496823 5.88156 4.588564 -0.3769165 5.156427 0.6871453  
## ACF1  
## Training set -0.07528577  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2019.735 84.66432 77.11191 92.21672 73.11391 96.21472  
## 2019.754 84.66432 76.18251 93.14612 71.69251 97.63612  
## 2019.774 84.66432 75.34534 93.98329 70.41218 98.91645  
## 2019.793 84.66432 74.57742 94.75121 69.23774 100.09089

Now let us check the residual analysis.

res\_analysis(residuals(v\_mort\_ses))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99538, p-value = 0.1364

Residual Analysis analysis for simple seasonality:

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (0.1364) from Shapiro-Wilk normality test is > 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

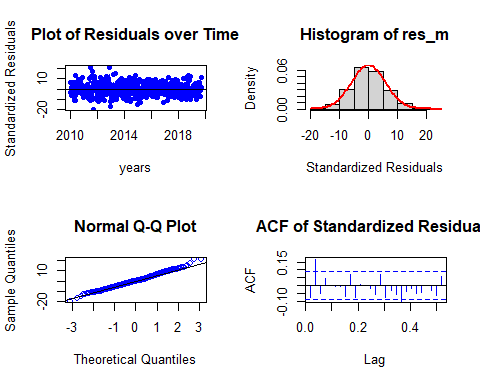
Now let us fit with damped trend.

v\_mort\_exp <- holt(v\_Mortality\_data\_TS, damped = TRUE, h = 4)  
summary(v\_mort\_exp)

##   
## Forecast method: Damped Holt's method  
##   
## Model Information:  
## Damped Holt's method   
##   
## Call:  
## holt(y = v\_Mortality\_data\_TS, h = 4, damped = TRUE)   
##   
## Smoothing parameters:  
## alpha = 0.5082   
## beta = 1e-04   
## phi = 0.9119   
##   
## Initial states:  
## l = 102.0415   
## b = -1.1632   
##   
## sigma: 5.9071  
##   
## AIC AICc BIC   
## 4976.632 4976.800 5002.015   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.02068605 5.877951 4.588308 -0.3396637 5.154426 0.687107  
## ACF1  
## Training set -0.0738815  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2019.735 84.64364 77.07339 92.21388 73.06595 96.22133  
## 2019.754 84.64444 76.15242 93.13646 71.65702 97.63187  
## 2019.774 84.64517 75.32181 93.96854 70.38632 98.90403  
## 2019.793 84.64584 74.55921 94.73248 69.21967 100.07202

Now let us check the residual analysis.

res\_analysis(residuals(v\_mort\_exp))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99547, p-value = 0.1477

Residual Analysis analysis for exponential trend:

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (0.1477) from Shapiro-Wilk normality test is > 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Due to week seasonality in the series there is no additive or multiplicative seasonality also there will be no damped in the series.

By exponential smoothing method we got the simple seasonal fit as the best model in terms of MASE and BIC scores.

### State Space Model Variations

Let us find the best ets model. Before all let us auto fit the model.

Since, the frequency is greater than 24 we cannot use ets() method. Therefore, stlf() is used.

Since, we need next 4 weeks point forecasts as well as confidence intervals, we used h = 4 (frequency).

v\_stlf\_fit <- stlf(v\_Mortality\_data\_TS, h = 4)  
summary(v\_stlf\_fit)

##   
## Forecast method: STL + ETS(M,N,N)  
##   
## Model Information:  
## ETS(M,N,N)   
##   
## Call:  
## ets(y = na.interp(x), model = etsmodel, allow.multiplicative.trend = allow.multiplicative.trend)   
##   
## Smoothing parameters:  
## alpha = 0.3015   
##   
## Initial states:  
## l = 92.3928   
##   
## sigma: 0.0537  
##   
## AIC AICc BIC   
## 4757.163 4757.210 4769.854   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.05169619 4.781323 3.768765 -0.3101758 4.269248 0.564379  
## ACF1  
## Training set -0.014883  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2019.735 83.70585 77.88797 89.52373 74.80818 92.60353  
## 2019.754 86.99713 80.91984 93.07442 77.70272 96.29155  
## 2019.774 89.32161 82.99547 95.64775 79.64662 98.99660  
## 2019.793 88.56710 82.00148 95.13272 78.52585 98.60835

STL + ETS(M, N, N)

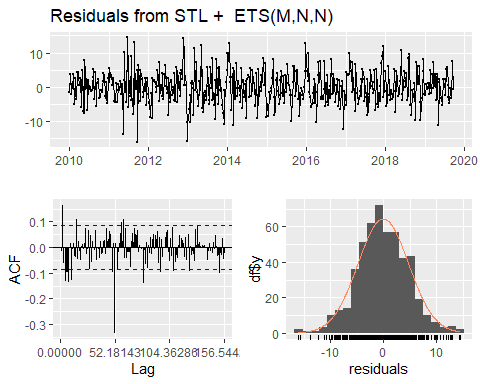
M - Multiplicative errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_stlf\_fit)



##   
## Ljung-Box test  
##   
## data: Residuals from STL + ETS(M,N,N)  
## Q\* = 245.16, df = 100, p-value = 3.397e-14  
##   
## Model df: 2. Total lags used: 102

Residual Analysis ETS(M, N, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Now let us fit ets variable combinations individually.

v\_ets\_fit1 <- ets(v\_Mortality\_data\_TS, model = "ANN")  
summary(v\_ets\_fit1)

## ETS(A,N,N)   
##   
## Call:  
## ets(y = v\_Mortality\_data\_TS, model = "ANN")   
##   
## Smoothing parameters:  
## alpha = 0.511   
##   
## Initial states:  
## l = 98.9364   
##   
## sigma: 5.8932  
##   
## AIC AICc BIC   
## 4971.256 4971.303 4983.947   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.05497816 5.88156 4.588578 -0.3769459 5.156433 0.6871475  
## ACF1  
## Training set -0.07517189

ETS(A, N, N)

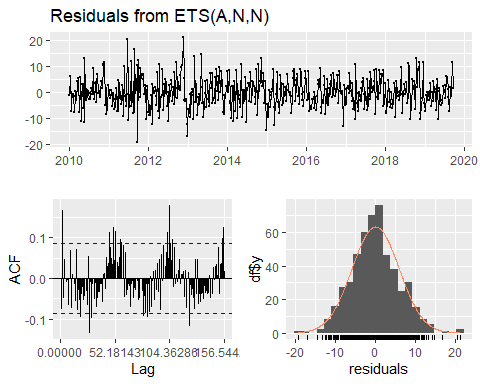
A - Additive errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit1)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(A,N,N)  
## Q\* = 191.57, df = 100, p-value = 1.006e-07  
##   
## Model df: 2. Total lags used: 102

Residual Analysis ETS(A, N, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit2 <- ets(v\_Mortality\_data\_TS, model = "AAN")  
summary(v\_ets\_fit2)

## ETS(A,A,N)   
##   
## Call:  
## ets(y = v\_Mortality\_data\_TS, model = "AAN")   
##   
## Smoothing parameters:  
## alpha = 0.5122   
## beta = 1e-04   
##   
## Initial states:  
## l = 100.9765   
## b = -0.029   
##   
## sigma: 5.906  
##   
## AIC AICc BIC   
## 4975.460 4975.579 4996.612   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.004457548 5.88274 4.590352 -0.3178407 5.155916 0.6874132  
## ACF1  
## Training set -0.07651869

ETS(A, A, N)

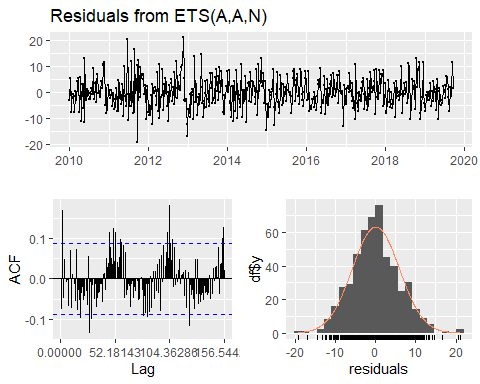
A - Additive errors

A - Additive trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit2)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(A,A,N)  
## Q\* = 192.28, df = 98, p-value = 4.215e-08  
##   
## Model df: 4. Total lags used: 102

Residual Analysis ETS(A, A, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit3 <- ets(v\_Mortality\_data\_TS, model = "MNN")  
summary(v\_ets\_fit3)

## ETS(M,N,N)   
##   
## Call:  
## ets(y = v\_Mortality\_data\_TS, model = "MNN")   
##   
## Smoothing parameters:  
## alpha = 0.4843   
##   
## Initial states:  
## l = 98.5582   
##   
## sigma: 0.0656  
##   
## AIC AICc BIC   
## 4954.111 4954.159 4966.803   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.05730399 5.88508 4.593891 -0.3849281 5.159608 0.6879431  
## ACF1  
## Training set -0.04372931

ETS(A, A, N)

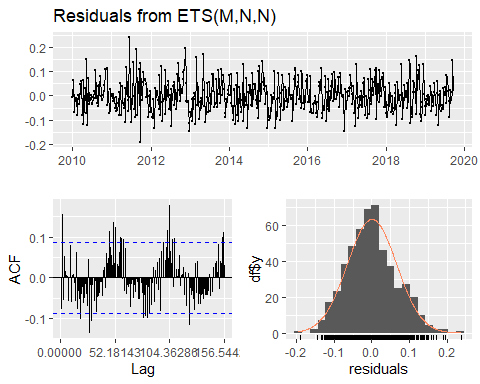
M - Multiplicative errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit3)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,N,N)  
## Q\* = 207.78, df = 100, p-value = 1.514e-09  
##   
## Model df: 2. Total lags used: 102

Residual Analysis ETS(M, N, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit4 <- ets(v\_Mortality\_data\_TS, model = "MAN")  
summary(v\_ets\_fit4)

## ETS(M,Ad,N)   
##   
## Call:  
## ets(y = v\_Mortality\_data\_TS, model = "MAN")   
##   
## Smoothing parameters:  
## alpha = 0.4311   
## beta = 0.0441   
## phi = 0.8   
##   
## Initial states:  
## l = 101.9204   
## b = -0.7466   
##   
## sigma: 0.0657  
##   
## AIC AICc BIC   
## 4957.818 4957.986 4983.201   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.041505 5.883233 4.604242 -0.3409889 5.167291 0.6894931  
## ACF1  
## Training set -0.02166677

ETS(A, A, N)

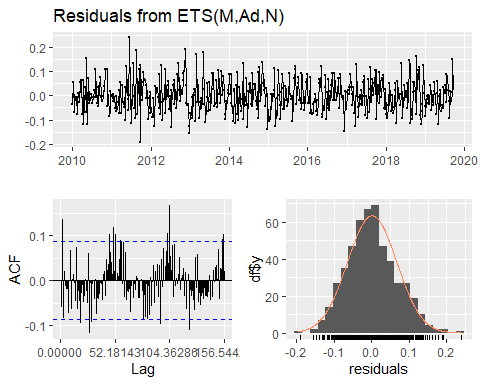
M - Multiplicative errors

A - Additive trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit4)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,Ad,N)  
## Q\* = 160.36, df = 97, p-value = 5.524e-05  
##   
## Model df: 5. Total lags used: 102

Residual Analysis ETS(M, A, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit5 <- ets(v\_Mortality\_data\_TS, model = "MMN")  
summary(v\_ets\_fit5)

## ETS(M,Md,N)   
##   
## Call:  
## ets(y = v\_Mortality\_data\_TS, model = "MMN")   
##   
## Smoothing parameters:  
## alpha = 0.4431   
## beta = 0.0363   
## phi = 0.8   
##   
## Initial states:  
## l = 100.788   
## b = 0.9747   
##   
## sigma: 0.0657  
##   
## AIC AICc BIC   
## 4957.923 4958.091 4983.306   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.06310061 5.884085 4.599372 -0.3702585 5.163763 0.6887639  
## ACF1  
## Training set -0.02809165

ETS(M, M, N)

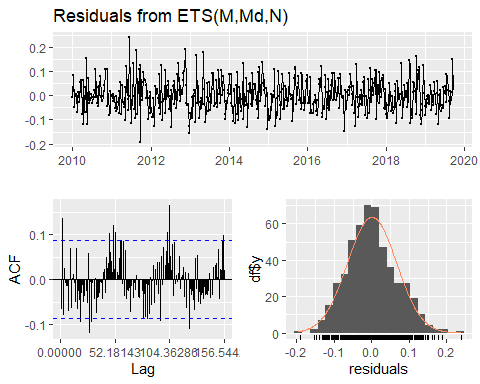
M - Multiplicative errors

M - Multiplicative trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit5)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,Md,N)  
## Q\* = 163.48, df = 97, p-value = 2.833e-05  
##   
## Model df: 5. Total lags used: 102

Residual Analysis ETS(M, M, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Comparitively, based on AIC, BIC and MASE scores ets(M, M, N) is better.

Among all the methods, ses in simple smoothing.

### Forecasting

Let us forecast the mortality rate for the next 4 weeks using the best model.

#### Forcasting with Smoothing method best model

fit <- ses(v\_Mortality\_data\_TS, seasonal = "simple", h = 4)  
  
v\_mort\_forecasts <- ts.intersect(ts(fit$lower[, 2], start = c(2020), frequency = 356.27/7), ts(fit$mean, start = c(2020), frequency = 356.27/7), ts(fit$upper[, 2], start = c(2020), frequency = 356.27/7))  
colnames(v\_mort\_forecasts) <- c("Lower bound", "Point forecast", "Upper bound")  
  
v\_mort\_forecasts

## Time Series:  
## Start = 2020   
## End = 2020.05894405928   
## Frequency = 50.8957142857143   
## Lower bound Point forecast Upper bound  
## 2020.000 73.11391 84.66432 96.21472  
## 2020.020 71.69251 84.66432 97.63612  
## 2020.039 70.41218 84.66432 98.91645  
## 2020.059 69.23774 84.66432 100.09089

Now let us plot the forecast.

plot(fit, fcol = "white", main = "Forecast of Mortality rate series for the next 4 weeks", ylab = "Mortality rate")  
lines(fitted(fit), col = "red")  
lines(fit$mean, col = "blue", lwd = 2)  
legend("top", inset = .03, cex = 0.9, box.lty = 2, box.lwd = 2, pch = 1, lty = 1, col = c("red", "blue"), c("Data", "Forecasts"))

Chart

Description automatically generated

Fig 1.19: Next 4 weeks forecast on the Mortality rate Series with Smoothing method model.

From the four weeks forecast results we can predict that there will be decrease in the mortality rate in the future.

#### Forcasting with ets method best model

fit3 <- ets(v\_Mortality\_data\_TS, model="MMN", damped = T)  
fit <- forecast.ets(fit3, h = 4)  
  
v\_mort\_forecasts <- ts.intersect(ts(fit$lower[, 2], start = c(2020), frequency = 356.27/7), ts(fit$mean, start = c(2020), frequency = 356.27/7), ts(fit$upper[, 2], start = c(2020), frequency = 356.27/7))  
colnames(v\_mort\_forecasts) <- c("Lower bound", "Point forecast", "Upper bound")  
  
v\_mort\_forecasts

## Time Series:  
## Start = 2020   
## End = 2020.05894405928   
## Frequency = 50.8957142857143   
## Lower bound Point forecast Upper bound  
## 2020.000 73.22889 84.62952 95.77543  
## 2020.020 73.36310 84.96962 97.26087  
## 2020.039 72.18016 85.24268 99.06327  
## 2020.059 71.36679 85.46177 100.76502

Now let us plot the forecast.

plot(fit, fcol = "white", main = "Forecast of Mortality rate series for the next 4 weeks", ylab = "Mortality rate")  
lines(fitted(fit), col = "red")  
lines(fit$mean, col = "blue", lwd = 2)  
legend("top", inset = .03, cex = 0.9, box.lty = 2, box.lwd = 2, pch = 1, lty = 1, col = c("red", "blue"), c("Data", "Forecasts"))

Chart

Description automatically generated

Fig 1.20: Next 4 weeks forecast on the Mortality rate Series with ets method model.

From the four weeks forecast results we can predict that there will be increase in the mortality rate in the future.

## Task 2

### Data

The data here used is the yearly averaged climate variables measured from 1984 – 2014 (31 years).

v\_First\_Flowering\_Day\_data <- read.csv("FFD.csv", header = TRUE)  
head(v\_First\_Flowering\_Day\_data)

## ï..Year Temperature Rainfall Radiation RelHumidity FFD  
## 1 1984 9.371585 2.489344 14.87158 93.92650 217  
## 2 1985 9.656164 2.475890 14.68493 94.93589 186  
## 3 1986 9.273973 2.421370 14.51507 94.09507 233  
## 4 1987 9.219178 2.319726 14.67397 94.49699 222  
## 5 1988 10.202186 2.465301 14.74863 94.08142 214  
## 6 1989 9.441096 2.735890 14.78356 96.08685 237

# Using str() to check the type of each column.  
str(v\_First\_Flowering\_Day\_data)

## 'data.frame': 31 obs. of 6 variables:  
## $ ï..Year : int 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 ...  
## $ Temperature: num 9.37 9.66 9.27 9.22 10.2 ...  
## $ Rainfall : num 2.49 2.48 2.42 2.32 2.47 ...  
## $ Radiation : num 14.9 14.7 14.5 14.7 14.7 ...  
## $ RelHumidity: num 93.9 94.9 94.1 94.5 94.1 ...  
## $ FFD : int 217 186 233 222 214 237 213 206 188 234 ...

Checking for Missing values.

colSums(is.na(v\_First\_Flowering\_Day\_data))

## ï..Year Temperature Rainfall Radiation RelHumidity FFD   
## 0 0 0 0 0 0

There are no missing values in the data.

Checking the class of v\_solar\_data. (It should be a data frame.)

class(v\_First\_Flowering\_Day\_data)

## [1] "data.frame"

v\_First\_Flowering\_Day\_Temp\_TS <- ts(v\_First\_Flowering\_Day\_data$Temperature, start = 1984, frequency = 1)  
v\_First\_Flowering\_Day\_Rainfall\_TS <- ts(v\_First\_Flowering\_Day\_data$Rainfall, start = 1984, frequency = 1)  
v\_First\_Flowering\_Day\_Radiation\_TS <- ts(v\_First\_Flowering\_Day\_data$Radiation, start = 1984, frequency = 1)  
v\_First\_Flowering\_Day\_RelHumidity\_TS <- ts(v\_First\_Flowering\_Day\_data$RelHumidity, start = 1984, frequency = 1)  
v\_First\_Flowering\_Day\_data\_TS <- ts(v\_First\_Flowering\_Day\_data$FFD, start = 1984, frequency = 1)

Confirming the class of each time series object.

class(v\_First\_Flowering\_Day\_Temp\_TS)

## [1] "ts"

class(v\_First\_Flowering\_Day\_Rainfall\_TS)

## [1] "ts"

class(v\_First\_Flowering\_Day\_Radiation\_TS)

## [1] "ts"

class(v\_First\_Flowering\_Day\_RelHumidity\_TS)

## [1] "ts"

class(v\_First\_Flowering\_Day\_data\_TS)

## [1] "ts"

Now let us perform descriptive analysis on each time series object.

### Descriptive Analysis

#### First Flowering Day

plot(v\_First\_Flowering\_Day\_data\_TS, type = "b", xlab = "years", ylab = "First Flowering Day", main = "Time series plot for yearly First Flowering Day data from 1984 – 2014 (31 years)", pch = 1)  
legend("topright", inset = .03, title = "First Flowering Day", legend = "First Flowering Day series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 1.1: First Flowering Day - Time series plot.

McLeod.Li.test(y = v\_First\_Flowering\_Day\_data\_TS, main = "McLeod-Li Test Statistics for First Flowering Day.")

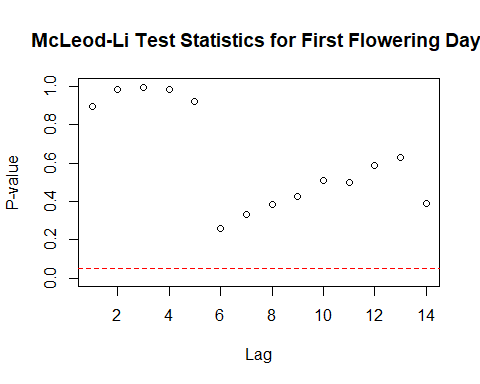


Fig 2.2: McLeod-Li Test Statistics for First Flowering Day.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around multiple years.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Temperature

plot(v\_First\_Flowering\_Day\_Temp\_TS, type = "b", xlab = "years", ylab = "Temperature", main = "Time series plot for yearly temperature from 1984 – 2014 (31 years)", pch = 1)  
legend("top", inset = .03, title = "Temperature", legend = "Temperature series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 2.3: Temperature - Time series plot.

McLeod.Li.test(y = v\_First\_Flowering\_Day\_Temp\_TS, main = "McLeod-Li Test Statistics for Temperature")

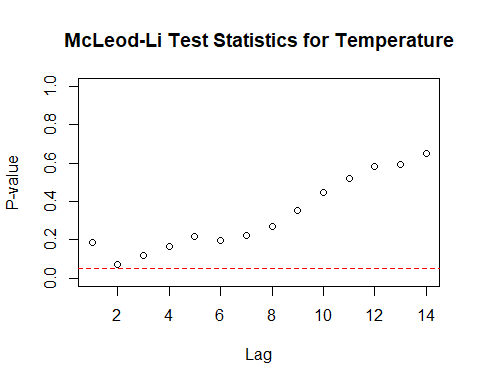


Fig 2.4: McLeod-Li Test Statistics for Temperature

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1996.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Rainfall

plot(v\_First\_Flowering\_Day\_Rainfall\_TS, type = "b", xlab = "years", ylab = "Rainfall", main = "Time series plot for yearly Rainfall from 1984 – 2014 (31 years)", pch = 1)  
legend("bottomleft", inset = .03, title = "Rainfall", legend = "Rainfall series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 2.5: Rainfall - Time series plot.

McLeod.Li.test(y = v\_First\_Flowering\_Day\_Rainfall\_TS, main = "McLeod-Li Test Statistics for Rainfall")

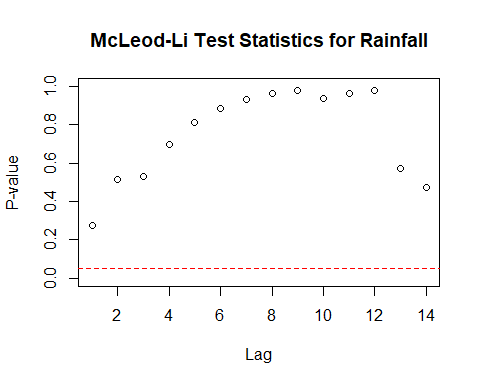


Fig 2.6: McLeod-Li Test Statistics for Rainfall

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1996.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Radiation

plot(v\_First\_Flowering\_Day\_Radiation\_TS, type = "b", xlab = "years", ylab = "Radiation", main = "Time series plot for yearly Radiation from 1984 – 2014 (31 years)", pch = 1)  
legend("topleft", inset = .03, title = "Radiation", legend = "Radiation series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 2.7: Solar radiation - Time series plot.

McLeod.Li.test(y = v\_First\_Flowering\_Day\_Radiation\_TS, main = "McLeod-Li Test Statistics for Radiation.")

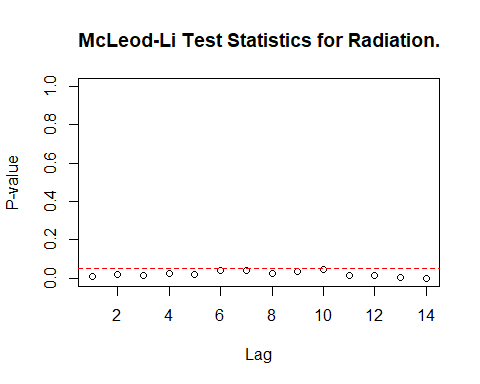


Fig 2.8: McLeod-Li Test Statistics for Radiation.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1992.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Relative Humidity

plot(v\_First\_Flowering\_Day\_RelHumidity\_TS, type = "b", xlab = "years", ylab = "Relative Humidity", main = "Time series plot for yearly Relative Humidity from 1984 – 2014 (31 years)", pch = 1)  
legend("bottomright", inset = .03, title = "Relative Humidity", legend = "Relative Humidity series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 2.9: Relative Humidity - Time series plot.

McLeod.Li.test(y = v\_First\_Flowering\_Day\_RelHumidity\_TS, main = "McLeod-Li Test Statistics for Relative Humidity")

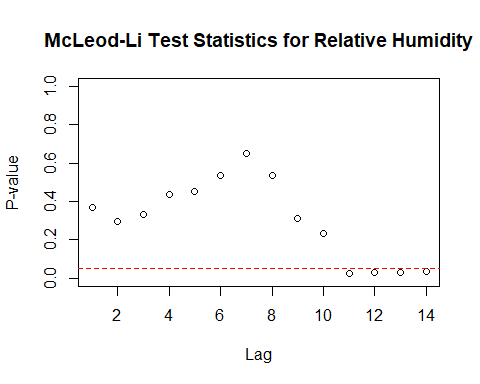


Fig 2.10: McLeod-Li Test Statistics for Relative Humidity.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1989.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

### Checking for Stationary in the series

Checking for Stationary on First Flowering Day series.

Stationary\_Check(v\_First\_Flowering\_Day\_data\_TS, "First Flowering Day - ACF plot", "First Flowering Day - PACF plot")

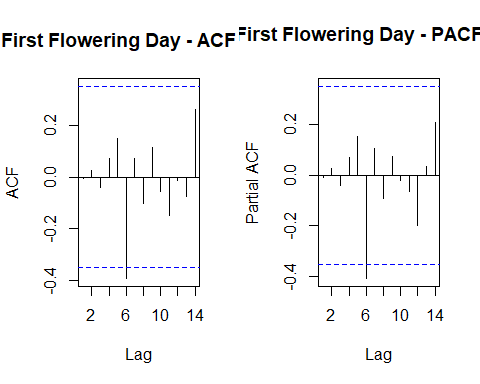


Fig 2.11: First Flowering Day - ACF

Fig 2.12: First Flowering Day - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -5.4552, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the First Flowering Day series is Stationary.

Checking for Stationary on Temperature data.

Stationary\_Check(v\_First\_Flowering\_Day\_Temp\_TS, "Temperature - ACF plot", "Temperature - PACF plot")

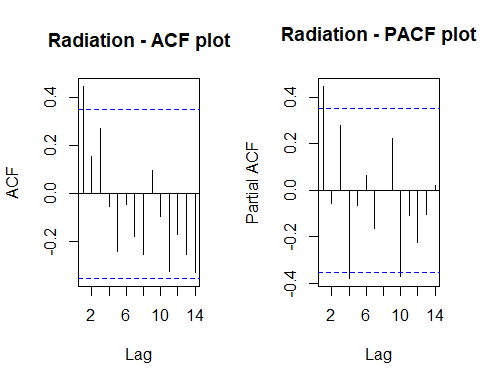


Fig 2.13: Temperature - ACF

Fig 2.14: Temperature - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -1.1484, Lag order = 2, p-value = 0.9002  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.9002 > 0.05

p - value is greater than 0.05 and hence the test is not statistically significant. Therefore, Null hypothesis cannot be rejected i.e., The data is not stationary.

Therefore, the Temperature series is not Stationary.

Checking for Stationary on Radiation data.

Stationary\_Check(v\_First\_Flowering\_Day\_Radiation\_TS, "Radiation - ACF plot", "Radiation - PACF plot")

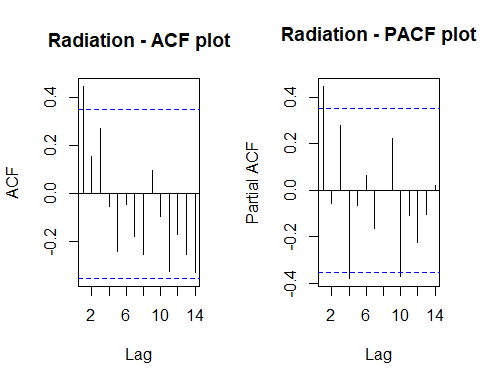


Fig 2.15: Radiation - ACF

Fig 2.16: Radiation - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -2.7317, Lag order = 4, p-value = 0.2911  
## alternative hypothesis: stationary

The is only one significant lag in the ACF and PACF plot.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.2911 > 0.05

p - value is greater than 0.05 and hence the test is not statistically significant. Therefore, Null hypothesis cannot be rejected i.e., The data is not stationary.

Therefore, the Radiation series is not Stationary.

Checking for Stationary on Rainfall data.

Stationary\_Check(v\_First\_Flowering\_Day\_Rainfall\_TS, "Rainfall - ACF plot", "Rainfall - PACF plot")

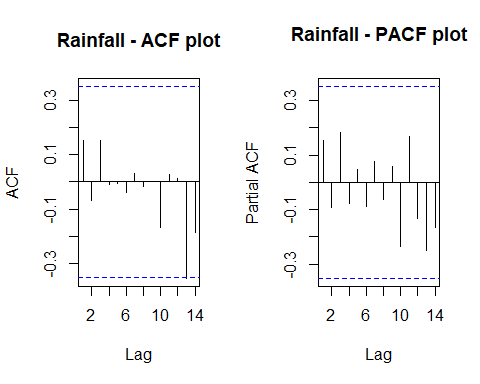


Fig 2.17: Rainfall - ACF

Fig 2.18: Rainfall - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5622, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Rainfall series is Stationary.

Checking for Stationary on Relative Humidity data.

Stationary\_Check(v\_First\_Flowering\_Day\_RelHumidity\_TS, "Relative Humidity - ACF plot", "Relative Humidity - PACF plot")

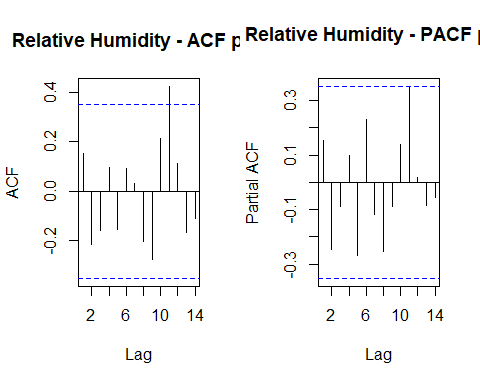


Fig 2.19: Relative Humidity - ACF

Fig 2.20: Relative Humidity - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5749, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Relative Humidity series is Stationary.

### Suitable Distributed lag models (Univariate analysis).

Before this let us find the correlation between the two series.

# Calculating the correlation coefficient.  
cor(v\_First\_Flowering\_Day\_data\_TS, v\_First\_Flowering\_Day\_Temp\_TS)

## [1] -0.2479337

cor(v\_First\_Flowering\_Day\_data\_TS, v\_First\_Flowering\_Day\_Rainfall\_TS)

## [1] 0.0506911

cor(v\_First\_Flowering\_Day\_data\_TS, v\_First\_Flowering\_Day\_Radiation\_TS)

## [1] 0.04677758

cor(v\_First\_Flowering\_Day\_data\_TS, v\_First\_Flowering\_Day\_RelHumidity\_TS)

## [1] -0.1285024

This suggests that FFD has a better correlation with Rainfall and Radiation.

As we are going to forecast the FFD data, our dependent variable “y” will be Mortality Rate series object and independent variable “x” will be Rainfall and Radiation.

As we need to check with and without intercept, let us convert the entire data set into time series.

v\_data\_TS\_22 <- ts(v\_First\_Flowering\_Day\_data, start = 1984, frequency = 1)  
  
cor(v\_data\_TS\_22)

## ï..Year Temperature Rainfall Radiation RelHumidity  
## ï..Year 1.0000000 0.148410676 -0.1752091 0.11881829 0.206355767  
## Temperature 0.1484107 1.000000000 0.3933255 -0.24096625 0.009646021  
## Rainfall -0.1752091 0.393325545 1.0000000 -0.58131610 0.338461007  
## Radiation 0.1188183 -0.240966245 -0.5813161 1.00000000 -0.055209652  
## RelHumidity 0.2063558 0.009646021 0.3384610 -0.05520965 1.000000000  
## FFD -0.2329975 -0.247933708 0.0506911 0.04677758 -0.128502440  
## FFD  
## ï..Year -0.23299747  
## Temperature -0.24793371  
## Rainfall 0.05069110  
## Radiation 0.04677758  
## RelHumidity -0.12850244  
## FFD 1.00000000

colnames(v\_data\_TS\_22) <- c("x1", "x2", "x3", "x4", "x5", "y")

#### Finite distributed lag model with Rainfall

##### With slope.

for ( i in 1:10){  
 model\_1 = dlm(formula=y ~ x3, data=data.frame(v\_data\_TS\_22), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_1$model), "BIC = ", BIC(model\_1$model), "MASE =", MASE(model\_1)$MASE, "\n")  
 }

## q = 1 AIC = 281.9769 BIC = 287.5817 MASE = 0.6758928   
## q = 2 AIC = 273.8231 BIC = 280.6596 MASE = 0.6752737   
## q = 3 AIC = 265.1863 BIC = 273.1795 MASE = 0.6326564   
## q = 4 AIC = 257.086 BIC = 266.1568 MASE = 0.6309338   
## q = 5 AIC = 246.1244 BIC = 256.1891 MASE = 0.5808908   
## q = 6 AIC = 238.0915 BIC = 249.0614 MASE = 0.5522086   
## q = 7 AIC = 231.4148 BIC = 243.1954 MASE = 0.5447815   
## q = 8 AIC = 224.8048 BIC = 237.2953 MASE = 0.5553913   
## q = 9 AIC = 216.4839 BIC = 229.5764 MASE = 0.5521884   
## q = 10 AIC = 197.7206 BIC = 211.2994 MASE = 0.391709

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC, BIC and MASE  
  
finite\_dlm\_FFD\_slope = dlm(formula = y ~ x3, data=data.frame(v\_data\_TS\_22), q = 10)  
summary(finite\_dlm\_FFD\_slope)

##   
## Call:  
## lm(formula = as.formula(model.formula), data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -25.4869 -13.6970 0.5488 9.6595 30.6423   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 260.256 89.788 2.899 0.0176 \*  
## x3.t -17.281 13.476 -1.282 0.2317   
## x3.1 15.520 12.336 1.258 0.2400   
## x3.2 -10.633 12.733 -0.835 0.4253   
## x3.3 32.450 12.574 2.581 0.0297 \*  
## x3.4 -32.881 13.959 -2.356 0.0429 \*  
## x3.5 40.010 13.844 2.890 0.0179 \*  
## x3.6 -5.032 13.415 -0.375 0.7163   
## x3.7 24.216 14.836 1.632 0.1371   
## x3.8 -20.585 14.164 -1.453 0.1801   
## x3.9 -8.781 15.032 -0.584 0.5735   
## x3.10 -38.573 15.741 -2.450 0.0367 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22.05 on 9 degrees of freedom  
## Multiple R-squared: 0.6663, Adjusted R-squared: 0.2585   
## F-statistic: 1.634 on 11 and 9 DF, p-value: 0.2351  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 197.7206 211.2994

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model with slope.**

**HA: The data fits the Finite distributed lag model slope.**

**Interpretations:**

F - statistic is 1.634  
R - squared is 0.6663

Adjusted R - squared is 0.2585

Degrees of freedom - DF are (11, 9)

p - value (0.2351) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Finite distributed lag model with slope.

No residual analysis is required.

Therefore, Further analysis is needed by removing slope to the lag model.

##### With out slope.

for ( i in 1:10){  
 model\_2 = dlm(formula=y ~ 0 + x3, data=data.frame(v\_data\_TS\_22), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_2$model), "BIC = ", BIC(model\_2$model), "MASE =", MASE(model\_2)$MASE, "\n")  
 }

## q = 1 AIC = 300.1422 BIC = 304.3458 MASE = 0.9477487   
## q = 2 AIC = 283.7963 BIC = 289.2655 MASE = 0.8145653   
## q = 3 AIC = 271.2089 BIC = 277.8699 MASE = 0.6998298   
## q = 4 AIC = 264.7646 BIC = 272.5397 MASE = 0.7023612   
## q = 5 AIC = 249.1985 BIC = 258.0052 MASE = 0.6495448   
## q = 6 AIC = 242.314 BIC = 252.065 MASE = 0.6502001   
## q = 7 AIC = 233.9019 BIC = 244.5044 MASE = 0.6214299   
## q = 8 AIC = 227.4548 BIC = 238.8097 MASE = 0.6177108   
## q = 9 AIC = 218.2618 BIC = 230.2633 MASE = 0.6063767   
## q = 10 AIC = 209.5666 BIC = 222.1008 MASE = 0.5785768

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC, BIC and MASE  
  
finite\_dlm\_FFD\_noslope = dlm(formula=y ~ 0 + x3, data=data.frame(v\_data\_TS\_22), q = 10)  
summary(finite\_dlm\_FFD\_noslope)

##   
## Call:  
## lm(formula = as.formula(model.formula), data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -38.59 -21.00 3.99 17.77 28.44   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## x3.t 1.629 15.554 0.105 0.9187   
## x3.1 23.569 15.856 1.486 0.1680   
## x3.2 2.783 15.647 0.178 0.8624   
## x3.3 35.931 16.511 2.176 0.0546 .  
## x3.4 -22.958 17.851 -1.286 0.2274   
## x3.5 40.349 18.261 2.210 0.0516 .  
## x3.6 2.049 17.401 0.118 0.9086   
## x3.7 29.584 19.418 1.524 0.1586   
## x3.8 -4.658 17.221 -0.270 0.7923   
## x3.9 2.018 19.211 0.105 0.9184   
## x3.10 -20.819 19.128 -1.088 0.3020   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 29.09 on 10 degrees of freedom  
## Multiple R-squared: 0.9907, Adjusted R-squared: 0.9805   
## F-statistic: 97.01 on 11 and 10 DF, p-value: 1.202e-08  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 209.5666 222.1008

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model without slope.**

**HA: The data fits the Finite distributed lag model with out slepe.**

**Interpretations:**

F - statistic is 97.01

R - squared is 0.9907

Adjusted R - squared is 0.9805

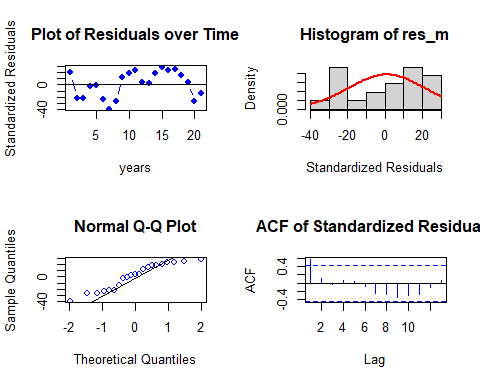
Degrees of freedom - DF are (11, 10)

p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Finite distributed lag model without slope.

This model suggests that there is only 98.05% of data variance. Suggesting that the model explains only 98.05% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

res\_analysis(residuals(finite\_dlm\_FFD\_noslope$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.92077, p-value = 0.0899

Residual Analysis for Finite DLM:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen in the data. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. QQplot also suggests that there is no normality in the trend.
4. There is only one significant lagin Autocorrelation plot.
5. p - value (0.0899) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Even though the model fits better it is poor when it comes to residual analysis. Therefore, Further analysis is needed by adding polynomial to the lag model.

y = v\_First\_Flowering\_Day\_data\_TS # Independent variable  
x1 = v\_First\_Flowering\_Day\_Rainfall\_TS # Dependent variable  
x2 = v\_First\_Flowering\_Day\_Radiation\_TS # Dependent variable

#### Polynomial distributed lag model with Rainfall

for (i in 1:3){  
 model\_3 <- polyDlm(x = as.vector(x1), y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_3$model), "BIC = ", BIC(model\_3$model), MASE(model\_3)$MASE, "\n")  
}

## q = 1 k = 1 AIC = 281.9769 BIC = 287.5817 0.6758928   
## q = 2 k = 2 AIC = 273.8231 BIC = 280.6596 0.6752737   
## q = 3 k = 3 AIC = 265.1863 BIC = 273.1795 0.6326564

Let us fit a polynomial model of order 3. Since least AIC and BIC scores.

# Ploynomial DLM  
  
PolyDLM\_model\_FFD = polyDlm(x = as.vector(x1), y = as.vector(y), q = 3, k = 3, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 0.784 12.5 0.0629 0.950  
## beta.1 4.950 12.5 0.3970 0.695  
## beta.2 7.740 12.5 0.6190 0.541  
## beta.3 14.900 12.6 1.1800 0.248

summary(PolyDLM\_model\_FFD)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -46.70 -13.09 2.31 14.28 45.17   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 142.2479 51.4936 2.762 0.0111 \*  
## z.t0 0.7836 12.4508 0.063 0.9504   
## z.t1 6.7375 55.3454 0.122 0.9042   
## z.t2 -3.5211 49.2855 -0.071 0.9437   
## z.t3 0.9461 10.8799 0.087 0.9315   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 24.55 on 23 degrees of freedom  
## Multiple R-squared: 0.09312, Adjusted R-squared: -0.0646   
## F-statistic: 0.5904 on 4 and 23 DF, p-value: 0.673

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 0.5904

R - squared is 0.09312

Adjusted R - squared is -0.0646

Degrees of freedom - DF are (4, 23)

p - value (0.673) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Polynomial distributed lag model.

Also, this model suggests that there is only -67.3% of data variance.

No residual analysis is required.

Let us fit Koyck model.

#### Koyck model with Rainfall

# Koyk DLM  
  
Koyck\_DLM\_FFD = koyckDlm(x = as.vector(x1) , y = as.vector(y))  
summary(Koyck\_DLM\_FFD)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -58.691 -21.222 2.697 14.856 68.192   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 1.266e+02 2.185e+02 0.579 0.567  
## Y.1 4.591e-03 2.196e-01 0.021 0.983  
## X.t 3.448e+01 8.772e+01 0.393 0.697  
##   
## Residual standard error: 27.55 on 27 degrees of freedom  
## Multiple R-Squared: -0.2505, Adjusted R-squared: -0.3431   
## Wald test: 0.07773 on 2 and 27 DF, p-value: 0.9254   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: 127.2254 34.48095 0.004590669

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

Wald test statistic is 0.07773  
R - squared is -0.2505

Adjusted R - squared is -0.3431

Degrees of freedom - DF are (2, 27)

p - value (0.9254) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Koyck distributed lag model.

Also, this model suggests that there is only -34.31% of data variance.

No residual analysis is required.

Let us fit ardlDlm model to check whether it fits better or not.

#### Autoregressive distributed lag model with Rainfall

##### with slope

for (i in 1:5){  
 for(j in 1:5){  
 model\_4 = ardlDlm(formula=y ~ x3, data=data.frame(v\_data\_TS\_22), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_4$model), "BIC = ", BIC(model\_4$model), "MASE =", MASE(model\_4)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = 283.9744 BIC = 290.9804 MASE = 0.6760282   
## p = 1 q = 2 AIC = 276.7348 BIC = 284.9385 MASE = 0.6838177   
## p = 1 q = 3 AIC = 269.3361 BIC = 278.6615 MASE = 0.6676193   
## p = 1 q = 4 AIC = 262.6875 BIC = 273.0542 MASE = 0.6489023   
## p = 1 q = 5 AIC = 255.5705 BIC = 266.8934 MASE = 0.6396905   
## p = 2 q = 1 AIC = 275.8163 BIC = 284.0201 MASE = 0.6749962   
## p = 2 q = 2 AIC = 277.8084 BIC = 287.3795 MASE = 0.6762326   
## p = 2 q = 3 AIC = 270.6698 BIC = 281.3274 MASE = 0.6621822   
## p = 2 q = 4 AIC = 264.2889 BIC = 275.9514 MASE = 0.6506873   
## p = 2 q = 5 AIC = 257.2016 BIC = 269.7826 MASE = 0.641259   
## p = 3 q = 1 AIC = 267.1614 BIC = 276.4868 MASE = 0.6355497   
## p = 3 q = 2 AIC = 269.1262 BIC = 279.7838 MASE = 0.639182   
## p = 3 q = 3 AIC = 270.8152 BIC = 282.8051 MASE = 0.6371823   
## p = 3 q = 4 AIC = 264.4739 BIC = 277.4323 MASE = 0.6246495   
## p = 3 q = 5 AIC = 258.2483 BIC = 272.0874 MASE = 0.6369725   
## p = 4 q = 1 AIC = 259.0485 BIC = 269.4152 MASE = 0.6272275   
## p = 4 q = 2 AIC = 260.686 BIC = 272.3485 MASE = 0.6099768   
## p = 4 q = 3 AIC = 262.1327 BIC = 275.091 MASE = 0.601955   
## p = 4 q = 4 AIC = 263.7563 BIC = 278.0105 MASE = 0.6017331   
## p = 4 q = 5 AIC = 257.7752 BIC = 272.8724 MASE = 0.6175106   
## p = 5 q = 1 AIC = 247.6247 BIC = 258.9476 MASE = 0.5835814   
## p = 5 q = 2 AIC = 249.3037 BIC = 261.8846 MASE = 0.5755967   
## p = 5 q = 3 AIC = 247.4178 BIC = 261.2568 MASE = 0.5484   
## p = 5 q = 4 AIC = 248.4355 BIC = 263.5326 MASE = 0.5256786   
## p = 5 q = 5 AIC = 250.0315 BIC = 266.3868 MASE = 0.5306522

(p, q) = (5, 4) has the least AIC, BIC and MASE scores.

# ARDLM model  
AR\_DLM\_FFD\_54\_slope = ardlDlm(formula = y ~ x3, data=data.frame(v\_data\_TS\_22), p = 5 , q = 4)  
summary(AR\_DLM\_FFD\_54\_slope)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -27.12 -15.50 -0.65 13.13 35.40   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 120.7626 87.1334 1.386 0.1860   
## x3.t 1.4092 15.0160 0.094 0.9265   
## x3.1 24.4971 15.7233 1.558 0.1401   
## x3.2 -13.2575 16.1170 -0.823 0.4236   
## x3.3 27.1489 13.2278 2.052 0.0580 .  
## x3.4 -23.8080 15.4248 -1.543 0.1435   
## x3.5 41.6349 16.3506 2.546 0.0224 \*  
## y.1 0.2037 0.2245 0.907 0.3785   
## y.2 -0.1554 0.3001 -0.518 0.6120   
## y.3 -0.4561 0.2707 -1.685 0.1127   
## y.4 0.1863 0.2452 0.760 0.4591   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.86 on 15 degrees of freedom  
## Multiple R-squared: 0.4339, Adjusted R-squared: 0.05646   
## F-statistic: 1.15 on 10 and 15 DF, p-value: 0.3911

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model with slope.**

**HA: The data fits the Autoregressive distributed lag model with slope.**

**Interpretations:**

F - statistic is 1.15  
R - squared is 0.4339

Adjusted R - squared is 0.05646  
Degrees of freedom - DF are (2, 27)

p - value (0.3911) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Autoregressive distributed lag model with slope.

Also, this model suggests that there is only 5% of data variance.

No residual analysis is required.

##### With out slope.

for (i in 1:5){  
 for(j in 1:5){  
 model\_5 = ardlDlm(formula=y ~ 0 + x3, data=data.frame(v\_data\_TS\_22), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_5$model), "BIC = ", BIC(model\_5$model), "MASE =", MASE(model\_5)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = 283.9744 BIC = 290.9804 MASE = 0.6760282   
## p = 1 q = 2 AIC = 276.7348 BIC = 284.9385 MASE = 0.6838177   
## p = 1 q = 3 AIC = 269.3361 BIC = 278.6615 MASE = 0.6676193   
## p = 1 q = 4 AIC = 262.6875 BIC = 273.0542 MASE = 0.6489023   
## p = 1 q = 5 AIC = 255.5705 BIC = 266.8934 MASE = 0.6396905   
## p = 2 q = 1 AIC = 275.8163 BIC = 284.0201 MASE = 0.6749962   
## p = 2 q = 2 AIC = 277.8084 BIC = 287.3795 MASE = 0.6762326   
## p = 2 q = 3 AIC = 270.6698 BIC = 281.3274 MASE = 0.6621822   
## p = 2 q = 4 AIC = 264.2889 BIC = 275.9514 MASE = 0.6506873   
## p = 2 q = 5 AIC = 257.2016 BIC = 269.7826 MASE = 0.641259   
## p = 3 q = 1 AIC = 267.1614 BIC = 276.4868 MASE = 0.6355497   
## p = 3 q = 2 AIC = 269.1262 BIC = 279.7838 MASE = 0.639182   
## p = 3 q = 3 AIC = 270.8152 BIC = 282.8051 MASE = 0.6371823   
## p = 3 q = 4 AIC = 264.4739 BIC = 277.4323 MASE = 0.6246495   
## p = 3 q = 5 AIC = 258.2483 BIC = 272.0874 MASE = 0.6369725   
## p = 4 q = 1 AIC = 259.0485 BIC = 269.4152 MASE = 0.6272275   
## p = 4 q = 2 AIC = 260.686 BIC = 272.3485 MASE = 0.6099768   
## p = 4 q = 3 AIC = 262.1327 BIC = 275.091 MASE = 0.601955   
## p = 4 q = 4 AIC = 263.7563 BIC = 278.0105 MASE = 0.6017331   
## p = 4 q = 5 AIC = 257.7752 BIC = 272.8724 MASE = 0.6175106   
## p = 5 q = 1 AIC = 247.6247 BIC = 258.9476 MASE = 0.5835814   
## p = 5 q = 2 AIC = 249.3037 BIC = 261.8846 MASE = 0.5755967   
## p = 5 q = 3 AIC = 247.4178 BIC = 261.2568 MASE = 0.5484   
## p = 5 q = 4 AIC = 248.4355 BIC = 263.5326 MASE = 0.5256786   
## p = 5 q = 5 AIC = 250.0315 BIC = 266.3868 MASE = 0.5306522

(p, q) = (5, 4) has the least AIC, BIC and MASE scores.

# Finite lag length based on AIC-BIC  
  
AR\_DLM\_FFD\_54\_noslope = ardlDlm(formula=y ~ 0 + x3, data=data.frame(v\_data\_TS\_22), p = 5 , q = 4)  
summary(AR\_DLM\_FFD\_54\_noslope)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -27.12 -15.50 -0.65 13.13 35.40   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 120.7626 87.1334 1.386 0.1860   
## x3.t 1.4092 15.0160 0.094 0.9265   
## x3.1 24.4971 15.7233 1.558 0.1401   
## x3.2 -13.2575 16.1170 -0.823 0.4236   
## x3.3 27.1489 13.2278 2.052 0.0580 .  
## x3.4 -23.8080 15.4248 -1.543 0.1435   
## x3.5 41.6349 16.3506 2.546 0.0224 \*  
## y.1 0.2037 0.2245 0.907 0.3785   
## y.2 -0.1554 0.3001 -0.518 0.6120   
## y.3 -0.4561 0.2707 -1.685 0.1127   
## y.4 0.1863 0.2452 0.760 0.4591   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.86 on 15 degrees of freedom  
## Multiple R-squared: 0.4339, Adjusted R-squared: 0.05646   
## F-statistic: 1.15 on 10 and 15 DF, p-value: 0.3911

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model without slope.**

**HA: The data fits the Autoregressive distributed lag model without slope.**

**Interpretations:**

F - statistic is 1.15  
R - squared is 0.4339

Adjusted R - squared is 0.05646  
Degrees of freedom - DF are (10, 15)

p - value (0.3911) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Autoregressive distributed lag model without slope.

Also, this model suggests that there is only 5.64% of data variance.

No residual analysis is required.

Therefore let us fit models with Radiation.

#### Finite distributed lag model with Radiation

##### With slope.

for ( i in 1:10){  
 model\_1 = dlm(formula=y ~ x4, data=data.frame(v\_data\_TS\_22), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_1$model), "BIC = ", BIC(model\_1$model), "MASE =", MASE(model\_1)$MASE, "\n")  
 }

## q = 1 AIC = 281.571 BIC = 287.1758 MASE = 0.6579604   
## q = 2 AIC = 273.2458 BIC = 280.0823 MASE = 0.6672639   
## q = 3 AIC = 264.666 BIC = 272.6592 MASE = 0.6164754   
## q = 4 AIC = 257.9491 BIC = 267.02 MASE = 0.5968786   
## q = 5 AIC = 249.3748 BIC = 259.4396 MASE = 0.5870893   
## q = 6 AIC = 235.4683 BIC = 246.4382 MASE = 0.5298748   
## q = 7 AIC = 225.705 BIC = 237.4856 MASE = 0.479178   
## q = 8 AIC = 219.9335 BIC = 232.424 MASE = 0.4839788   
## q = 9 AIC = 211.6427 BIC = 224.7353 MASE = 0.4680972   
## q = 10 AIC = 204.3142 BIC = 217.893 MASE = 0.4716663

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC, BIC and MASE  
  
finite\_dlm\_FFD\_slope\_rad = dlm(formula = y ~ x4, data=data.frame(v\_data\_TS\_22), q = 10)  
summary(finite\_dlm\_FFD\_slope\_rad)

##   
## Call:  
## lm(formula = as.formula(model.formula), data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -36.650 -10.774 5.135 12.268 21.533   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 77.1798 518.3355 0.149 0.8849   
## x4.t -14.2644 31.1884 -0.457 0.6583   
## x4.1 22.6955 37.2327 0.610 0.5572   
## x4.2 -41.5518 22.3579 -1.858 0.0960 .  
## x4.3 41.3553 21.3508 1.937 0.0847 .  
## x4.4 -23.8377 22.3631 -1.066 0.3142   
## x4.5 36.6585 21.2569 1.725 0.1187   
## x4.6 -46.5116 22.0486 -2.110 0.0641 .  
## x4.7 21.3519 21.6192 0.988 0.3491   
## x4.8 0.1483 21.9721 0.007 0.9948   
## x4.9 18.1562 27.1471 0.669 0.5204   
## x4.10 -5.3017 27.2786 -0.194 0.8502   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 25.8 on 9 degrees of freedom  
## Multiple R-squared: 0.5432, Adjusted R-squared: -0.01504   
## F-statistic: 0.9731 on 11 and 9 DF, p-value: 0.5253  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 204.3142 217.893

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model with slope.**

**HA: The data fits the Finite distributed lag model with slope.**

**Interpretations:**

F - statistic is 0.9731  
R - squared is 0.5432

Adjusted R - squared is -0.01504

Degrees of freedom - DF are (11, 9)

p - value (0.5253) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model doesn’t fits the Finite distributed lag model with slope.

Therefore, no residual analysis is required.

Therefore, Further analysis is needed by removing slope to the lag model.

##### With out slope.

for ( i in 1:10){  
 model\_2 = dlm(formula=y ~ 0 + x4, data=data.frame(v\_data\_TS\_22), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_2$model), "BIC = ", BIC(model\_2$model), "MASE =", MASE(model\_2)$MASE, "\n")  
 }

## q = 1 AIC = 281.4295 BIC = 285.6331 MASE = 0.6606648   
## q = 2 AIC = 274.0228 BIC = 279.492 MASE = 0.6743494   
## q = 3 AIC = 264.5047 BIC = 271.1657 MASE = 0.6317954   
## q = 4 AIC = 256.7858 BIC = 264.5608 MASE = 0.588564   
## q = 5 AIC = 247.3861 BIC = 256.1928 MASE = 0.5877368   
## q = 6 AIC = 235.0562 BIC = 244.8072 MASE = 0.5208836   
## q = 7 AIC = 224.0442 BIC = 234.6467 MASE = 0.4710983   
## q = 8 AIC = 218.1936 BIC = 229.5485 MASE = 0.4792433   
## q = 9 AIC = 209.7754 BIC = 221.7769 MASE = 0.4655184   
## q = 10 AIC = 202.3659 BIC = 214.9002 MASE = 0.4686756

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC, BIC and MASE  
  
finite\_dlm\_FFD\_noslope\_rad = dlm(formula = y ~ 0 + x4, data=data.frame(v\_data\_TS\_22), q = 10)  
summary(finite\_dlm\_FFD\_noslope\_rad)

##   
## Call:  
## lm(formula = as.formula(model.formula), data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -37.504 -11.163 4.155 12.292 21.823   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## x4.t -12.329 26.930 -0.458 0.6569   
## x4.1 21.788 34.888 0.625 0.5463   
## x4.2 -40.600 20.350 -1.995 0.0740 .  
## x4.3 41.228 20.264 2.035 0.0693 .  
## x4.4 -23.444 21.093 -1.111 0.2924   
## x4.5 37.123 19.972 1.859 0.0927 .  
## x4.6 -46.252 20.877 -2.215 0.0511 .  
## x4.7 21.334 20.535 1.039 0.3233   
## x4.8 1.023 20.111 0.051 0.9604   
## x4.9 17.627 25.564 0.690 0.5062   
## x4.10 -3.306 22.569 -0.147 0.8864   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 24.5 on 10 degrees of freedom  
## Multiple R-squared: 0.9934, Adjusted R-squared: 0.9862   
## F-statistic: 137.1 on 11 and 10 DF, p-value: 2.186e-09  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 202.3659 214.9002

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model without slope.**

**HA: The data fits the Finite distributed lag model without slope.**

**Interpretations:**

F - statistic is 137.1  
R - squared is 0.9934

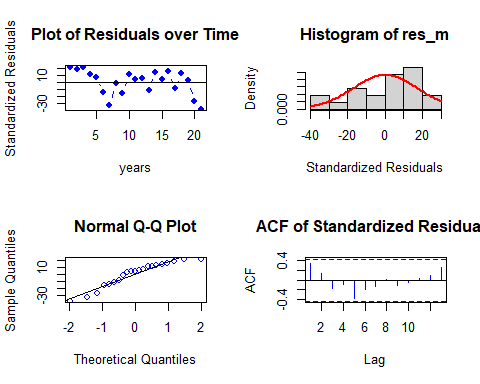
Adjusted R - squared is 0.9862

Degrees of freedom - DF are (11, 10) p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Finite distributed lag model without slope.

This model suggests that there is only 98.62% of data variance. Suggesting that the model explains only 98.62% of the trend. Which implies that the model fits better.

Now let us perform residual analysis.

res\_analysis(residuals(finite\_dlm\_FFD\_noslope\_rad$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.91671, p-value = 0.0746

Residual Analysis for Finite DLM:

1. The data points are above the line at the start and below the line at the end of the trend. Randomness is seen in the data. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. QQplot also suggests that there is no normality in the trend.
4. There is only one significant lagin Autocorrelation plot.
5. p - value (0.08) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Even though the model fits better it is poor when it comes to residual analysis. Therefore, Further analysis is needed by adding polynomial to the lag model.

#### Polynomial distributed lag model with Radiation

for (i in 1:3){  
 model\_3 <- polyDlm(x = as.vector(x2), y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_3$model), "BIC = ", BIC(model\_3$model), MASE(model\_3)$MASE, "\n")  
}

## q = 1 k = 1 AIC = 281.571 BIC = 287.1758 0.6579604   
## q = 2 k = 2 AIC = 273.2458 BIC = 280.0823 0.6672639   
## q = 3 k = 3 AIC = 264.666 BIC = 272.6592 0.6164754

Let us fit a polynomial model of order 3. Since least AIC and BIC scores.

# Ploynomial DLM  
  
PolyDLM\_model\_FFD\_rad = polyDlm(x = as.vector(x2), y = as.vector(y), q = 3, k = 3, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 3.040 13.0 0.2350 0.816  
## beta.1 0.551 14.0 0.0395 0.969  
## beta.2 -20.700 14.0 -1.4800 0.151  
## beta.3 11.700 12.9 0.9040 0.375

summary(PolyDLM\_model\_FFD\_rad)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -57.162 -12.623 -0.322 17.604 37.509   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 288.35 230.78 1.249 0.224  
## z.t0 3.04 12.96 0.235 0.817  
## z.t1 31.02 69.57 0.446 0.660  
## z.t2 -45.57 62.52 -0.729 0.473  
## z.t3 12.06 13.82 0.873 0.392  
##   
## Residual standard error: 24.32 on 23 degrees of freedom  
## Multiple R-squared: 0.1098, Adjusted R-squared: -0.045   
## F-statistic: 0.7093 on 4 and 23 DF, p-value: 0.5939

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 0.7093  
R - squared is 0.1098

Adjusted R - squared is -0.045

Degrees of freedom - DF are (4, 23)

p - value (0.5939) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Polynomial distributed lag model.

Also, this model suggests that there is only -4.5% of data variance.

No residual analysis is required.

Let us fit Koyck model.

#### Koyck model with Radiation

# Koyk DLM  
  
Koyck\_DLM\_FFD\_rad = koyckDlm(x = as.vector(x2) , y = as.vector(y))  
summary(Koyck\_DLM\_FFD\_rad)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -55.229 -19.662 3.956 16.232 54.756   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 418.31843 384.12678 1.089 0.286  
## Y.1 -0.03254 0.20729 -0.157 0.876  
## X.t -13.87153 25.49529 -0.544 0.591  
##   
## Residual standard error: 25.54 on 27 degrees of freedom  
## Multiple R-Squared: -0.07436, Adjusted R-squared: -0.1539   
## Wald test: 0.1486 on 2 and 27 DF, p-value: 0.8626   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: 405.1353 -13.87153 -0.03254005

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

Wald test statistic is 0.1486  
R - squared is -0.07436

Adjusted R - squared is -0.1539

Degrees of freedom - DF are (2, 27)

p - value (0.8626) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Koyck distributed lag model.

Also, this model suggests that there is only -15.39% of data variance.

No residual analysis is required.

Let us fit ardlDlm model to check whether it fits better or not.

#### Autoregressive distributed lag model with Radiation

##### with slope

for (i in 1:5){  
 for(j in 1:5){  
 model\_4 = ardlDlm(formula=y ~ x4, data=data.frame(v\_data\_TS\_22), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_4$model), "BIC = ", BIC(model\_4$model), "MASE =", MASE(model\_4)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = 283.5658 BIC = 290.5718 MASE = 0.6599906   
## p = 1 q = 2 AIC = 276.5262 BIC = 284.73 MASE = 0.6718305   
## p = 1 q = 3 AIC = 269.1591 BIC = 278.4845 MASE = 0.6531487   
## p = 1 q = 4 AIC = 262.6904 BIC = 273.0571 MASE = 0.6386406   
## p = 1 q = 5 AIC = 255.6012 BIC = 266.9241 MASE = 0.6196376   
## p = 2 q = 1 AIC = 275.24 BIC = 283.4438 MASE = 0.6688985   
## p = 2 q = 2 AIC = 277.1882 BIC = 286.7593 MASE = 0.6661641   
## p = 2 q = 3 AIC = 269.5463 BIC = 280.204 MASE = 0.646406   
## p = 2 q = 4 AIC = 262.9001 BIC = 274.5626 MASE = 0.6221685   
## p = 2 q = 5 AIC = 256.4326 BIC = 269.0136 MASE = 0.6306393   
## p = 3 q = 1 AIC = 266.5357 BIC = 275.8611 MASE = 0.6144131   
## p = 3 q = 2 AIC = 268.3699 BIC = 279.0276 MASE = 0.6149087   
## p = 3 q = 3 AIC = 270.1803 BIC = 282.1701 MASE = 0.6220337   
## p = 3 q = 4 AIC = 263.6351 BIC = 276.5935 MASE = 0.6016041   
## p = 3 q = 5 AIC = 257.3509 BIC = 271.1899 MASE = 0.6164068   
## p = 4 q = 1 AIC = 259.9163 BIC = 270.283 MASE = 0.5983539   
## p = 4 q = 2 AIC = 261.3427 BIC = 273.0052 MASE = 0.5933603   
## p = 4 q = 3 AIC = 263.3127 BIC = 276.2711 MASE = 0.5959315   
## p = 4 q = 4 AIC = 265.2261 BIC = 279.4803 MASE = 0.5907476   
## p = 4 q = 5 AIC = 258.7707 BIC = 273.8679 MASE = 0.6045879   
## p = 5 q = 1 AIC = 251.3746 BIC = 262.6975 MASE = 0.587024   
## p = 5 q = 2 AIC = 252.9746 BIC = 265.5556 MASE = 0.5756885   
## p = 5 q = 3 AIC = 254.7 BIC = 268.5391 MASE = 0.5539405   
## p = 5 q = 4 AIC = 255.7433 BIC = 270.8405 MASE = 0.5278081   
## p = 5 q = 5 AIC = 257.5483 BIC = 273.9036 MASE = 0.5314267

(p, q) = (5, 4) has the least AIC, BIC and MASE scores.

# ARDLM model  
AR\_DLM\_FFD\_54\_slope\_rad = ardlDlm(formula = y ~ x4, data=data.frame(v\_data\_TS\_22), p = 5 , q = 4)  
summary(AR\_DLM\_FFD\_54\_slope\_rad)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -47.589 -7.045 0.804 12.449 32.637   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -2.903e+02 4.908e+02 -0.591 0.563  
## x4.t 1.353e+01 1.929e+01 0.702 0.494  
## x4.1 9.354e+00 2.091e+01 0.447 0.661  
## x4.2 -3.163e+01 1.882e+01 -1.681 0.114  
## x4.3 1.114e+01 1.984e+01 0.561 0.583  
## x4.4 -4.193e+00 2.204e+01 -0.190 0.852  
## x4.5 2.912e+01 2.014e+01 1.446 0.169  
## y.1 -8.686e-03 2.456e-01 -0.035 0.972  
## y.2 1.181e-01 2.663e-01 0.443 0.664  
## y.3 1.556e-01 3.111e-01 0.500 0.624  
## y.4 2.052e-01 2.737e-01 0.750 0.465  
##   
## Residual standard error: 27.46 on 15 degrees of freedom  
## Multiple R-squared: 0.2501, Adjusted R-squared: -0.2498   
## F-statistic: 0.5004 on 10 and 15 DF, p-value: 0.8644

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model with slope.**

**HA: The data fits the Autoregressive distributed lag model with slope.**

**Interpretations:**

F - statistic is 0.5004  
R - squared is 0.2501

Adjusted R - squared is -0.2498  
Degrees of freedom - DF are (10, 15)

p - value (0.8644) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Autoregressive distributed lag model with slope.

Also, this model suggests that there is only -24.98% of data variance.

No residual analysis is required.

##### With out slope.

for (i in 1:5){  
 for(j in 1:5){  
 model\_5 = ardlDlm(formula=y ~ 0 + x4, data=data.frame(v\_data\_TS\_22), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_5$model), "BIC = ", BIC(model\_5$model), "MASE =", MASE(model\_5)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = 283.5658 BIC = 290.5718 MASE = 0.6599906   
## p = 1 q = 2 AIC = 276.5262 BIC = 284.73 MASE = 0.6718305   
## p = 1 q = 3 AIC = 269.1591 BIC = 278.4845 MASE = 0.6531487   
## p = 1 q = 4 AIC = 262.6904 BIC = 273.0571 MASE = 0.6386406   
## p = 1 q = 5 AIC = 255.6012 BIC = 266.9241 MASE = 0.6196376   
## p = 2 q = 1 AIC = 275.24 BIC = 283.4438 MASE = 0.6688985   
## p = 2 q = 2 AIC = 277.1882 BIC = 286.7593 MASE = 0.6661641   
## p = 2 q = 3 AIC = 269.5463 BIC = 280.204 MASE = 0.646406   
## p = 2 q = 4 AIC = 262.9001 BIC = 274.5626 MASE = 0.6221685   
## p = 2 q = 5 AIC = 256.4326 BIC = 269.0136 MASE = 0.6306393   
## p = 3 q = 1 AIC = 266.5357 BIC = 275.8611 MASE = 0.6144131   
## p = 3 q = 2 AIC = 268.3699 BIC = 279.0276 MASE = 0.6149087   
## p = 3 q = 3 AIC = 270.1803 BIC = 282.1701 MASE = 0.6220337   
## p = 3 q = 4 AIC = 263.6351 BIC = 276.5935 MASE = 0.6016041   
## p = 3 q = 5 AIC = 257.3509 BIC = 271.1899 MASE = 0.6164068   
## p = 4 q = 1 AIC = 259.9163 BIC = 270.283 MASE = 0.5983539   
## p = 4 q = 2 AIC = 261.3427 BIC = 273.0052 MASE = 0.5933603   
## p = 4 q = 3 AIC = 263.3127 BIC = 276.2711 MASE = 0.5959315   
## p = 4 q = 4 AIC = 265.2261 BIC = 279.4803 MASE = 0.5907476   
## p = 4 q = 5 AIC = 258.7707 BIC = 273.8679 MASE = 0.6045879   
## p = 5 q = 1 AIC = 251.3746 BIC = 262.6975 MASE = 0.587024   
## p = 5 q = 2 AIC = 252.9746 BIC = 265.5556 MASE = 0.5756885   
## p = 5 q = 3 AIC = 254.7 BIC = 268.5391 MASE = 0.5539405   
## p = 5 q = 4 AIC = 255.7433 BIC = 270.8405 MASE = 0.5278081   
## p = 5 q = 5 AIC = 257.5483 BIC = 273.9036 MASE = 0.5314267

(p, q) = (5, 4) has the least AIC, BIC and MASE scores.

# Finite lag length based on AIC-BIC  
  
AR\_DLM\_FFD\_54\_noslope\_rad = ardlDlm(formula=y ~ 0 + x4, data=data.frame(v\_data\_TS\_22), p = 5 , q = 4)  
summary(AR\_DLM\_FFD\_54\_noslope\_rad)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -47.589 -7.045 0.804 12.449 32.637   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -2.903e+02 4.908e+02 -0.591 0.563  
## x4.t 1.353e+01 1.929e+01 0.702 0.494  
## x4.1 9.354e+00 2.091e+01 0.447 0.661  
## x4.2 -3.163e+01 1.882e+01 -1.681 0.114  
## x4.3 1.114e+01 1.984e+01 0.561 0.583  
## x4.4 -4.193e+00 2.204e+01 -0.190 0.852  
## x4.5 2.912e+01 2.014e+01 1.446 0.169  
## y.1 -8.686e-03 2.456e-01 -0.035 0.972  
## y.2 1.181e-01 2.663e-01 0.443 0.664  
## y.3 1.556e-01 3.111e-01 0.500 0.624  
## y.4 2.052e-01 2.737e-01 0.750 0.465  
##   
## Residual standard error: 27.46 on 15 degrees of freedom  
## Multiple R-squared: 0.2501, Adjusted R-squared: -0.2498   
## F-statistic: 0.5004 on 10 and 15 DF, p-value: 0.8644

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model without slope.**

**HA: The data fits the Autoregressive distributed lag model without slope.**

**Interpretations:**

F - statistic is 0.5004  
R - squared is 0.2501

Adjusted R - squared is -0.2498  
Degrees of freedom - DF are (10, 15)

p - value (0.8644) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Autoregressive distributed lag model with slope.

Also, this model suggests that there is only -24.98% of data variance.

No residual analysis is required.

Therefore, Finite DLM without slope w.r.t radiation fits better.

Now let us fit dynamic lm model

#### Dynamic model

##### Rainfall

###### With slope

v\_FFd\_dyna\_rain <- dynlm(y ~ x3, data = data.frame(v\_data\_TS\_22))  
summary(v\_FFd\_dyna\_rain)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ x3, data = data.frame(v\_data\_TS\_22))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -55.774 -16.601 3.458 17.127 55.805   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 201.919 27.888 7.240 5.68e-08 \*\*\*  
## x3 3.178 11.627 0.273 0.787   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.78 on 29 degrees of freedom  
## Multiple R-squared: 0.00257, Adjusted R-squared: -0.03182   
## F-statistic: 0.07471 on 1 and 29 DF, p-value: 0.7865

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model with slope.**

**HA: The data fits the Dynamic linear model with slope.**

**Interpretations:**

F - statistic is 0.07471  
R - squared is 0.00257

Adjusted R - squared is -0.03182  
Degrees of freedom - DF are (1, 29)

p - value (0.7865) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model doesn’t fit the Dynamic linear model with slope.

This model suggests that there is only -3.18% of data variance.

No residual analysis is required.

Therefore, Further analysis is needed by removing slope to the lag model.

###### With out slope.

v\_FFd\_dyna\_rain\_noslope <- dynlm(y ~ 0 + x3, data = data.frame(v\_data\_TS\_22))  
summary(v\_FFd\_dyna\_rain\_noslope)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ 0 + x3, data = data.frame(v\_data\_TS\_22))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -59.484 -18.121 1.731 22.895 93.445   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## x3 86.366 2.934 29.44 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 39.18 on 30 degrees of freedom  
## Multiple R-squared: 0.9665, Adjusted R-squared: 0.9654   
## F-statistic: 866.6 on 1 and 30 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model without slope.**

**HA: The data fits the Dynamic linear model without slope.**

**Interpretations:**

F - statistic is 866.6  
R - squared is 0.9665

Adjusted R - squared is 0.9654  
Degrees of freedom - DF are (1, 30)

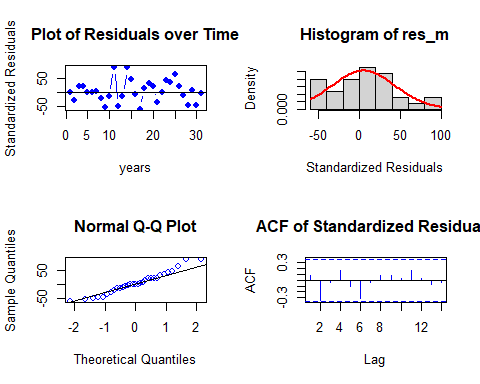
p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Dynamic linear model.

This model suggests that there is only 96.54% of data variance. Suggesting that the model explains only 96,54% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(v\_FFd\_dyna\_rain\_noslope))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.96579, p-value = 0.4113

Residual Analysis for v\_FFd\_dyna\_rain\_noslope:

1. The data points are below the line at both the start and end of the trend. Randomness is not seen here.
2. From normal distribution curve, the distribution is almost symmetric. This suggests a good fit with a very few data falling outside the normal curve indicating Kurtosis.
3. The data at the tails is deviated but is normal for most part of the line suggesting normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.4113) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Even though the model fits better it is poor when it comes to residual analysis. Let us fit with Radiation.

##### Radiation

###### With slope

v\_FFd\_dyna\_rad <- dynlm(y ~ x4, data = data.frame(v\_data\_TS\_22))  
summary(v\_FFd\_dyna\_rad)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ x4, data = data.frame(v\_data\_TS\_22))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -55.625 -15.751 2.051 17.086 54.734   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 169.916 156.834 1.083 0.288  
## x4 2.709 10.744 0.252 0.803  
##   
## Residual standard error: 23.79 on 29 degrees of freedom  
## Multiple R-squared: 0.002188, Adjusted R-squared: -0.03222   
## F-statistic: 0.0636 on 1 and 29 DF, p-value: 0.8027

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model with slope.**

**HA: The data fits the Dynamic linear model with slope.**

**Interpretations:**

F - statistic is 0.0636  
R - squared is 0.002188

Adjusted R - squared is -0.03222  
Degrees of freedom - DF are (1, 29)

p - value (0.8027) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model doesn’t fit the Dynamic linear model with slope.

This model suggests that there is only -3.22% of data variance.

No residual analysis is required.

Therefore, Further analysis is needed by removing slope to the lag model.

###### With out slope.

v\_FFd\_dyna\_rad\_noslope <- dynlm(y ~ 0 + x4, data = data.frame(v\_data\_TS\_22))  
summary(v\_FFd\_dyna\_rad\_noslope)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ 0 + x4, data = data.frame(v\_data\_TS\_22))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -56.243 -19.058 2.424 17.458 55.657   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## x4 14.3455 0.2935 48.87 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.86 on 30 degrees of freedom  
## Multiple R-squared: 0.9876, Adjusted R-squared: 0.9872   
## F-statistic: 2388 on 1 and 30 DF, p-value: < 2.2e-16

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model without slope.**

**HA: The data fits the Dynamic linear model without slope.**

**Interpretations:**

F - statistic is 2388  
R - squared is 0.9876

Adjusted R - squared is 0.9872  
Degrees of freedom - DF are (1, 30)

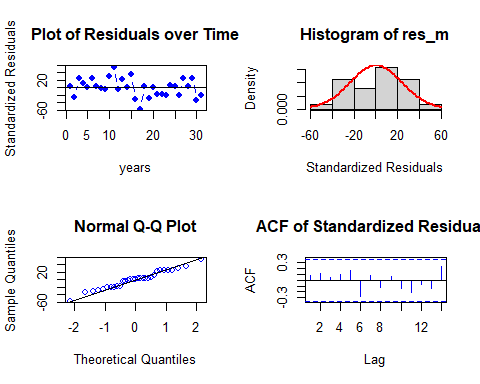
p - value (~ 0.01) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Dynamic linear model without slope.

This model suggests that there is only 98.72% of data variance. Suggesting that the model explains only 98.72% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(v\_FFd\_dyna\_rad\_noslope))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.97519, p-value = 0.6705

Residual Analysis for v\_FFd\_dyna\_rad\_noslope:

1. The data points are below the line at both the start and end of the trend. Randomness is not seen here.
2. From normal distribution curve, the distribution is almost symmetric. This suggests a good fit with a very few data falling outside the normal curve indicating Kurtosis.
3. The data at the tails is deviated but is normal for most part of the line suggesting normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.6705) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Therefore, Dynamic model without slope w.r.t radiation fits better.

### Exponential Smoothing

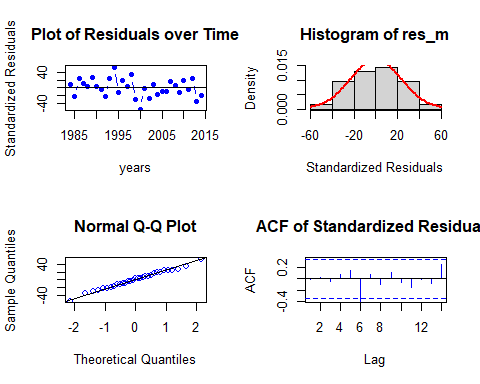
Since the Seasonality component is week we cannot get additive and multiplicative seasonality. So let us fit with simple seasonality. Since, we need next 3 years point forecasts as well as confidence intervals, we used h = 3 (frequency).

fit\_ses <- ses(v\_First\_Flowering\_Day\_data\_TS, seasonal = "simple", h = 4)  
summary(fit\_ses)

##   
## Forecast method: Simple exponential smoothing  
##   
## Model Information:  
## Simple exponential smoothing   
##   
## Call:  
## ses(y = v\_First\_Flowering\_Day\_data\_TS, h = 4, seasonal = "simple")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
##   
## Initial states:  
## l = 209.4505   
##   
## sigma: 23.8149  
##   
## AIC AICc BIC   
## 306.9455 307.8343 311.2474   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.003678178 23.03386 18.69592 -1.26477 9.173671 0.6614123  
## ACF1  
## Training set -0.006864134  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2015 209.4505 178.9305 239.9705 162.7742 256.1268  
## 2016 209.4505 178.9305 239.9705 162.7742 256.1268  
## 2017 209.4505 178.9305 239.9705 162.7742 256.1268  
## 2018 209.4505 178.9305 239.9705 162.7742 256.1268

Now let us check the residual analysis.

res\_analysis(residuals(fit\_ses))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.99329, p-value = 0.9992

Residual Analysis analysis for simple seasonality:

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

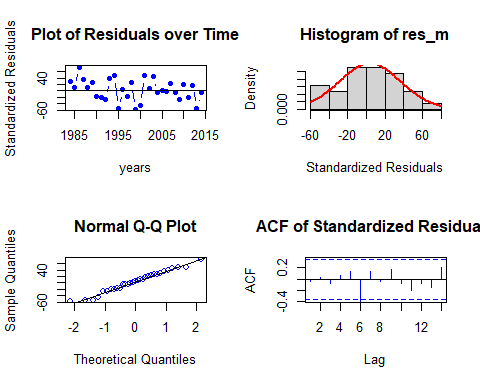
Now let us fit with simple seasonality with both alpha and gamma.

fit2 <- holt(v\_First\_Flowering\_Day\_data\_TS, initial = "simple", h = 4)  
  
#both alpha and beta  
summary(fit2)

##   
## Forecast method: Holt's method  
##   
## Model Information:  
## Holt's method   
##   
## Call:  
## holt(y = v\_First\_Flowering\_Day\_data\_TS, h = 4, initial = "simple")   
##   
## Smoothing parameters:  
## alpha = 0.4947   
## beta = 0.4105   
##   
## Initial states:  
## l = 217   
## b = -31   
##   
## sigma: 32.4048  
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 3.643616 32.40477 26.33733 0.4670524 12.74033 0.9317453  
## ACF1  
## Training set -0.03620177  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2015 183.4402 141.91186 224.9686 119.928060 246.9524  
## 2016 175.3764 119.36198 231.3909 89.709726 261.0431  
## 2017 167.3126 89.06345 245.5618 47.640838 286.9844  
## 2018 159.2488 53.12709 265.3705 -3.050347 321.5479

Now let us check the residual analysis.

res\_analysis(residuals(fit2))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.9795, p-value = 0.7985

Residual Analysis analysis for simple seasonality:

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

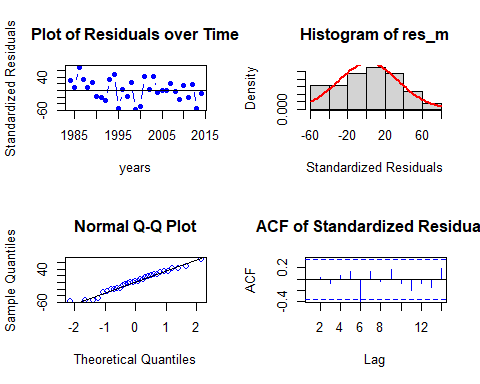
Now let us fit with exponential trend.

fit3 <- holt(v\_First\_Flowering\_Day\_data\_TS, initial="simple", exponential = TRUE, h = 4)# Fit with exponential trend  
summary(fit3)

##   
## Forecast method: Holt's method with exponential trend  
##   
## Model Information:  
## Holt's method with exponential trend   
##   
## Call:  
## holt(y = v\_First\_Flowering\_Day\_data\_TS, h = 4, initial = "simple",   
##   
## Call:  
## exponential = TRUE)   
##   
## Smoothing parameters:  
## alpha = 0.4502   
## beta = 0.4436   
##   
## Initial states:  
## l = 217   
## b = 0.8571   
##   
## sigma: 0.1598  
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 1.966628 31.73873 25.84104 -0.3370878 12.55847 0.9141876  
## ACF1  
## Training set -0.01314563  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2015 185.9214 146.63995 223.6395 127.70016 244.4981  
## 2016 178.6954 131.37582 230.5629 109.28928 261.5664  
## 2017 171.7503 112.46477 244.9998 87.76724 295.4699  
## 2018 165.0750 91.71386 268.4287 67.21901 357.7409

Now let us check the residual analysis.

res\_analysis(residuals(fit3))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.97835, p-value = 0.7654

Residual Analysis analysis for exponential trend:

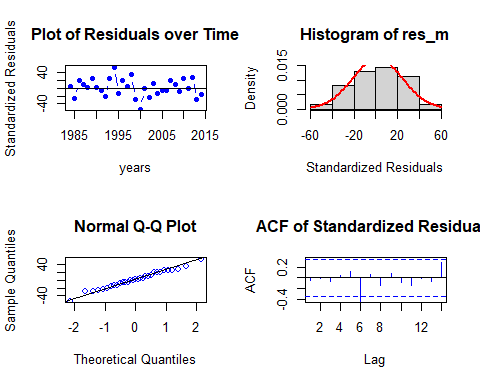
1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

fit4 <- holt(v\_First\_Flowering\_Day\_data\_TS, damped = TRUE, h = 4)# Fit with damped trend  
summary(fit4)

##   
## Forecast method: Damped Holt's method  
##   
## Model Information:  
## Damped Holt's method   
##   
## Call:  
## holt(y = v\_First\_Flowering\_Day\_data\_TS, h = 4, damped = TRUE)   
##   
## Smoothing parameters:  
## alpha = 1e-04   
## beta = 1e-04   
## phi = 0.9724   
##   
## Initial states:  
## l = 213.8104   
## b = -0.5047   
##   
## sigma: 24.6844  
##   
## AIC AICc BIC   
## 311.7836 315.2836 320.3875   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 1.646287 22.60624 17.87068 -0.4309696 8.698589 0.6322173  
## ACF1  
## Training set -0.0440588  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 2015 203.4051 171.7707 235.0394 155.0245 251.7856  
## 2016 203.2069 171.5725 234.8412 154.8263 251.5874  
## 2017 203.0142 171.3798 234.6485 154.6336 251.3947  
## 2018 202.8268 171.1924 234.4611 154.4462 251.2073

Now let us check the residual analysis.

res\_analysis(residuals(fit4))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.98975, p-value = 0.9885

Residual Analysis analysis for exponential trend:

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Due to week seasonality in the series there is no additive or multiplicative seasonality also there will be no damped in the series.

By exponential smoothing method we got the simple seasonal fit as the best model in terms of MASE and BIC scores.

### State Space Model Variations

Let us find the best ets model. Before all let us auto fit the model.

v\_ets\_fit <- ets(v\_First\_Flowering\_Day\_data\_TS)  
summary(v\_ets\_fit)

## ETS(M,N,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS)   
##   
## Smoothing parameters:  
## alpha = 1e-04   
##   
## Initial states:  
## l = 209.448   
##   
## sigma: 0.1137  
##   
## AIC AICc BIC   
## 306.9448 307.8337 311.2468   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.001214657 23.03386 18.696 -1.263579 9.173601 0.6614151  
## ACF1  
## Training set -0.006864096

ETS(M, N, N)

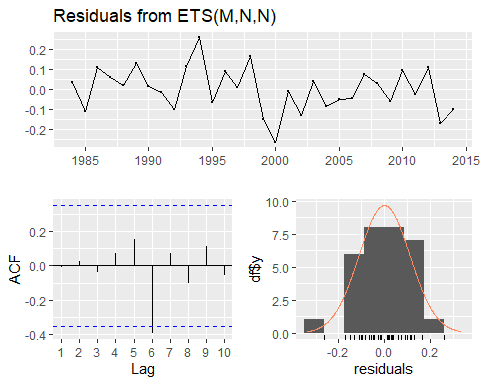
M - Multiplicative errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,N,N)  
## Q\* = 7.489, df = 4, p-value = 0.1122  
##   
## Model df: 2. Total lags used: 6

Residual Analysis ETS(A, AD, A):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Now let us fit ets variable combinations individually.

v\_ets\_fit1 <- ets(v\_First\_Flowering\_Day\_data\_TS, model = "ANN")  
summary(v\_ets\_fit1)

## ETS(A,N,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS, model = "ANN")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
##   
## Initial states:  
## l = 209.4516   
##   
## sigma: 23.8149  
##   
## AIC AICc BIC   
## 306.9455 307.8343 311.2474   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.004800665 23.03386 18.69589 -1.265312 9.173703 0.661411  
## ACF1  
## Training set -0.006864147

ETS(A, N, N)

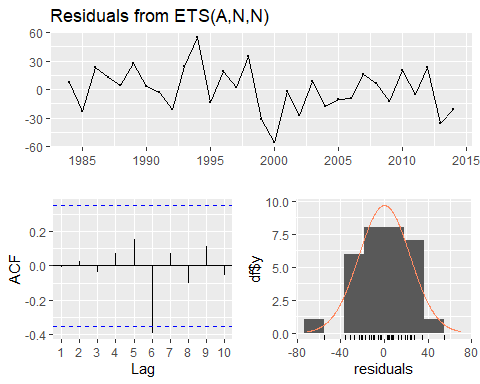
A - Additive errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit1)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(A,N,N)  
## Q\* = 7.489, df = 4, p-value = 0.1122  
##   
## Model df: 2. Total lags used: 6

Residual Analysis ETS(A, N, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit2 <- ets(v\_First\_Flowering\_Day\_data\_TS, model = "AAN")  
summary(v\_ets\_fit2)

## ETS(A,A,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS, model = "AAN")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
## beta = 1e-04   
##   
## Initial states:  
## l = 218.839   
## b = -0.5898   
##   
## sigma: 24.0066  
##   
## AIC AICc BIC   
## 309.2274 311.6274 316.3973   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 0.04681346 22.40433 17.52335 -1.174704 8.595767 0.6199298  
## ACF1  
## Training set -0.05273509

ETS(A, A, N)

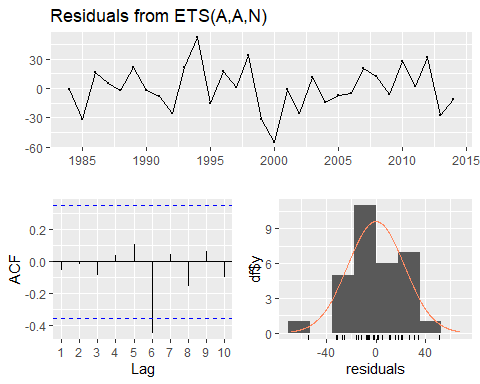
A - Additive errors

A - Additive trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit2)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(A,A,N)  
## Q\* = 9.1235, df = 3, p-value = 0.02769  
##   
## Model df: 4. Total lags used: 7

Residual Analysis ETS(A, A, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit3 <- ets(v\_First\_Flowering\_Day\_data\_TS, model = "MNN")  
summary(v\_ets\_fit3)

## ETS(M,N,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS, model = "MNN")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
##   
## Initial states:  
## l = 209.448   
##   
## sigma: 0.1137  
##   
## AIC AICc BIC   
## 306.9448 307.8337 311.2468   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -0.001214657 23.03386 18.696 -1.263579 9.173601 0.6614151  
## ACF1  
## Training set -0.006864096

ETS(A, A, N)

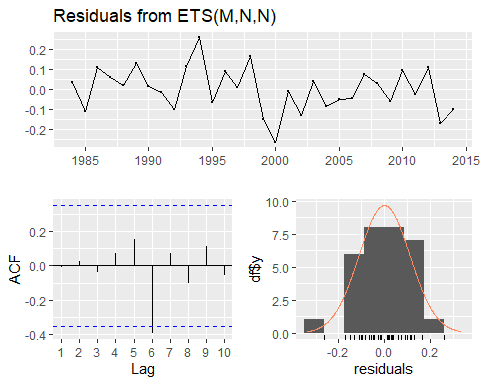
M - Multiplicative errors

N - No trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit3)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,N,N)  
## Q\* = 7.489, df = 4, p-value = 0.1122  
##   
## Model df: 2. Total lags used: 6

Residual Analysis ETS(M, N, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit4 <- ets(v\_First\_Flowering\_Day\_data\_TS, model = "MAN")  
summary(v\_ets\_fit4)

## ETS(M,A,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS, model = "MAN")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
## beta = 1e-04   
##   
## Initial states:  
## l = 215.3797   
## b = -0.4305   
##   
## sigma: 0.1155  
##   
## AIC AICc BIC   
## 309.4179 311.8179 316.5879   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 0.9196759 22.47637 17.60976 -0.7708524 8.605156 0.6229869  
## ACF1  
## Training set -0.04986601

ETS(A, A, N)

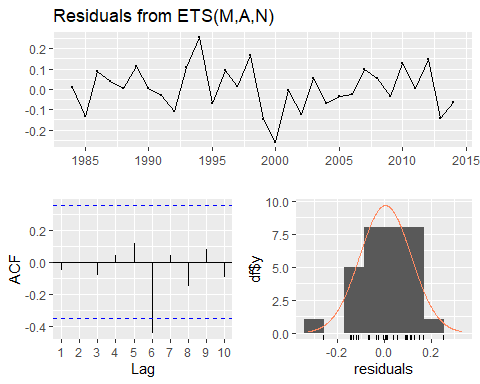
M - Multiplicative errors

A - Additive trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit4)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,A,N)  
## Q\* = 8.9614, df = 3, p-value = 0.02981  
##   
## Model df: 4. Total lags used: 7

Residual Analysis ETS(M, A, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

v\_ets\_fit5 <- ets(v\_First\_Flowering\_Day\_data\_TS, model = "MMN")  
summary(v\_ets\_fit5)

## ETS(M,M,N)   
##   
## Call:  
## ets(y = v\_First\_Flowering\_Day\_data\_TS, model = "MMN")   
##   
## Smoothing parameters:  
## alpha = 1e-04   
## beta = 1e-04   
##   
## Initial states:  
## l = 216.4878   
## b = 0.9977   
##   
## sigma: 0.1153  
##   
## AIC AICc BIC   
## 309.3333 311.7333 316.5032   
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 0.8316197 22.443 17.48424 -0.8068944 8.548035 0.6185461  
## ACF1  
## Training set -0.05190951

ETS(M, M, N)

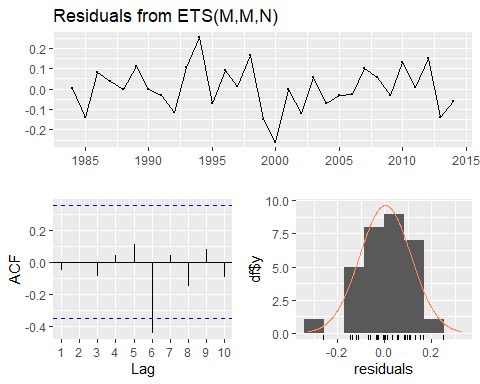
M - Multiplicative errors

M - Multiplicative trend

N - No seasonality.

Let us perform residual analysis on this ETS model.

checkresiduals(v\_ets\_fit5)



##   
## Ljung-Box test  
##   
## data: Residuals from ETS(M,M,N)  
## Q\* = 9.0495, df = 3, p-value = 0.02864  
##   
## Model df: 4. Total lags used: 7

Residual Analysis ETS(M, M, N):

1. The data points are below the line at the start and below the line at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (~ 0.01) from Shapiro-Wilk normality test is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Suggesting some normality.

Comparitively, based on MASE and RMSE score ets(M, N, N) is better.

Among all the methods, holt in exponential smoothing with damped trend.

### Forecasting

Let us forecast for the next 4 years on First Flowering Day series. From 2015 to 2018. For the optimal model from each method.

#### Forcasting with Smoothing method best model

fit <- holt(v\_First\_Flowering\_Day\_data\_TS, damped = TRUE, h = 4)  
  
v\_FFD\_forecasts <- ts.intersect(ts(fit$lower[, 2], start = c(2015), frequency = 1), ts(fit$mean, start = c(2015), frequency = 1), ts(fit$upper[, 2], start = c(2015), frequency = 1))  
colnames(v\_FFD\_forecasts) <- c("Lower bound", "Point forecast", "Upper bound")  
  
v\_FFD\_forecasts

## Time Series:  
## Start = 2015   
## End = 2018   
## Frequency = 1   
## Lower bound Point forecast Upper bound  
## 2015 155.0245 203.4051 251.7856  
## 2016 154.8263 203.2069 251.5874  
## 2017 154.6336 203.0142 251.3947  
## 2018 154.4462 202.8268 251.2073

Now let us plot the forecast.

plot(fit, fcol = "white", main = "Forecast of First Flowering Day series for the next 3 years (2015, 2017)", ylab = "First Flowering Day")  
lines(fitted(fit), col = "red")  
lines(fit$mean, col = "blue", lwd = 2)  
legend("bottom", inset = .03, cex = 0.9, box.lty = 2, box.lwd = 2, pch = 1, lty = 1, col = c("red", "blue"), c("Data", "Forecasts"))

Chart, line chart

Description automatically generated

Fig 2.21: Next 4 years forecast on the First Flowering Day Series with Smoothing method model.

From the four year forecast results we can predict that there will be decrease in the First Flowering Day in the future. This suggests that the impact of the chemical components will be less in future.

#### Forcasting with ets method best model

fit3 <- ets(v\_First\_Flowering\_Day\_data\_TS, model="MMN", damped = T)  
fit1 <- forecast.ets(fit3, h = 4)  
plot(fit1, fcol = "white", main = "First Flowerind Day series with 4 years ahead forecasts", ylab = "Radiation")  
lines(fitted(fit1), col = "darkgreen")  
lines(fit1$mean, col = "darkgreen", lwd = 2)  
legend("topleft", lty = 1, col = c("black", "darkgreen"), c("Data", "Forecasts"))

Chart, line chart

Description automatically generated

Fig 2.22: Next 4 years forecast on the First Flowering Day Series with ets method model.

forecasts <- ts.intersect(ts(fit1$lower[, 2], start = c(2015), frequency = 1), ts(fit1$mean, start = c(2015), frequency = 1), ts(fit1$upper[, 2], start = c(2015), frequency = 1))  
colnames(forecasts) <- c("Lower bound", "Point forecast", "Upper bound")  
forecasts

## Time Series:  
## Start = 2015   
## End = 2018   
## Frequency = 1   
## Lower bound Point forecast Upper bound  
## 2015 157.2584 204.2685 251.0862  
## 2016 154.4180 204.0840 252.5719  
## 2017 155.1222 203.9041 249.3414  
## 2018 154.7424 203.7289 251.1135

From the four year forecast results we can predict that there will be decrease in the First Flowering Day in the future. This suggests that the impact of the chemical components will be less in future.

## Task 3

### Part (a)

#### Data

The data here used is the the contemporaneous yearly averaged climate variables measured from 1984 – 2014 (31 years).

v\_RBO\_data <- read.csv("RBO.csv", header = TRUE)  
head(v\_RBO\_data)

## ï..Year RBO Temperature Rainfall Radiation RelHumidity  
## 1 1984 0.7550088 9.371585 2.489344 14.87158 93.92650  
## 2 1985 0.7407520 9.656164 2.475890 14.68493 94.93589  
## 3 1986 0.8423860 9.273973 2.421370 14.51507 94.09507  
## 4 1987 0.7484425 9.219178 2.319726 14.67397 94.49699  
## 5 1988 0.7984084 10.202186 2.465301 14.74863 94.08142  
## 6 1989 0.7938803 9.441096 2.735890 14.78356 96.08685

# Using str() to check the type of each column.  
str(v\_RBO\_data)

## 'data.frame': 31 obs. of 6 variables:  
## $ ï..Year : int 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 ...  
## $ RBO : num 0.755 0.741 0.842 0.748 0.798 ...  
## $ Temperature: num 9.37 9.66 9.27 9.22 10.2 ...  
## $ Rainfall : num 2.49 2.48 2.42 2.32 2.47 ...  
## $ Radiation : num 14.9 14.7 14.5 14.7 14.7 ...  
## $ RelHumidity: num 93.9 94.9 94.1 94.5 94.1 ...

Checking for Missing values.

colSums(is.na(v\_RBO\_data))

## ï..Year RBO Temperature Rainfall Radiation RelHumidity   
## 0 0 0 0 0 0

There are no missing values in the data.

Checking the class of v\_solar\_data. (It should be a data frame.)

class(v\_RBO\_data)

## [1] "data.frame"

v\_RBO\_Temp\_TS <- ts(v\_RBO\_data$Temperature, start = c(1984), frequency = 1)  
v\_RBO\_Rainfall\_TS <- ts(v\_RBO\_data$Rainfall, start = c(1984), frequency = 1)  
v\_RBO\_Radiation\_TS <- ts(v\_RBO\_data$Radiation, start = c(1984), frequency = 1)  
v\_RBO\_RelHumidity\_TS <- ts(v\_RBO\_data$RelHumidity, start = c(1984), frequency = 1)  
v\_RBO\_data\_TS <- ts(v\_RBO\_data$RBO, start = c(1984), frequency = 1)

Confirming the class of each time series object.

class(v\_RBO\_Temp\_TS)

## [1] "ts"

class(v\_RBO\_Rainfall\_TS)

## [1] "ts"

class(v\_RBO\_Radiation\_TS)

## [1] "ts"

class(v\_RBO\_RelHumidity\_TS)

## [1] "ts"

class(v\_RBO\_data\_TS)

## [1] "ts"

Now let us perform descriptive analysis on each time series object.

#### Descriptive Analysis

##### Rank-based Order similarity metric

plot(v\_RBO\_data\_TS, type = "b", xlab = "years", ylab = "Rank-based Order similarity metric", main = "Time series plot for yearly Rank-based Order similarity metric from 1984 – 2014 (31 years)", pch = 1)  
legend("topright", inset = .03, title = "Rank-based Order similarity metric", legend = "Rank-based Order similarity metric series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 3.1: Rank-based Order similarity metric - Time series plot.

McLeod.Li.test(y = v\_RBO\_data\_TS, main = "McLeod-Li Test Statistics for Rank-based Order similarity metric.")

Chart

Description automatically generated

Fig 3.2: McLeod-Li Test Statistics for Rank-based Order similarity metric.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around multiple years.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Temperature

plot(v\_RBO\_Temp\_TS, type = "b", xlab = "years", ylab = "Temperature", main = "Time series plot for yearly temperature from 1984 – 2014 (31 years)", pch = 1)  
legend("top", inset = .03, title = "Temperature", legend = "Temperature series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 3.3: Temperature - Time series plot.

McLeod.Li.test(y = v\_RBO\_Rainfall\_TS, main = "McLeod-Li Test Statistics for Temperature")

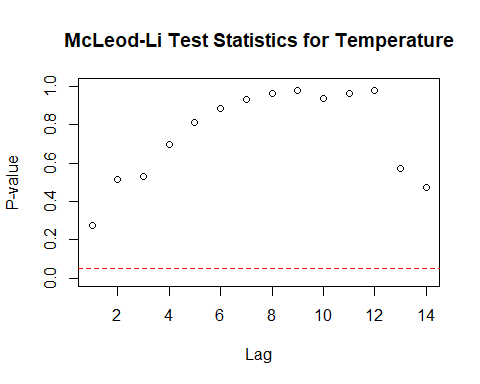


Fig 3.4: McLeod-Li Test Statistics for Temperature

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1996.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Rainfall

plot(v\_RBO\_Rainfall\_TS, type = "b", xlab = "years", ylab = "Rainfall", main = "Time series plot for yearly Rainfall from 1984 – 2014 (31 years)", pch = 1)  
legend("bottomleft", inset = .03, title = "Rainfall", legend = "Rainfall series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 3.5: Rainfall - Time series plot.

McLeod.Li.test(y = v\_RBO\_Rainfall\_TS, main = "McLeod-Li Test Statistics for Rainfall")

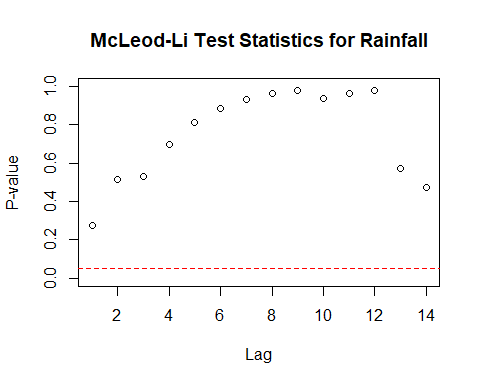


Fig 3.6: McLeod-Li Test Statistics for Rainfall

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1996.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Radiation

plot(v\_RBO\_Radiation\_TS, type = "b", xlab = "years", ylab = "Radiation", main = "Time series plot for yearly Radiation from 1984 – 2014 (31 years)", pch = 1)  
legend("topleft", inset = .03, title = "Radiation", legend = "Radiation series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 3.7: Radiation - Time series plot.

McLeod.Li.test(y = v\_RBO\_Rainfall\_TS, main = "McLeod-Li Test Statistics for Radiation")

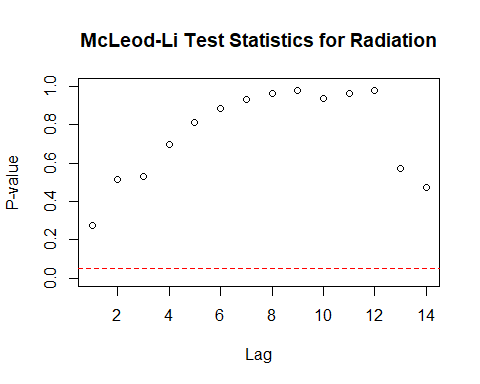


Fig 3.8: McLeod-Li Test Statistics for Radiation

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1992.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Relative Humidity

plot(v\_RBO\_RelHumidity\_TS, type = "b", xlab = "years", ylab = "Relative Humidity", main = "Time series plot for yearly Relative Humidity from 1984 – 2014 (31 years)", pch = 1)  
legend("bottomright", inset = .03, title = "Relative Humidity", legend = "Relative Humidity series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 3.9: Relative Humidity - Time series plot.

McLeod.Li.test(y = v\_RBO\_RelHumidity\_TS, main = "McLeod-Li Test Statistics for Relative Humidity")

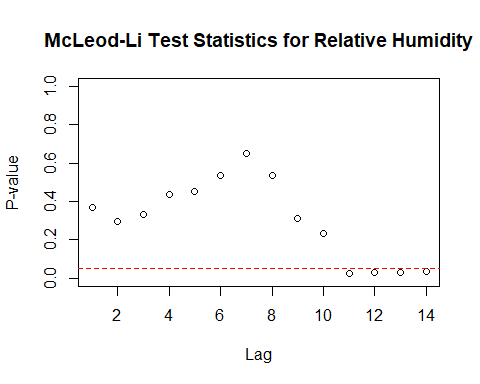


Fig 3.10: McLeod-Li Test Statistics for Relative Humidity.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1989.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Checking for Stationary in the series

Checking for Stationary on Rank-based Order similarity metric series.

Stationary\_Check(v\_RBO\_data\_TS, "Rank-based Order similarity metric - ACF plot", "Rank-based Order similarity metric - PACF plot")

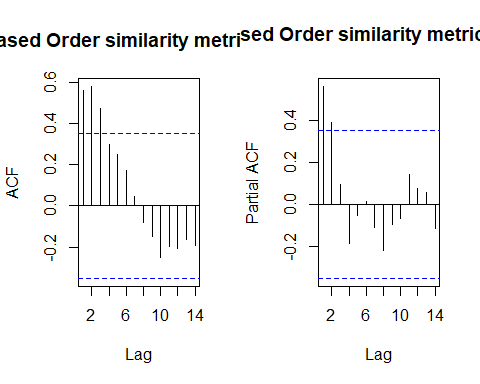


Fig 3.11: Rank-based Order similarity metric - ACF

Fig 3.12: Rank-based Order similarity metric - ACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -1.4542, Lag order = 2, p-value = 0.7829  
## alternative hypothesis: stationary

There are significant lags in the ACF and PACF plot suggesting the stochastic component is not white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.7829 > 0.05

p - value is greater than 0.05 and hence the test is not statistically significant. Therefore, Null hypothesis cannot be rejected i.e., The data is not stationary.

Therefore, the Rank - based Order similarity metric series is not Stationary.

Checking for Stationary on Temperature data.

Stationary\_Check(v\_RBO\_Temp\_TS, "Temperature - ACF plot", "Temperature - PACF plot")

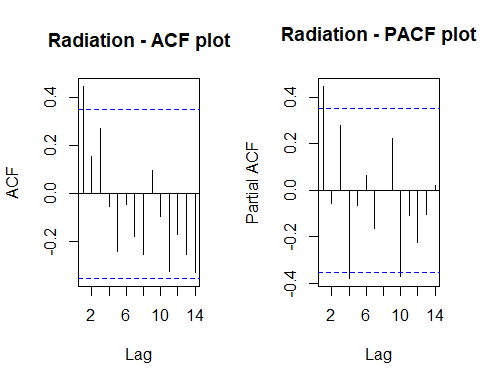


Fig 3.13: Temperature - ACF

Fig 3.14: Temperature - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -1.1484, Lag order = 2, p-value = 0.9002  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.9002 > 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Temperature series is Stationary.

Checking for Stationary on Radiation data.

Stationary\_Check(v\_RBO\_Radiation\_TS, "Radiation - ACF plot", "Radiation - PACF plot")

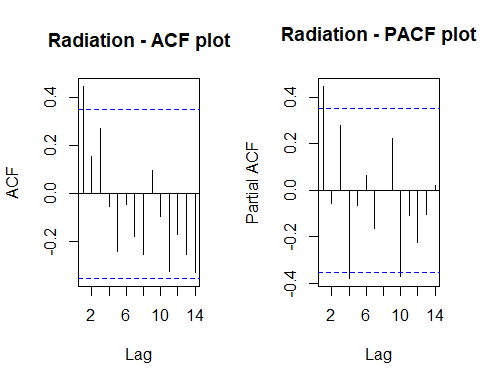


Fig 3.15: Radiation - ACF

Fig 3.16: Radiation - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -2.7317, Lag order = 4, p-value = 0.2911  
## alternative hypothesis: stationary

The is only one significant lag in the ACF and PACF plot.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.2911 > 0.05

p - value is greater than 0.05 and hence the test is not statistically significant. Therefore, Null hypothesis cannot be rejected i.e., The data is not stationary.

Therefore, the Radiation series is not Stationary.

Checking for Stationary on Rainfall data.

Stationary\_Check(v\_RBO\_Rainfall\_TS, "Rainfall - ACF plot", "Rainfall - PACF plot")

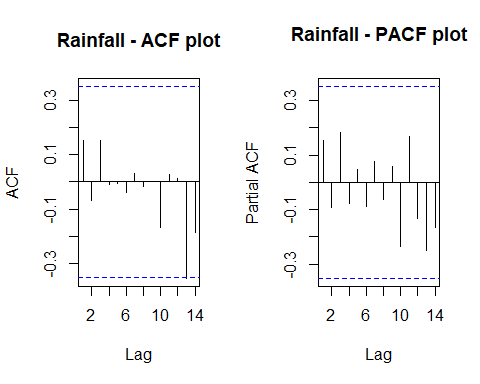


Fig 3.17: Rainfall - ACF

Fig 3.18: Rainfall - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5622, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Rainfall series is Stationary.

Checking for Stationary on Relative Humidity data.

Stationary\_Check(v\_RBO\_RelHumidity\_TS, "Relative Humidity - ACF plot", "Relative Humidity - PACF plot")

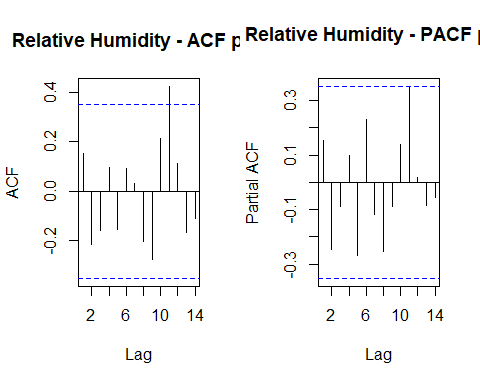


Fig 3.19: Relative Humidity - ACF

Fig 3.20: Relative Humidity - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5749, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Relative Humidity series is Stationary.

#### Suitable distributed lag model.

Before this let us find the correlation between the series.

# Calculating the correlation coefficient  
cor(v\_RBO\_data\_TS, v\_RBO\_Temp\_TS)

## [1] 0.2610007

cor(v\_RBO\_data\_TS, v\_RBO\_Rainfall\_TS)

## [1] 0.3932282

cor(v\_RBO\_data\_TS, v\_RBO\_Radiation\_TS)

## [1] -0.3173602

cor(v\_RBO\_data\_TS, v\_RBO\_RelHumidity\_TS)

## [1] -0.1776349

This suggests that FFD has a better correlation with Rainfall and Radiation.

As we are going to forecast the FFD data, our dependent variable “y” will be Mortality Rate series object and independent variable “x” will be Rainfall and Temperature.

##### Finite distributed lag model Rainfall

x1 = v\_RBO\_Rainfall\_TS # Independent variable  
x2 = v\_RBO\_Temp\_TS # Independent variable  
y = v\_RBO\_data\_TS # Dependent variable  
  
  
for ( i in 1:10){  
 model\_1 = dlm(x = as.vector(x1) , y = as.vector(y), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_1$model), "BIC = ", BIC(model\_1$model), "MASE =", MASE(model\_1)$MASE, "\n")  
 }

## q = 1 AIC = -100.898 BIC = -95.29319 MASE = 0.9417954   
## q = 2 AIC = -96.70956 BIC = -89.87308 MASE = 0.9993747   
## q = 3 AIC = -97.19966 BIC = -89.20643 MASE = 0.9796852   
## q = 4 AIC = -90.46187 BIC = -81.39101 MASE = 1.038827   
## q = 5 AIC = -87.24242 BIC = -77.17765 MASE = 0.925677   
## q = 6 AIC = -82.31788 BIC = -71.348 MASE = 0.8543964   
## q = 7 AIC = -77.98405 BIC = -66.20351 MASE = 0.829337   
## q = 8 AIC = -76.81922 BIC = -64.32879 MASE = 0.7233794   
## q = 9 AIC = -80.79432 BIC = -67.70181 MASE = 0.6205897   
## q = 10 AIC = -76.93255 BIC = -63.35376 MASE = 0.585617

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC-BIC-MASE  
  
finite\_dlm\_rain = dlm( x = as.vector(x1) , y = as.vector(y), q = 10)  
summary(finite\_dlm\_rain)

##   
## Call:  
## lm(formula = model.formula, data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.050274 -0.013229 -0.001445 0.015071 0.039030   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.709414 0.129781 5.466 0.000397 \*\*\*  
## x.t 0.016449 0.019478 0.845 0.420260   
## x.1 0.009100 0.017831 0.510 0.622079   
## x.2 0.014628 0.018404 0.795 0.447163   
## x.3 -0.006321 0.018174 -0.348 0.735980   
## x.4 -0.006181 0.020176 -0.306 0.766285   
## x.5 0.004570 0.020010 0.228 0.824453   
## x.6 -0.007054 0.019391 -0.364 0.724424   
## x.7 -0.011836 0.021444 -0.552 0.594424   
## x.8 0.004407 0.020473 0.215 0.834366   
## x.9 -0.021710 0.021728 -0.999 0.343817   
## x.10 0.006802 0.022752 0.299 0.771745   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.03187 on 9 degrees of freedom  
## Multiple R-squared: 0.3261, Adjusted R-squared: -0.4975   
## F-statistic: 0.3959 on 11 and 9 DF, p-value: 0.9251  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 -76.93255 -63.35376

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model.**

**HA: The data fits the Finite distributed lag model.**

**Interpretations:**

F - statistic is 0.3959  
R - squared is 0.3261

Adjusted R - squared is -0.4975

Degrees of freedom - DF are (11, 9)

p - value (0.9251) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Finite distributed lag model with slope.

No residual analysis is required.

Therefore, Further analysis is needed by adding polynomial to the lag model.

##### Polynomial distributed lag model with Rainfall

for (i in 1:3){  
 model\_3 <- polyDlm(x = as.vector(x1), y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_3$model), "BIC = ", BIC(model\_3$model), "MASE = ", MASE(model\_3)$MASE, "\n")  
}

## q = 1 k = 1 AIC = -100.898 BIC = -95.29319 MASE = 0.9417954   
## q = 2 k = 2 AIC = -96.70956 BIC = -89.87308 MASE = 0.9993747   
## q = 3 k = 3 AIC = -97.19966 BIC = -89.20643 MASE = 0.9796852

Let us fit a polynomial model of order 3. Since least AIC and BIC scores.

# Ploynomial DLM  
  
PolyDLM\_model\_Rain = polyDlm(x = as.vector(x1), y = as.vector(y), q = 1, k = 1, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 0.0420 0.0206 2.04 0.0505  
## beta.1 0.0303 0.0206 1.47 0.1520

summary(PolyDLM\_model\_Rain)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.105903 -0.024178 -0.006166 0.014773 0.099699   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.56594 0.06397 8.847 1.84e-09 \*\*\*  
## z.t0 0.04199 0.02058 2.040 0.0512 .   
## z.t1 -0.01167 0.03126 -0.373 0.7117   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04153 on 27 degrees of freedom  
## Multiple R-squared: 0.2156, Adjusted R-squared: 0.1575   
## F-statistic: 3.711 on 2 and 27 DF, p-value: 0.03767

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 3.711  
R - squared is 0.2156

Adjusted R - squared is 0.1575

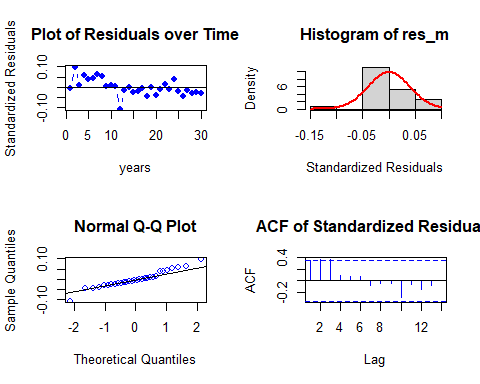
Degrees of freedom - DF are (4, 500)

p - value (0.03767) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Polynomial distributed lag model.

This model suggests that there is only 15.75% of data variance. Suggesting that the model explains only 15.75% of the trend. Which implies that the model shows some trend.

**Residual analysis**

res\_analysis(residuals(PolyDLM\_model\_Rain$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.95719, p-value = 0.2621

Residual Analysis for Polynomial DLM with part:

1. The data points are both below the line at the start and at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (0.2621) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Now let us fit Koyck model.

##### Koyck model with Rainfall

# Koyk DLM  
  
Koyck\_DLM\_Rain = koyckDlm(x = as.vector(x1) , y = as.vector(y))  
summary(Koyck\_DLM\_Rain)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.3665 -0.4155 -0.1142 0.3241 1.6012   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.3207 2.4302 0.132 0.896  
## Y.1 -6.5147 243.8216 -0.027 0.979  
## X.t 2.2101 76.0635 0.029 0.977  
##   
## Residual standard error: 0.7951 on 27 degrees of freedom  
## Multiple R-Squared: -286.5, Adjusted R-squared: -307.8   
## Wald test: 0.01549 on 2 and 27 DF, p-value: 0.9846   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: 0.04267914 2.21011 -6.514689

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

Wald test statistic is 0.01549  
R - squared is -286.5

Adjusted R - squared is -307.8

Degrees of freedom - DF are (2, 27) p - value (0.9846) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Koyck distributed lag model.

No residual analysis is required.

Let us fit ardlDlm model to check whether it fits better or not.

##### Autoregressive distributed lag model Rainfall

for (i in 1:5){  
 for(j in 1:5){  
 model\_4 = ardlDlm(x = as.vector(x1) , y = as.vector(y), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_4$model), "BIC = ", BIC(model\_4$model), "MASE =", MASE(model\_4)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = -105.2619 BIC = -98.25588 MASE = 0.828275   
## p = 1 q = 2 AIC = -103.3681 BIC = -95.16429 MASE = 0.8543791   
## p = 1 q = 3 AIC = -106.9248 BIC = -97.59935 MASE = 0.8322089   
## p = 1 q = 4 AIC = -102.0678 BIC = -91.70114 MASE = 0.8714349   
## p = 1 q = 5 AIC = -95.80256 BIC = -84.47969 MASE = 0.8152025   
## p = 2 q = 1 AIC = -99.21505 BIC = -91.01127 MASE = 0.9189202   
## p = 2 q = 2 AIC = -101.4552 BIC = -91.88416 MASE = 0.8562257   
## p = 2 q = 3 AIC = -104.9284 BIC = -94.27076 MASE = 0.8328582   
## p = 2 q = 4 AIC = -100.0892 BIC = -88.4267 MASE = 0.8706236   
## p = 2 q = 5 AIC = -93.80342 BIC = -81.22246 MASE = 0.81573   
## p = 3 q = 1 AIC = -102.0287 BIC = -92.70325 MASE = 0.8852654   
## p = 3 q = 2 AIC = -106.4754 BIC = -95.8178 MASE = 0.8316226   
## p = 3 q = 3 AIC = -105.1996 BIC = -93.2098 MASE = 0.8307901   
## p = 3 q = 4 AIC = -99.66585 BIC = -86.70748 MASE = 0.860244   
## p = 3 q = 5 AIC = -93.30294 BIC = -79.46387 MASE = 0.8095313   
## p = 4 q = 1 AIC = -96.40802 BIC = -86.04133 MASE = 0.8956382   
## p = 4 q = 2 AIC = -100.4881 BIC = -88.82555 MASE = 0.8337316   
## p = 4 q = 3 AIC = -100.0049 BIC = -87.04657 MASE = 0.7754451   
## p = 4 q = 4 AIC = -98.96532 BIC = -84.71111 MASE = 0.7942758   
## p = 4 q = 5 AIC = -92.62017 BIC = -77.52301 MASE = 0.7390848   
## p = 5 q = 1 AIC = -93.55318 BIC = -82.23031 MASE = 0.7936346   
## p = 5 q = 2 AIC = -94.0473 BIC = -81.46633 MASE = 0.7842158   
## p = 5 q = 3 AIC = -93.91526 BIC = -80.0762 MASE = 0.7237105   
## p = 5 q = 4 AIC = -92.68035 BIC = -77.58319 MASE = 0.7490465   
## p = 5 q = 5 AIC = -90.68282 BIC = -74.32757 MASE = 0.7469974

(p, q) = (5, 3) has the least AIC, BIC and MASE scores.

# ARDLM model  
AR\_DLM\_Rain\_53 = ardlDlm(x = as.vector(x1) , y = as.vector(y), p = 5, q = 3)  
summary(AR\_DLM\_Rain\_53)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data, start = 1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.072284 -0.008546 0.000512 0.019051 0.039909   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.184797 0.130413 1.417 0.176  
## X.t 0.015255 0.019378 0.787 0.443  
## X.1 0.002519 0.019466 0.129 0.899  
## X.2 -0.005928 0.020127 -0.295 0.772  
## X.3 -0.022491 0.019505 -1.153 0.266  
## X.4 -0.021936 0.020202 -1.086 0.294  
## X.5 -0.001398 0.019784 -0.071 0.945  
## Y.1 0.319059 0.246953 1.292 0.215  
## Y.2 0.275252 0.244529 1.126 0.277  
## Y.3 0.255317 0.233870 1.092 0.291  
##   
## Residual standard error: 0.0332 on 16 degrees of freedom  
## Multiple R-squared: 0.592, Adjusted R-squared: 0.3625   
## F-statistic: 2.58 on 9 and 16 DF, p-value: 0.04715

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model.**

**HA: The data fits the Autoregressive distributed lag model.**

**Interpretations:**

F - statistic is 2.58  
R - squared is 0.592

Adjusted R - squared is 0.3625

Degrees of freedom - DF are (9, 16)

p - value (0.04715) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Autoregressive distributed lag model.

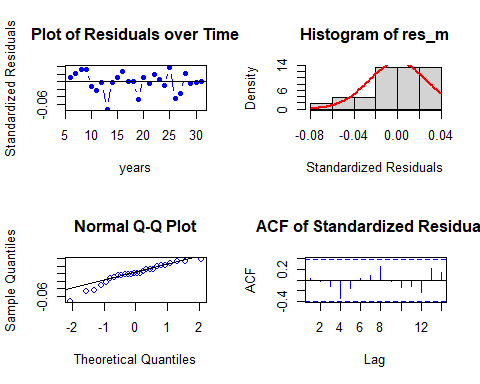
This model suggests that there is only 36.25% of data variance. Suggesting that the model explains only 33.25% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(AR\_DLM\_Rain\_53))

## Time Series:  
## Start = 6   
## End = 31   
## Frequency = 1   
## 6 7 8 9 10   
## 0.0113163308 0.0235060707 0.0335227623 0.0337536460 -0.0134163971   
## 11 12 13 14 15   
## -0.0224225605 -0.0013008064 -0.0722835555 -0.0002717885 0.0132324968   
## 16 17 18 19 20   
## 0.0274869553 0.0024517345 0.0004684645 -0.0462011411 0.0128746231   
## 21 22 23 24 25   
## -0.0040178536 0.0209900331 0.0068564335 -0.0100556290 0.0399094686   
## 26 27 28 29 30   
## -0.0439012239 -0.0306578979 0.0215109321 -0.0036641111 -0.0002427409   
## 31   
## 0.0005557539



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.93614, p-value = 0.1086

Residual Analysis for AR\_DLM\_Rain\_53:

1. The data points are both below the line at the start and at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (0.1086) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Now let us fit with Temperature variable.

##### Finite distributed lag model temperature

for ( i in 1:10){  
 model\_1 = dlm(x = as.vector(x2) , y = as.vector(y), q = i )  
 cat("q = ", i, "AIC = ", AIC(model\_1$model), "BIC = ", BIC(model\_1$model), "MASE =", MASE(model\_1)$MASE, "\n")  
 }

## q = 1 AIC = -101.8617 BIC = -96.2569 MASE = 0.9239038   
## q = 2 AIC = -95.49894 BIC = -88.66246 MASE = 1.032564   
## q = 3 AIC = -96.76727 BIC = -88.77404 MASE = 1.033663   
## q = 4 AIC = -92.75653 BIC = -83.68567 MASE = 1.005373   
## q = 5 AIC = -91.46337 BIC = -81.3986 MASE = 0.8594175   
## q = 6 AIC = -85.74127 BIC = -74.77139 MASE = 0.8103361   
## q = 7 AIC = -82.04015 BIC = -70.25962 MASE = 0.7518958   
## q = 8 AIC = -83.28717 BIC = -70.79674 MASE = 0.6497633   
## q = 9 AIC = -88.10651 BIC = -75.014 MASE = 0.5120906   
## q = 10 AIC = -87.92965 BIC = -74.35085 MASE = 0.458588

As we have the least AIC, BIC and MASE values at q = 10. Let us fit the finite distributed lag model with q = 10.

# Finite lag length based on AIC-BIC-MASE  
  
finite\_dlm\_temp = dlm( x = as.vector(x2) , y = as.vector(y), q = 10)  
summary(finite\_dlm\_temp)

##   
## Call:  
## lm(formula = model.formula, data = design)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.033097 -0.011942 0.005304 0.008460 0.030820   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.521379 0.537055 0.971 0.3570   
## x.t 0.003982 0.016965 0.235 0.8197   
## x.1 0.047512 0.018674 2.544 0.0315 \*  
## x.2 -0.010070 0.019814 -0.508 0.6235   
## x.3 -0.024214 0.020051 -1.208 0.2580   
## x.4 0.011690 0.021762 0.537 0.6042   
## x.5 -0.001764 0.023152 -0.076 0.9409   
## x.6 0.017653 0.019245 0.917 0.3829   
## x.7 0.015177 0.018673 0.813 0.4373   
## x.8 -0.011418 0.020339 -0.561 0.5882   
## x.9 -0.036343 0.019400 -1.873 0.0938 .  
## x.10 0.008222 0.018870 0.436 0.6733   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02453 on 9 degrees of freedom  
## Multiple R-squared: 0.6008, Adjusted R-squared: 0.1129   
## F-statistic: 1.232 on 11 and 9 DF, p-value: 0.3834  
##   
## AIC and BIC values for the model:  
## AIC BIC  
## 1 -87.92965 -74.35085

**Hypotheses:**

**H0: The data doesn′t fit the Finite distributed lag model.**

**HA: The data fits the Finite distributed lag model.**

**Interpretations:**

F - statistic is 1.232  
R - squared is 0.6008

Adjusted R - squared is 0.1129  
Degrees of freedom - DF are (11, 9)

p - value (0.3834) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Finite distributed lag model.

No residual analysis is required.

Therefore, Further analysis is needed by adding polynomial to the lag model.

##### Polynomial distributed lag model Temperature

for (i in 1:3){  
 model\_3 <- polyDlm(x = as.vector(x2), y = as.vector(y), q = i , k = i, show.beta = FALSE)  
 cat("q = ", i, "k = ", i, "AIC = ", AIC(model\_3$model), "BIC = ", BIC(model\_3$model), "MASE = ", MASE(model\_3)$MASE, "\n")  
}

## q = 1 k = 1 AIC = -101.8617 BIC = -96.2569 MASE = 0.9239038   
## q = 2 k = 2 AIC = -95.49894 BIC = -88.66246 MASE = 1.032564   
## q = 3 k = 3 AIC = -96.76727 BIC = -88.77404 MASE = 1.033663

Let us fit a polynomial model of order 3. Since least AIC and BIC scores.

# Ploynomial DLM  
  
PolyDLM\_model\_temp = polyDlm(x = as.vector(x2), y = as.vector(y), q = 1, k = 1, show.beta = TRUE)

## Estimates and t-tests for beta coefficients:  
## Estimate Std. Error t value P(>|t|)  
## beta.0 0.0192 0.0202 0.952 0.3490  
## beta.1 0.0529 0.0216 2.450 0.0205

summary(PolyDLM\_model\_temp)

##   
## Call:  
## "Y ~ (Intercept) + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.075797 -0.024417 -0.002201 0.020077 0.100856   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.05297 0.24450 0.217 0.830  
## z.t0 0.01922 0.02019 0.952 0.350  
## z.t1 0.03364 0.03291 1.022 0.316  
##   
## Residual standard error: 0.04087 on 27 degrees of freedom  
## Multiple R-squared: 0.2404, Adjusted R-squared: 0.1842   
## F-statistic: 4.273 on 2 and 27 DF, p-value: 0.02442

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

F - statistic is 4.273  
R - squared is 0.2404

Adjusted R - squared is 0.1842

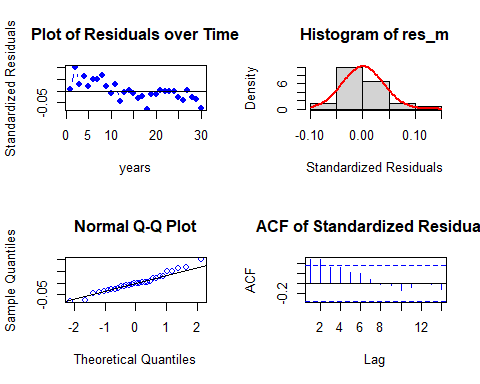
Degrees of freedom - DF are (2, 27)

p - value (0.02442) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Polynomial distributed lag model.

This model suggests that there is only 18.42% of data variance. Suggesting that the model explains only 18.42% of the trend. Which implies that the model shows some trend.

**Residual analysis**

res\_analysis(residuals(PolyDLM\_model\_temp$model))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.9706, p-value = 0.5558

Residual Analysis for Polynomial DLM with part:

1. The data points are both below the line at the start and at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise. Also, ACF shows seasonality pattern.
5. p - value (0.5558) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

fit3 <- ets(v\_RBO\_data\_TS, model="MMN", damped = T)  
fit1 <- forecast.ets(fit3, h = 3)

Now let us fit Koyck model.

##### Koyck model with temperature

# Koyk DLM  
  
Koyck\_DLM\_temp = koyckDlm(x = as.vector(x2) , y = as.vector(y))  
summary(Koyck\_DLM\_temp)

##   
## Call:  
## "Y ~ (Intercept) + Y.1 + X.t"  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.15981 -0.04678 -0.01440 0.04750 0.14952   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -1.2032 1.6184 -0.743 0.464  
## Y.1 0.2469 0.4609 0.536 0.597  
## X.t 0.1847 0.1947 0.949 0.351  
##   
## Residual standard error: 0.07557 on 27 degrees of freedom  
## Multiple R-Squared: -1.597, Adjusted R-squared: -1.789   
## Wald test: 2.119 on 2 and 27 DF, p-value: 0.1397   
##   
## Diagnostic tests:  
## NULL  
##   
## alpha beta phi  
## Geometric coefficients: -1.597614 0.1847273 0.246894

**Hypotheses:**

**H0: The data doesn′t fit the Polynomial distributed lag model.**

**HA: The data fits the Polynomial distributed lag model.**

**Interpretations:**

Wald test statistic is 2.119  
R - squared is -1.597

Adjusted R - squared is -1.789

Degrees of freedom - DF are (2, 27)

p - value (0.1397 ) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the data doesn’t fit the Koyck distributed lag model.

No residual analysis is required.

Let us fit ardlDlm model to check whether it fits better or not.

##### Autoregressive distributed lag model with temperature

for (i in 1:5){  
 for(j in 1:5){  
 model\_4 = ardlDlm(x = as.vector(x2) , y = as.vector(y), p = i , q = j )  
 cat("p = ", i, "q = ", j, "AIC = ", AIC(model\_4$model), "BIC = ", BIC(model\_4$model), "MASE =", MASE(model\_4)$MASE, "\n")  
 }  
}

## p = 1 q = 1 AIC = -107.8419 BIC = -100.8359 MASE = 0.8074316   
## p = 1 q = 2 AIC = -105.4398 BIC = -97.236 MASE = 0.8117033   
## p = 1 q = 3 AIC = -109.6659 BIC = -100.3404 MASE = 0.8057573   
## p = 1 q = 4 AIC = -102.3384 BIC = -91.97173 MASE = 0.8610456   
## p = 1 q = 5 AIC = -97.42825 BIC = -86.10538 MASE = 0.7735245   
## p = 2 q = 1 AIC = -102.1496 BIC = -93.94579 MASE = 0.9024954   
## p = 2 q = 2 AIC = -104.8752 BIC = -95.30414 MASE = 0.7913477   
## p = 2 q = 3 AIC = -109.639 BIC = -98.98132 MASE = 0.790971   
## p = 2 q = 4 AIC = -102.2956 BIC = -90.63304 MASE = 0.8324633   
## p = 2 q = 5 AIC = -96.20616 BIC = -83.62519 MASE = 0.7544973   
## p = 3 q = 1 AIC = -106.2352 BIC = -96.90972 MASE = 0.8876139   
## p = 3 q = 2 AIC = -112.3458 BIC = -101.6882 MASE = 0.7726235   
## p = 3 q = 3 AIC = -111.2734 BIC = -99.28358 MASE = 0.7779966   
## p = 3 q = 4 AIC = -103.7311 BIC = -90.7727 MASE = 0.8253913   
## p = 3 q = 5 AIC = -97.11952 BIC = -83.28046 MASE = 0.7661286   
## p = 4 q = 1 AIC = -102.0941 BIC = -91.72737 MASE = 0.8648001   
## p = 4 q = 2 AIC = -105.9038 BIC = -94.24126 MASE = 0.8119505   
## p = 4 q = 3 AIC = -104.0851 BIC = -91.12676 MASE = 0.8210751   
## p = 4 q = 4 AIC = -102.0982 BIC = -87.844 MASE = 0.8188817   
## p = 4 q = 5 AIC = -95.40963 BIC = -80.31247 MASE = 0.7643666   
## p = 5 q = 1 AIC = -97.85381 BIC = -86.53094 MASE = 0.7652401   
## p = 5 q = 2 AIC = -99.0233 BIC = -86.44233 MASE = 0.7549862   
## p = 5 q = 3 AIC = -97.26093 BIC = -83.42187 MASE = 0.7593949   
## p = 5 q = 4 AIC = -95.45616 BIC = -80.359 MASE = 0.7544104   
## p = 5 q = 5 AIC = -93.50136 BIC = -77.1461 MASE = 0.7617211

(p, q) = (5, 2) has the least AIC, BIC and MASE scores.

# ARDLM model  
AR\_DLM\_temp\_52 = ardlDlm(x = as.vector(x2) , y = as.vector(y), p = 5, q = 2)  
summary(AR\_DLM\_temp\_52)

##   
## Time series regression with "ts" data:  
## Start = 6, End = 31  
##   
## Call:  
## dynlm(formula = as.formula(model.text), data = data, start = 1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.051543 -0.011809 0.002385 0.019521 0.044557   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.093940 0.372713 0.252 0.804  
## X.t 0.013231 0.019672 0.673 0.510  
## X.1 0.035524 0.022038 1.612 0.125  
## X.2 -0.025862 0.023362 -1.107 0.284  
## X.3 -0.027595 0.022784 -1.211 0.242  
## X.4 0.015043 0.019203 0.783 0.444  
## X.5 -0.001256 0.021582 -0.058 0.954  
## Y.1 0.403786 0.239703 1.685 0.110  
## Y.2 0.343301 0.231245 1.485 0.156  
##   
## Residual standard error: 0.03034 on 17 degrees of freedom  
## Multiple R-squared: 0.638, Adjusted R-squared: 0.4676   
## F-statistic: 3.745 on 8 and 17 DF, p-value: 0.01057

**Hypotheses:**

**H0: The data doesn′t fit the Autoregressive distributed lag model.**

**HA: The data fits the Autoregressive distributed lag model.**

**Interpretations:**

F - statistic is 3.745  
R - squared is 0.638

Adjusted R - squared is 0.4676  
Degrees of freedom - DF are (9, 16)

p - value (0.01057) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Autoregressive distributed lag model.

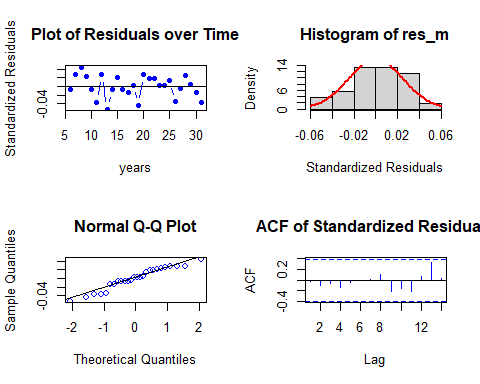
This model suggests that there is only 46.76% of data variance. Suggesting that the model explains only 46.76% of the trend. Which implies that the model shows some trend.

Now let us perform residual analysis.

**Residual analysis**

res\_analysis(residuals(AR\_DLM\_temp\_52))

## Time Series:  
## Start = 6   
## End = 31   
## Frequency = 1   
## 6 7 8 9 10 11   
## -0.005875304 0.027889332 0.044557126 0.023955102 -0.006835884 -0.036247181   
## 12 13 14 15 16 17   
## 0.028047614 -0.051543021 -0.007389583 0.020090374 -0.006168359 -0.014619673   
## 18 19 20 21 22 23   
## 0.002939194 -0.042575018 0.027293790 0.017812015 0.017447685 0.002867739   
## 24 25 26 27 28 29   
## 0.001903187 0.013329216 -0.035009871 -0.004075594 0.025957684 0.005081968   
## 30 31   
## -0.013282240 -0.035550300



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.95433, p-value = 0.2922

Residual Analysis for AR\_DLM\_temp\_52:

1. The data points are both below the line at the start and at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is almost symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are no significant lags in Autocorrelation plot suggesting that the stochastic component is white noise.
5. p - value (0.2922) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

Now let us calculate AIC, BIC and MASE scores and store them in a dataframe to check the better model based on MASE score.

attr(AR\_DLM\_Rain\_53$model, "class") = "lm"  
attr(AR\_DLM\_temp\_52$model, "class") = "lm"  
  
v\_model\_name <- c("PolyDLM\_model\_Rain", "AR\_DLM\_Rain\_53", "PolyDLM\_model\_temp", "AR\_DLM\_temp\_52")

MASE <- MASE(PolyDLM\_model\_Rain$model, AR\_DLM\_Rain\_53$model, PolyDLM\_model\_temp$model, AR\_DLM\_temp\_52$model)$MASE  
  
aic <- AIC(PolyDLM\_model\_Rain$model, AR\_DLM\_Rain\_53$model, PolyDLM\_model\_temp$model, AR\_DLM\_temp\_52$model)$AIC  
  
bic <- BIC(PolyDLM\_model\_Rain$model, AR\_DLM\_Rain\_53$model, PolyDLM\_model\_temp$model, AR\_DLM\_temp\_52$model)$BIC

v\_score <- data.frame(v\_model\_name, MASE, aic, bic)  
colnames(v\_score) <- c("MODEL\_NAME", "MASE", "AIC", "BIC")  
v\_score

## MODEL\_NAME MASE AIC BIC  
## 1 PolyDLM\_model\_Rain 0.9417954 -100.89798 -95.29319  
## 2 AR\_DLM\_Rain\_53 0.7237105 -93.91526 -80.07620  
## 3 PolyDLM\_model\_temp 0.9239038 -101.86168 -96.25690  
## 4 AR\_DLM\_temp\_52 0.7549862 -99.02330 -86.44233

Therefore, AR\_DLM\_Rain\_53 is the better model.

Now let us fit dynamic model.

For dlm model we need some modifications in the series.

v\_data\_TS\_33 <- ts(v\_RBO\_data, start = 1984, frequency = 1)  
colnames(v\_data\_TS\_33) <- c("x1", "y", "x2", "x3", "x4", "x5")

##### Dynamic model with Rainfall

v\_rain\_dyna <- dynlm(y ~ x3, data = data.frame(v\_data\_TS\_33))  
summary(v\_rain\_dyna)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ x3, data = data.frame(v\_data\_TS\_33))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.094310 -0.027945 -0.003845 0.021733 0.102109   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.62656 0.04891 12.810 1.82e-13 \*\*\*  
## x3 0.04696 0.02039 2.303 0.0286 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04171 on 29 degrees of freedom  
## Multiple R-squared: 0.1546, Adjusted R-squared: 0.1255   
## F-statistic: 5.304 on 1 and 29 DF, p-value: 0.02864

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 5.304  
R - squared is 0.1546

Adjusted R - squared is 0.1255  
Degrees of freedom - DF are (1, 29)

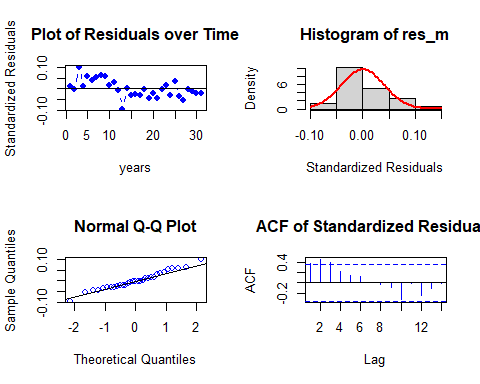
p - value (0.02864) is < 0.05 and therefore, it is statistically significant. Therefore, Null hypothesis is rejected. Hence, the model fits the Dynamic linear model.

This model suggests that there is only 12.55% of data variance. Suggesting that the model explains only 12.55% of the trend. Which implies that the model shows some trend.

Now let us check residuals.

**Residual analysis**

res\_analysis(residuals(v\_rain\_dyna))



##   
## Shapiro-Wilk normality test  
##   
## data: res\_m  
## W = 0.9785, p-value = 0.7697

Residual Analysis for v\_rain\_dyna:

1. The data points are above at the start and below at the end of the trend. Randomness is seen to some extent. So, we cannot decide anything at this stage. Further analysis is required.
2. From normal distribution curve, the distribution is not symmetric.
3. The data at the tails is deviated more leaving some part on the line suggesting there is some normality in the trend.
4. There are significant lags in Autocorrelation plot suggesting that the stochastic component is not white noise.
5. p - value (0.7697) from Shapiro-Wilk normality test is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected.

##### Dynamic model with Temperature

v\_temp\_dyna <- dynlm(y ~ x2, data = data.frame(v\_data\_TS\_33))  
summary(v\_temp\_dyna)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 31  
##   
## Call:  
## dynlm(formula = y ~ x2, data = data.frame(v\_data\_TS\_33))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.071171 -0.029750 -0.009143 0.022486 0.111775   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.44790 0.19931 2.247 0.0324 \*  
## x2 0.03048 0.02094 1.456 0.1561   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04379 on 29 degrees of freedom  
## Multiple R-squared: 0.06812, Adjusted R-squared: 0.03599   
## F-statistic: 2.12 on 1 and 29 DF, p-value: 0.1561

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 2.12  
R - squared is 0.06812

Adjusted R - squared is 0.03599  
Degrees of freedom - DF are (1, 29)

p - value (0.1561) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Dynamic linear model.

No residual analysis is required.

#### Forecasting

For forecasting we need the optimal model among each method. We got only 5 fit models with 2 different predictors.

1. “PolyDLM\_model\_Rain”
2. “AR\_DLM\_Rain\_53”
3. “PolyDLM\_model\_temp”
4. “AR\_DLM\_temp\_52”
5. “v\_rain\_dyna”

Among all the polynomial model with temperature has the least MASE score compared.

Therefore using these models to predict RBO series.

Let us forecast for the next 3 years on RBO series. From 2014 to 2016.

plot(fit1, fcol = "white", main = "First Flowerind Day series with 3 years ahead forecasts", ylab = "Radiation")  
lines(fitted(fit1), col = "darkgreen")  
lines(fit1$mean, col = "darkgreen", lwd = 2)  
legend("topleft", lty = 1, col = c("black", "darkgreen"), c("Data", "Forecasts"))

Chart, line chart

Description automatically generated

Fig 3.21: Next 3 years forecast on the RBO Series.

forecasts <- ts.intersect(ts(fit1$lower[, 2], start = c(2015), frequency = 1), ts(fit1$mean, start = c(2015), frequency = 1), ts(fit1$upper[, 2], start = c(2015), frequency = 1))  
colnames(forecasts) <- c("Lower bound", "Point forecast", "Upper bound")  
forecasts

## Time Series:  
## Start = 2015   
## End = 2017   
## Frequency = 1   
## Lower bound Point forecast Upper bound  
## 2015 0.6528272 0.7233829 0.7953437  
## 2016 0.6443240 0.7226351 0.8007507  
## 2017 0.6391059 0.7220372 0.8104688

From the three year forecast results we can predict that there will be decrease in the Rank-based Order similarity metric in the future. This suggests that the impact of the chemical components will be less in future.

### Part (b)

#### Data

The data here used is the the contemporaneous yearly averaged climate variables measured from 1984 – 2014 (31 years), particularly during the Millennium Drought (1997 – 2009) (13 years).

To get this, the data is trimmed as below,

v\_RBO\_data\_task\_b <- tail(v\_RBO\_data, -13)  
v\_RBO\_data\_task\_b <- head(v\_RBO\_data\_task\_b, -5)  
head(v\_RBO\_data\_task\_b)

## ï..Year RBO Temperature Rainfall Radiation RelHumidity  
## 14 1997 0.6941213 9.038356 1.403014 14.77534 93.74685  
## 15 1998 0.7045545 8.934247 2.289041 14.60000 94.60822  
## 16 1999 0.6992259 9.547945 2.126301 14.61370 96.22603  
## 17 2000 0.7137116 9.680328 2.471858 14.65574 95.65738  
## 18 2001 0.7267423 9.561644 2.227945 14.14521 94.70712  
## 19 2002 0.6629484 9.389041 1.740000 14.63836 93.53233

# Using str() to check the type of each column.  
str(v\_RBO\_data\_task\_b)

## 'data.frame': 13 obs. of 6 variables:  
## $ ï..Year : int 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 ...  
## $ RBO : num 0.694 0.705 0.699 0.714 0.727 ...  
## $ Temperature: num 9.04 8.93 9.55 9.68 9.56 ...  
## $ Rainfall : num 1.4 2.29 2.13 2.47 2.23 ...  
## $ Radiation : num 14.8 14.6 14.6 14.7 14.1 ...  
## $ RelHumidity: num 93.7 94.6 96.2 95.7 94.7 ...

Checking for Missing values.

colSums(is.na(v\_RBO\_data\_task\_b))

## ï..Year RBO Temperature Rainfall Radiation RelHumidity   
## 0 0 0 0 0 0

There are no missing values in the data.

Checking the class of v\_solar\_data. (It should be a data frame.)

class(v\_RBO\_data\_task\_b)

## [1] "data.frame"

v\_RBO\_Temp\_task\_b\_TS <- ts(v\_RBO\_data\_task\_b$Temperature, start = c(1997), frequency = 1)  
v\_RBO\_Rainfall\_task\_b\_TS <- ts(v\_RBO\_data\_task\_b$Rainfall, start = c(1997), frequency = 1)  
v\_RBO\_Radiation\_task\_b\_TS <- ts(v\_RBO\_data\_task\_b$Radiation, start = c(1997), frequency = 1)  
v\_RBO\_RelHumidity\_task\_b\_TS <- ts(v\_RBO\_data\_task\_b$RelHumidity, start = c(1997), frequency = 1)  
v\_RBO\_data\_task\_b\_TS <- ts(v\_RBO\_data\_task\_b$RBO, start = c(1997), frequency = 1)

Confirming the class of each time series object.

class(v\_RBO\_Temp\_task\_b\_TS)

## [1] "ts"

class(v\_RBO\_Rainfall\_task\_b\_TS)

## [1] "ts"

class(v\_RBO\_Radiation\_task\_b\_TS)

## [1] "ts"

class(v\_RBO\_RelHumidity\_task\_b\_TS)

## [1] "ts"

class(v\_RBO\_data\_task\_b\_TS)

## [1] "ts"

Now let us perform descriptive analysis on each time series object.

#### Descriptive Analysis

##### Rank-based Order similarity metric

plot(v\_RBO\_data\_task\_b\_TS, type = "b", xlab = "years", ylab = "Rank-based Order similarity metric", main = "Time series plot for yearly Rank-based Order similarity metric during the Millennium Drought (1997 – 2009) (13 years)", pch = 1)  
legend("topleft", inset = .03, title = "Rank-based Order similarity metric", legend = "Rank-based Order similarity metric series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 4.1: Rank-based Order similarity metric - Time series plot.

McLeod.Li.test(y = v\_RBO\_data\_task\_b\_TS, main = "McLeod-Li Test Statistics for Rank-based Order similarity metric.")

Chart, scatter chart

Description automatically generated

Fig 4.2: McLeod-Li Test Statistics for Rank-based Order similarity metric.

Descriptive analysis

1. From the series plot, we can observe that there is some downward trend in the data.
2. There is an intervention around multiple years.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Temperature

plot(v\_RBO\_Temp\_task\_b\_TS, type = "b", xlab = "years", ylab = "Temperature", main = "Time series plot for yearly temperature during the Millennium Drought (1997 – 2009) (13 years)", pch = 1)  
legend("top", inset = .03, title = "Temperature", legend = "Temperature series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 4.3: Temperature - Time series plot.

McLeod.Li.test(y = v\_RBO\_Temp\_task\_b\_TS, main = "McLeod-Li Test Statistics for Temperature")

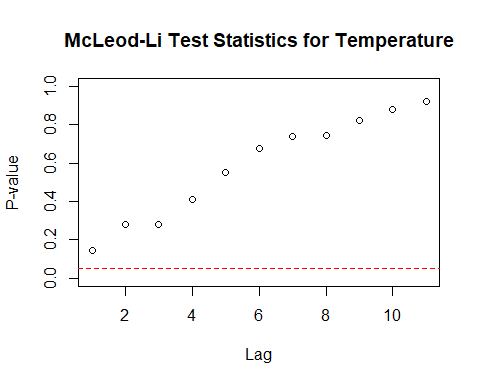


Fig 4.4: McLeod-Li Test Statistics for Temperature

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention at multiple points.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Rainfall

plot(v\_RBO\_Rainfall\_task\_b\_TS, type = "b", xlab = "years", ylab = "Rainfall", main = "Time series plot for yearly Rainfall during the Millennium Drought (1997 – 2009) (13 years)", pch = 1)  
legend("bottom", inset = .03, title = "Rainfall", legend = "Rainfall series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 4.5: Rainfall - Time series plot.

McLeod.Li.test(y = v\_RBO\_Rainfall\_task\_b\_TS, main = "McLeod-Li Test Statistics for Rainfall")

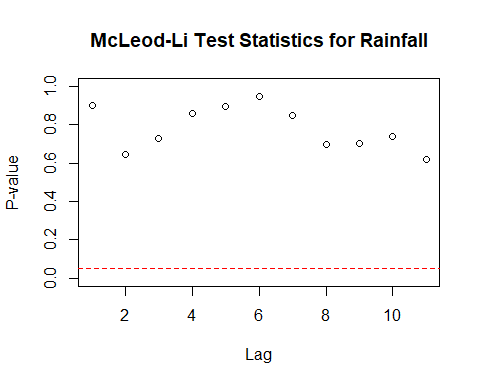


Fig 4.6: McLeod-Li Test Statistics for Rainfall

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 2004.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Radiation

plot(v\_RBO\_Radiation\_task\_b\_TS, type = "b", xlab = "years", ylab = "Radiation", main = "Time series plot for yearly Radiation during the Millennium Drought (1997 – 2009) (13 years)", pch = 1)  
legend("topleft", inset = .03, title = "Radiation", legend = "Radiation series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 4.7: Radiation - Time series plot.

McLeod.Li.test(y = v\_RBO\_Rainfall\_task\_b\_TS, main = "McLeod-Li Test Statistics for Radiation")

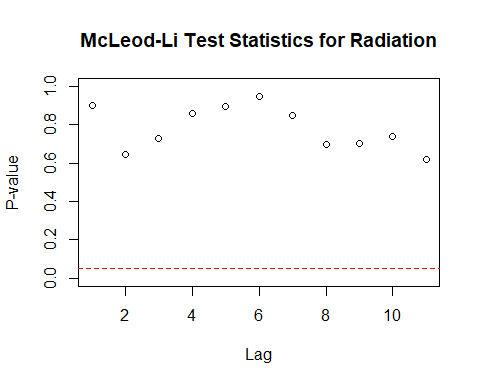


Fig 4.8: McLeod-Li Test Statistics for Radiation

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1992.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

##### Relative Humidity

plot(v\_RBO\_RelHumidity\_task\_b\_TS, type = "b", xlab = "years", ylab = "Relative Humidity", main = "Time series plot for yearly Relative Humidity during the Millennium Drought (1997 – 2009) (13 years)", pch = 1)  
legend("topright", inset = .03, title = "Relative Humidity", legend = "Relative Humidity series", horiz = TRUE, cex = 0.8, lty = 1, box.lty = 2, box.lwd = 2, pch = 1)

Chart, line chart

Description automatically generated

Fig 4.9: Relative Humidity - Time series plot.

McLeod.Li.test(y = v\_RBO\_RelHumidity\_TS, main = "McLeod-Li Test Statistics for Relative Humidity")

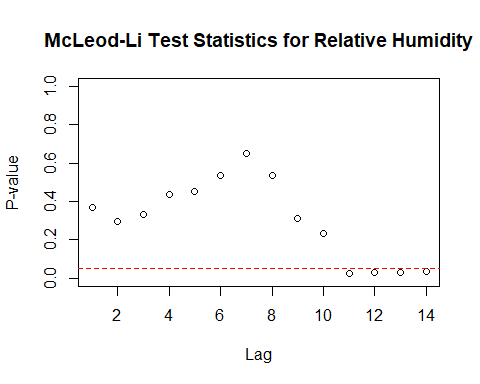


Fig 4.10: McLeod-Li Test Statistics for Relative Humidity.

Descriptive analysis

1. From the series plot, we can observe that there is no trend in the data.
2. There is an intervention around the year 1998.
3. From the series plot, we can conclude that there is no seasonality in the series.
4. There is no consistency across the observed period of time due to higher and lower values.
5. The series shows Autoregressive and moving average behaviour.
6. Also, we can see change in variance.

#### Checking for Stationary in the series

Checking for Stationary on Rank-based Order similarity metric series.

Stationary\_Check(v\_RBO\_data\_task\_b\_TS, "Rank-based Order similarity metric - ACF plot", "Rank-based Order similarity metric - PACF plot")

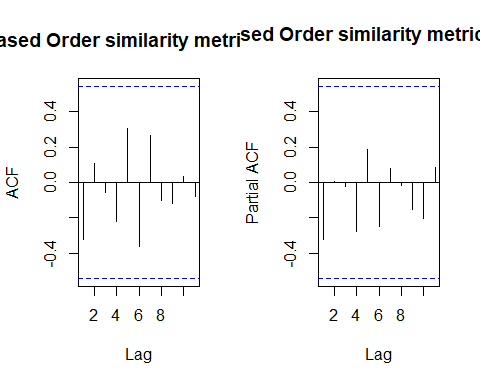


Fig 4.11: Rank-based Order similarity metric – ACF

Fig 4.12: Rank-based Order similarity metric - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.7218, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value : 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Rank - based Order similarity metric series is Stationary.

Checking for Stationary on Temperature data.

Stationary\_Check(v\_RBO\_Temp\_task\_b\_TS, "Temperature - ACF plot", "Temperature - PACF plot")

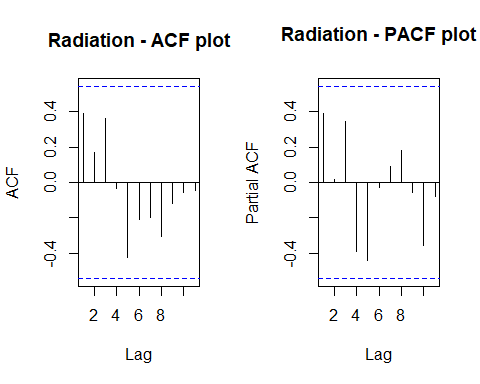


Fig 4.13: Temperature - ACF

Fig 4.14: Temperature - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5475, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Temperature series is Stationary.

Checking for Stationary on Radiation data.

Stationary\_Check(v\_RBO\_Radiation\_task\_b\_TS, "Radiation - ACF plot", "Radiation - PACF plot")

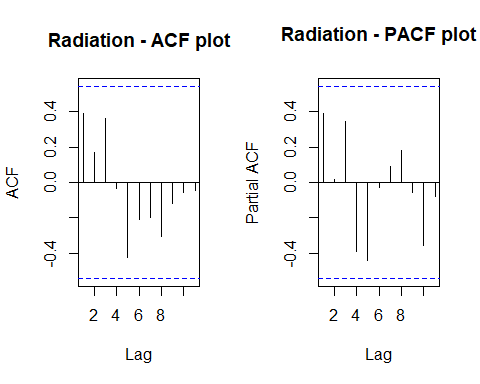


Fig 4.15: Radiation - ACF

Fig 4.16: Radiation - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -2.9296, Lag order = 1, p-value = 0.2182  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.2182 > 0.05

p - value is greater than 0.05 and hence the test is not statistically significant. Therefore, Null hypothesis cannot be rejected i.e., The data is not stationary.

Therefore, the Radiation series is not Stationary.

Checking for Stationary on Rainfall data.

Stationary\_Check(v\_RBO\_Rainfall\_task\_b\_TS, "Rainfall - ACF plot", "Rainfall - PACF plot")

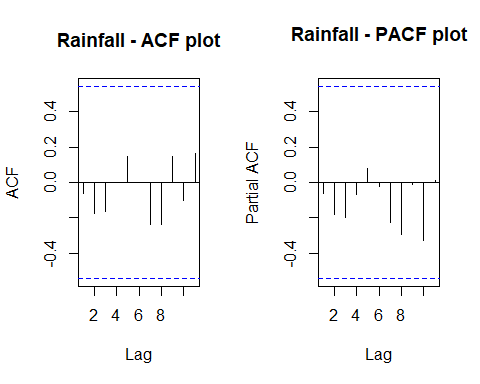


Fig 4.17: Rainfall - ACF

Fig 4.18: Rainfall - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.2994, Lag order = 0, p-value = 0.01281  
## alternative hypothesis: stationary

The are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: 0.01281 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Rainfall series is Stationary.

Checking for Stationary on Relative Humidity data.

Stationary\_Check(v\_RBO\_RelHumidity\_TS, "Relative Humidity - ACF plot", "Relative Humidity - PACF plot")

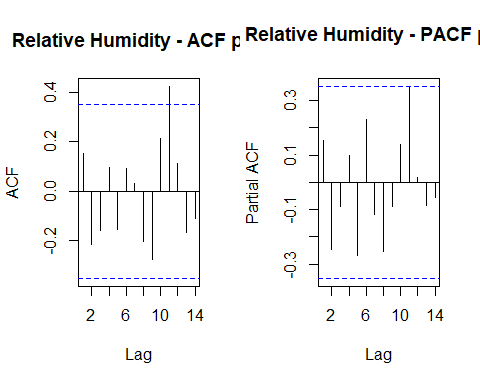


Fig 4.19: Relative Humidity - ACF

Fig 4.20: Relative Humidity - PACF

##   
## Augmented Dickey-Fuller Test  
##   
## data: x  
## Dickey-Fuller = -4.5749, Lag order = 0, p-value = 0.01  
## alternative hypothesis: stationary

There are no significant lags in the ACF and PACF plot suggesting the stochastic component is white noise.

**Hypotheses:**

**H0: The data is not stationary.**

**HA: The data is stationary.**

**Interpretations:**

p - value: ~ 0.01 < 0.05

p - value is less than 0.05 and hence the test is statistically significant. Therefore, we Null hypothesis can be rejected i.e., The data is stationary.

Therefore, the Relative Humidity series is Stationary.

#### Suitable distributed lag model.

Before this let us find the correlation between the series.

# Calculating the correlation coefficient  
cor(v\_RBO\_data\_task\_b\_TS, v\_RBO\_Temp\_task\_b\_TS)

## [1] -0.1684678

cor(v\_RBO\_data\_task\_b\_TS, v\_RBO\_Rainfall\_task\_b\_TS)

## [1] 0.1669995

cor(v\_RBO\_data\_task\_b\_TS, v\_RBO\_Radiation\_task\_b\_TS)

## [1] 0.07287555

cor(v\_RBO\_data\_task\_b\_TS, v\_RBO\_RelHumidity\_task\_b\_TS)

## [1] 0.2371467

This suggests that RBO has a better correlation with Relative humidity and Rainfall.

As we are going to forecast the RBO data, our dependent variable “y” will be Mortality Rate series object and independent variable “x” will be Relative humidity and Rainfall.

For dlm model we need some modifications in the series.

v\_data\_TS\_44 <- ts(v\_RBO\_data\_task\_b, start = 1997, frequency = 1)  
colnames(v\_data\_TS\_44) <- c("x1", "y", "x2", "x3", "x4", "x5")

##### Dynamic model with Relative humidity

v\_rain\_dyna\_taskb <- dynlm(y ~ x5, data = data.frame(v\_data\_TS\_44))  
summary(v\_rain\_dyna\_taskb)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 13  
##   
## Call:  
## dynlm(formula = y ~ x5, data = data.frame(v\_data\_TS\_44))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.037403 -0.007743 -0.003386 0.016193 0.038078   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 0.040585 0.824531 0.049 0.962  
## x5 0.007054 0.008713 0.810 0.435  
##   
## Residual standard error: 0.02177 on 11 degrees of freedom  
## Multiple R-squared: 0.05624, Adjusted R-squared: -0.02956   
## F-statistic: 0.6555 on 1 and 11 DF, p-value: 0.4353

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 0.6555  
R - squared is 0.05624

Adjusted R - squared is -0.02956  
Degrees of freedom - DF are (1, 11)

p - value (0.4353) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Dynamic linear model.

No residual analysis is required.

##### Dynamic model with Rainfall

v\_rel\_hum\_dyna\_taskb <- dynlm(y ~ x3, data = data.frame(v\_data\_TS\_44))  
summary(v\_rel\_hum\_dyna\_taskb)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 13  
##   
## Call:  
## dynlm(formula = y ~ x3, data = data.frame(v\_data\_TS\_44))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.041465 -0.009269 -0.005660 0.017174 0.040068   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.68603 0.03981 17.234 2.62e-09 \*\*\*  
## x3 0.01057 0.01881 0.562 0.586   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.0221 on 11 degrees of freedom  
## Multiple R-squared: 0.02789, Adjusted R-squared: -0.06048   
## F-statistic: 0.3156 on 1 and 11 DF, p-value: 0.5855

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 0.3156  
R - squared is 0.02789

Adjusted R - squared is -0.06048  
Degrees of freedom - DF are (1, 11)

p - value (0.5855) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Dynamic linear model.

No residual analysis is required.

Since the model didn’t fit on the two variables let us fit the remaining 2 variables also.

##### Dynamic model with Radiation

v\_rad\_dyna\_taskb <- dynlm(y ~ x4, data = data.frame(v\_data\_TS\_44))  
summary(v\_rad\_dyna\_taskb)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 13  
##   
## Call:  
## dynlm(formula = y ~ x4, data = data.frame(v\_data\_TS\_44))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.044285 -0.009024 -0.002502 0.015010 0.036522   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.639791 0.282039 2.268 0.0444 \*  
## x4 0.004607 0.019011 0.242 0.8130   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02235 on 11 degrees of freedom  
## Multiple R-squared: 0.005311, Adjusted R-squared: -0.08512   
## F-statistic: 0.05873 on 1 and 11 DF, p-value: 0.813

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 0.05873  
R - squared is 0.005311

Adjusted R - squared is -0.08512  
Degrees of freedom - DF are (1, 11)

p - value (0.813) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Dynamic linear model.

No residual analysis is required.

##### Dynamic model with Temperature

v\_temp\_dyna\_taskb <- dynlm(y ~ x2, data = data.frame(v\_data\_TS\_44))  
summary(v\_temp\_dyna\_taskb)

##   
## Time series regression with "numeric" data:  
## Start = 1, End = 13  
##   
## Call:  
## dynlm(formula = y ~ x2, data = data.frame(v\_data\_TS\_44))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.044526 -0.007965 -0.002129 0.010852 0.033104   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.81163 0.18269 4.443 0.000991 \*\*\*  
## x2 -0.01109 0.01957 -0.567 0.582200   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02209 on 11 degrees of freedom  
## Multiple R-squared: 0.02838, Adjusted R-squared: -0.05995   
## F-statistic: 0.3213 on 1 and 11 DF, p-value: 0.5822

**Hypotheses:**

**H0: The data doesn′t fit the Dynamic linear model.**

**HA: The data fits the Dynamic linear model.**

**Interpretations:**

F - statistic is 0.3156  
R - squared is 0.02789

Adjusted R - squared is -0.06048  
Degrees of freedom - DF are (1, 11)

p - value (0.5855) is > 0.05 and therefore, it is not statistically significant. Therefore, Null hypothesis is not rejected. Hence, the model does not fit the Dynamic linear model.

No residual analysis is required.

To forecast there is predictor variable that is fitted on the dynamic model.

# Conclusion

## Task 1:

The data columns are converted into the time series objects for each column in the data set. They all have seasonality in their series with no obvious trend, behavior or change in variance and with intervention points.

Mortality rate is stationary and hence it can be directly send to the models as it is a dependent variable. Similarly, all other variables becomes the independent variables or predictors. All the variables are stationary.

Decomposition of components suggested that, the seasonality component is week and the trend component doesn’t show any trend with the data.

We got Mortality rate has a strong correlation with Chemical emission 1 and Particle size. Therefore, we use only these variables to fit the models.

Now to find the best model we have three approaches.

1. Suitable Dlag models and dynamic LM: Since, we are doing multivariate analysis it is required to convert the entire data frame into time series. We sent the two correlated variables into the formula. The respective summary and residuals are analysed then.

This approach only supports Finite dlm, AR dlm as well as dynamic model. But for polynomial and Koyck we fit predictors individualls they donot perform multivariate analysis.

Here we got AR\_DLM\_solar\_53 as the better model in terms of residual analysis as well as MASE scores.

Though dynamic model performed better, it suffered with randomness, normality and correlation to some extent.

1. Smoothing methods:

Since the Seasonality component is week we cannot get additive and multiplicative seasonality. The best model here obtained is the simple seasonality model.

1. State space model variations: Here we got ETS(M, N, N) as the best model automatically. it’s mase score is high compared to other models but is efficient in case of BIC and AIC. It suffered with randomness, normality and correlation to some extent.

But the MASE scores are shown least for simple seasonal smoothing method. Also, ets(M, M, N) is better in state space variations method.

So we considered these as the best model for forecasting the Mortality rate data for the next 4 weeks.

From the 4 weeks forecast results of smoothing we can predict that there will be decrease in the Mortality rate in the future. This suggests that the impact of the chemical components will be more in future.

Also, from the 4 weeks forecast results of smoothing we can predict that there will be increase in the Mortality rate in the future. This suggests that the impact of the chemical components will be more in future.

But as we forecast with 95% confidence intervals, we cannot consider this as accurate.

## Task 2:

The data columns are converted into the time series objects for each column in the data set. They all have some obvious trend, behavior or change in variance and with intervention points leaving with no seasonality.

First Flower Day series is stationary and hence it can be directly send to the models as it is a dependent variable. Similarly, all other variables becomes the independent variables or predictors. All the variables are stationary except Temperature and Radiation but it is not necessary to make them stationary.

We got FFD has a strong correlation with Rainfall and Radiation comparitively. Therefore, we use only these variables to fit the models.

Now to find the best model we have three approaches.

1. Suitable Dlag models and dynamic LM: Since, we are doing univariate analysis we will model predictors individually. Here, we also require to analysis the models by fitting them with and without slope. We sent each correlated variables into the formula. The respective summary and residuals are analysed then.

This approach only supports Finite dlm, AR dlm as well as dynamic model.

Here only three models are fitted.

finite dlm with Temperature without slope.  
finite dlm with Radiation without slope.  
Dynamic model without slope with radiation.

Without slope we got best results but they did not perform better with residuals. But dynamic model is better in terms of r squared.

Though dynamic model performed better, it suffered with randomness, normality and correlation to some extent.

1. Smoothing methods:

Since there in no Seasonality in the series cannot use additive and multiplicative seasonality. The best model here obtained is the simple seasonality model.

1. State space model variations: Here we got ETS(M, N, N) as the best model automatically. It’s MASE score is high compared to other models but is efficient in case of BIC and AIC. It suffered with randomness, normality and correlation to some extent.

Among all the methods, holt in exponential smoothing with damped trend. When compared with MASE, BIC and AIC score is better in Smoothing method. Also, ets(M, M, N) is better in state space variations method.

So we considered these as the best model for forecasting the First Flower Day data for the next 4 years. From the 4 years forecast results we can predict that there will be decrease in the First Flower Day in the future. This suggests that the impact of the chemical components will be less in future. But as we forecast with 95% confidence intervals we cannot consider this as accurate.

## Task 3:

### Part (a):

The data columns are converted into the time series objects for each column in the data set. They all have some obvious trend, behavior or change in variance and with intervention points leaving with no seasonality.

Rank-based Order similarity metric series is stationary and hence it can be directly send to the models as it is a dependent variable. Similarly, all other variables becomes the independent variables or predictors. All the variables are stationary except Temperature and Radiation but it is not necessary to make them stationary.

We got Rank-based Order similarity metric has a strong correlation with Rainfall and Temperature comparitively. Therefore, we use only these variables to fit the models.

Now to find the best model we use Suitable Dlag models and dynamic LM:

Since, we are doing univariate analysis we will model predictors individually. The respective summary and residuals are analysed then.

Here, the fitted models among each method with 2 different predictors are

1. “PolyDLM\_model\_Rain”
2. “AR\_DLM\_Rain\_53”
3. “PolyDLM\_model\_temp”
4. “AR\_DLM\_temp\_52”
5. “v\_rain\_dyna”

Among the polynomial models temperature has the least MASE score compared. Whereas, AR\_DLM has Rainfall and Dynamic has only one variable fit i.e., with rainfall.

So we considered these as the best models for forecasting the Rank-based Order similarity metric data for the next 3 years. From the 3 years forecast results we can predict that there will be decrease in the Rank-based Order similarity metric in the future. But as we forecast with 95% confidence intervals we cannot consider this as accurate.

### Part (b):

The RBO data in Task 3 is filtered from 1997 to 2009 considering the millennium drought period.

The data columns are converted into the time series objects for each column in the data set. They all have some obvious trend, behavior or change in variance and with intervention points leaving with no seasonality.

Rank-based Order similarity metric series is stationary and hence it can be directly send to the models as it is a dependent variable. Similarly, all other variables becomes the independent variables or predictors. All the variables are stationary except for Radiation but it is not necessary to make them stationary.

We got Rank-based Order similarity metric has a strong correlation with Relative humidity and Rainfall comparitively. Therefore, we use only these variables to fit the models.

Now to find the best model we use only dynamic LM.

Since, we are doing univariate analysis we will model predictors individually. The respective summary and residuals are analysed then.

The model is not fit on both Relative humidity and Rainfall. Therefore, model is fitted on the other two variables too.

No model is fitted on the data and hence no forecasting is done.

# References

|  |  |
| --- | --- |
| [1] | [Online]. Available: https://cran.r-project.org/web/packages/forecast/index.html. |
| [2] | lmtest, [Online]. Available: https://cran.r-project.org/web/packages/lmtest/index.html. |
| [3] | tseries, [Online]. Available: https://cran.r-project.org/web/packages/tseries/index.html. |
| [4] | fUnitRoots, [Online]. Available: https://cran.r-project.org/web/packages/fUnitRoots/index.html. |
| [5] | expsmooth, [Online]. Available: https://cran.r-project.org/web/packages/expsmooth/index.html. |
| [6] | urca, [Online]. Available: https://cran.r-project.org/web/packages/urca/index.html. |
| [7] | Week 3, 4, 5 and 6 module tasks. |

[8] My previous works for Time series Analysis in my Second year First Sem.