Chapter 4 Experimental Analysis and Results

4.1 ARIMA Model Analysis

The pre-requisite of the ARIMA model is that the forecast variables and predictor variables should be transformed into time series format. This can be achieved in Rstudio by using the in-built function. After transforming the respective variables into time series, the dataset were as follows:

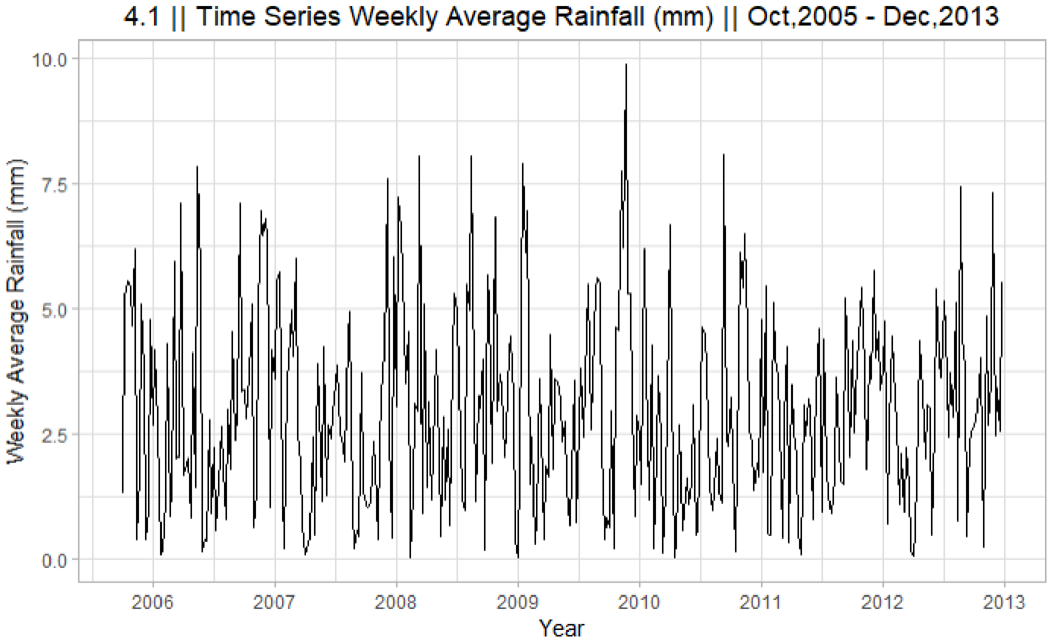


Figure 4.1 Time Series plot of Weekly Average Rainfall data

The above plot is the time series plot of Weekly Average Rainfall in which the x-axis represent the year from 40th week of 2005 to 52th week in 2013 and the y-axis represents the amount of rainfall recorded in ‘mm’ in a particular week.

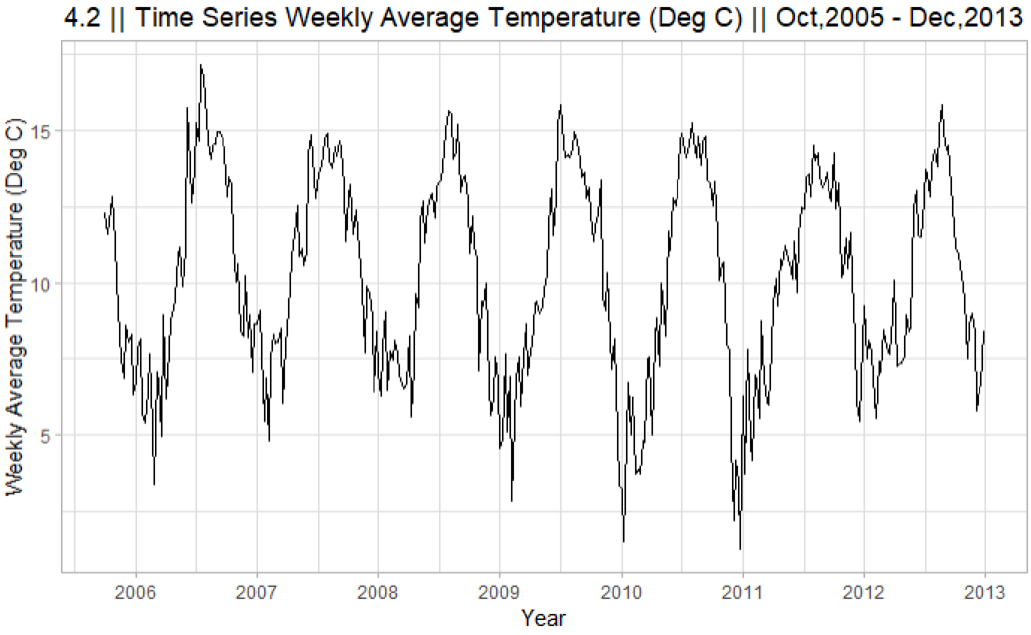


Figure 4.2 Time Series plot of Weekly Average Temperature data

The above plot is the time series plot Weekly Average Temperature data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the amount of temperature recorded in 'Deg C' in a particular week.

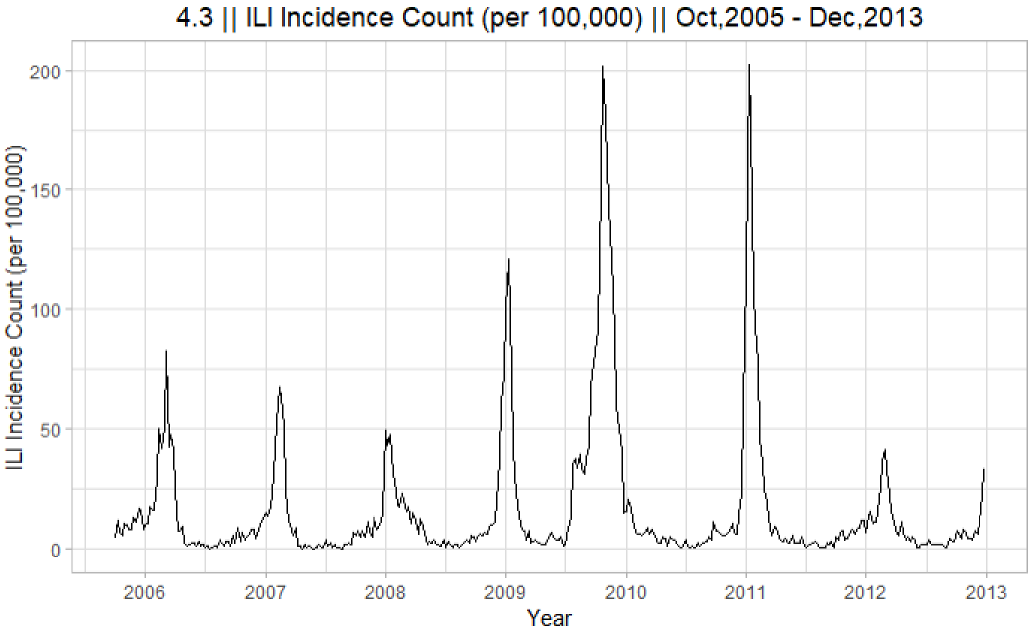


Figure 4.3 Time Series plot of Weekly ILI Incidence count data

The above plot is the time series plot of ILI incidence count data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the number of cases recorded per 100,000 in a particular week.

After plotting time series for all the variables, it was time to check for volatility or outliners by visually examining the time series plots. In the case of Influenza ILI counts plots (Figure 4.3), the Influenza count was below 25 (per 100,000) after week 26th of 2009, and it crossed 200 (per 100,000) before week 52th of 2009 (figure 4.3). This clearly showed the presence of outliners which could bias the model, resulting in skewed statistical summaries. We can use ‘tsclean()’ function which is a part of forecast package. It identifies and replaces outliners with the help of smoothing and decomposition [42]. The time series after cleaning look like below:

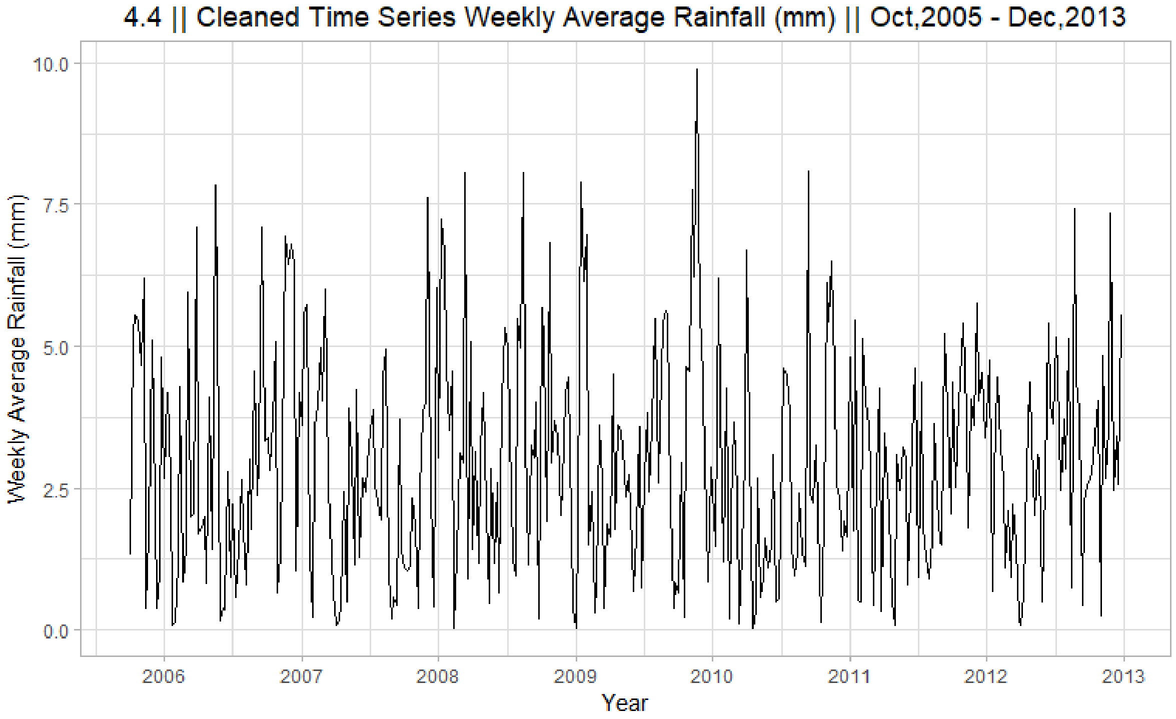


Figure 4.4 Time Series plot of Cleaned Weekly Average Rainfall data

The above plot is the time series plot of cleaned Weekly Average Rainfall in which the x-axis represent the year from 40th week of 2005 to 52th week in 2013 and the y-axis represents the amount of rainfall recorded in ‘mm’ in a particular week.

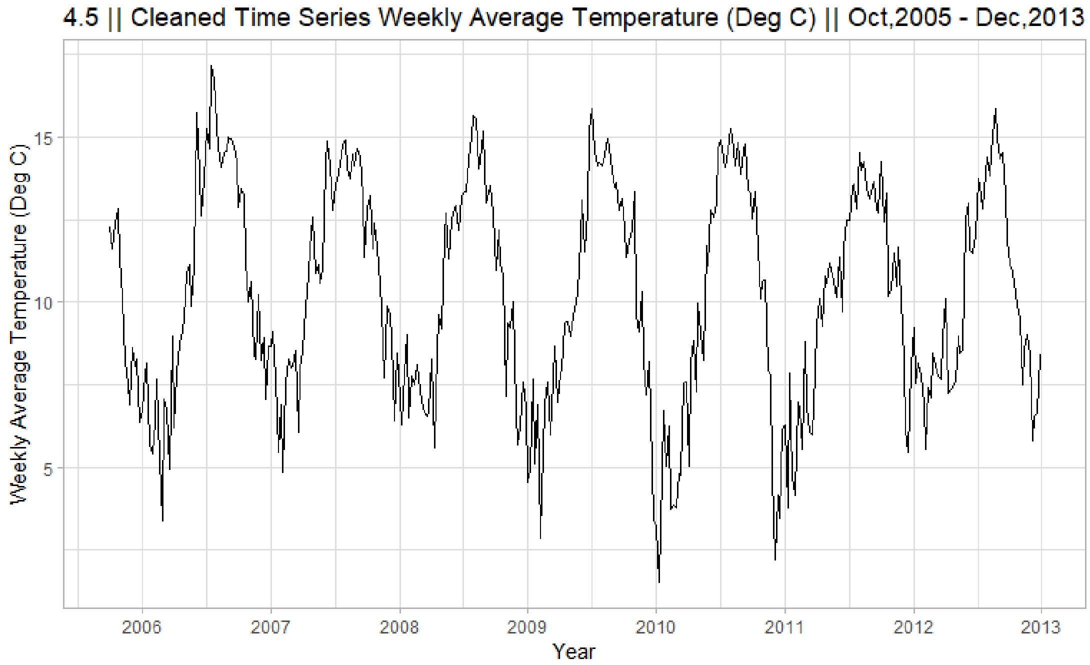


Figure 4.5 Time Series plot of Cleaned Weekly Average Temperature data

The above plot is the time series plot of cleaned Weekly Average Temperature data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the amount of temperature recorded in 'Deg C' in a particular week.

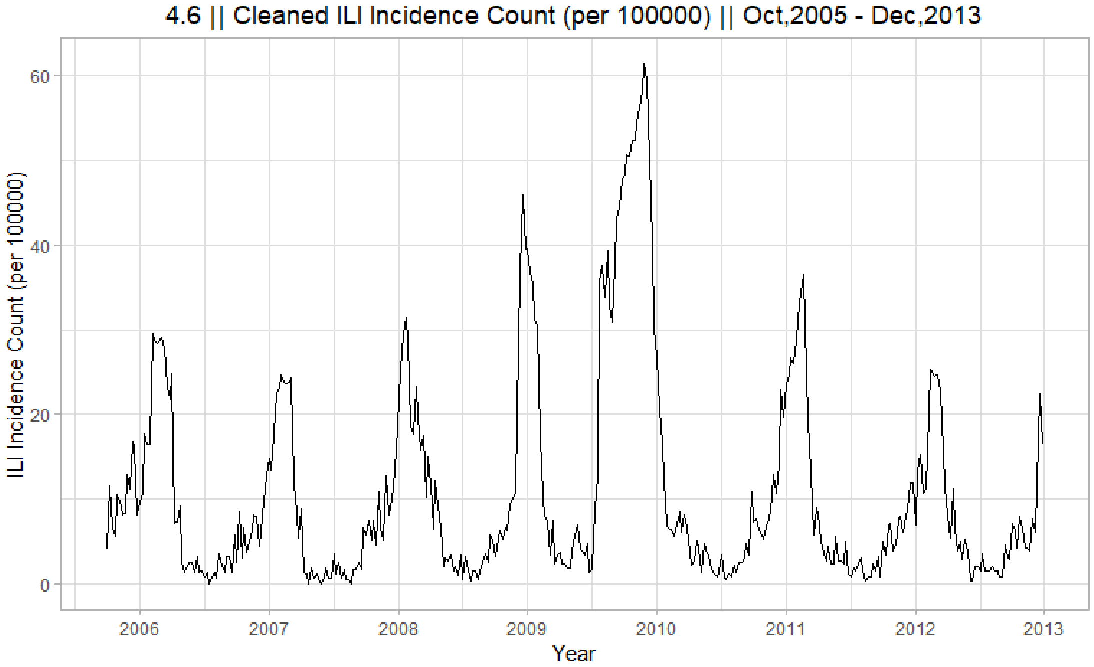


Figure 4.6 Time Series plot of Cleaned Weekly ILI Incidence count data

The above plot is the time series plot of ILI incidence count data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the number of cases recorded per 100,000 in a particular week.

After removal of the outliners, the time series of the above variables is still volatile. The effective way to smooth these series and to remove the noisy fluctuations is by drawing a line through these bigger troughs and peaks. This line represents the simplest concept in time series analysis which is known by the name of moving average. In this concept, the points across several time periods are averaged, therefore the observed time series is smoothen in a predictable and stable time series [42].

In mathematical terms, an order 'm' moving average is calculated by taking an average of series Y, k periods around each point:

$$MA = \frac{1}{m}\sum_{j\,=\,-k}^{k} y_{t\,+\,j}$$

, where m = 2k +1.

The smoothing of the original time series depends on the size of the window selected. If the window of the moving average is wider, then the smoothness of the time series will be more.

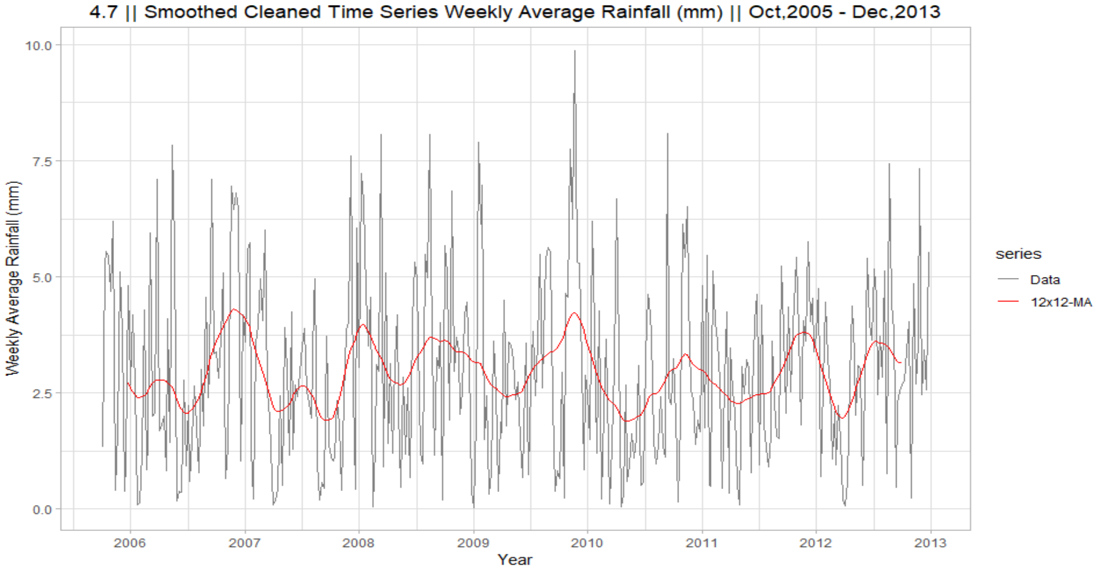


Figure 4.7 Time Series plot of Smoothed Cleaned Weekly Average Rainfall data

The above plot is the time series plot of smoothed cleaned Weekly Average Rainfall in which the x-axis represent the year from 40th week of 2005 to 52th week in 2013 and the y-axis represents the amount of rainfall recorded in 'mm' in a particular week. The 'red' is the smoothed series and 'grey' is the original series.

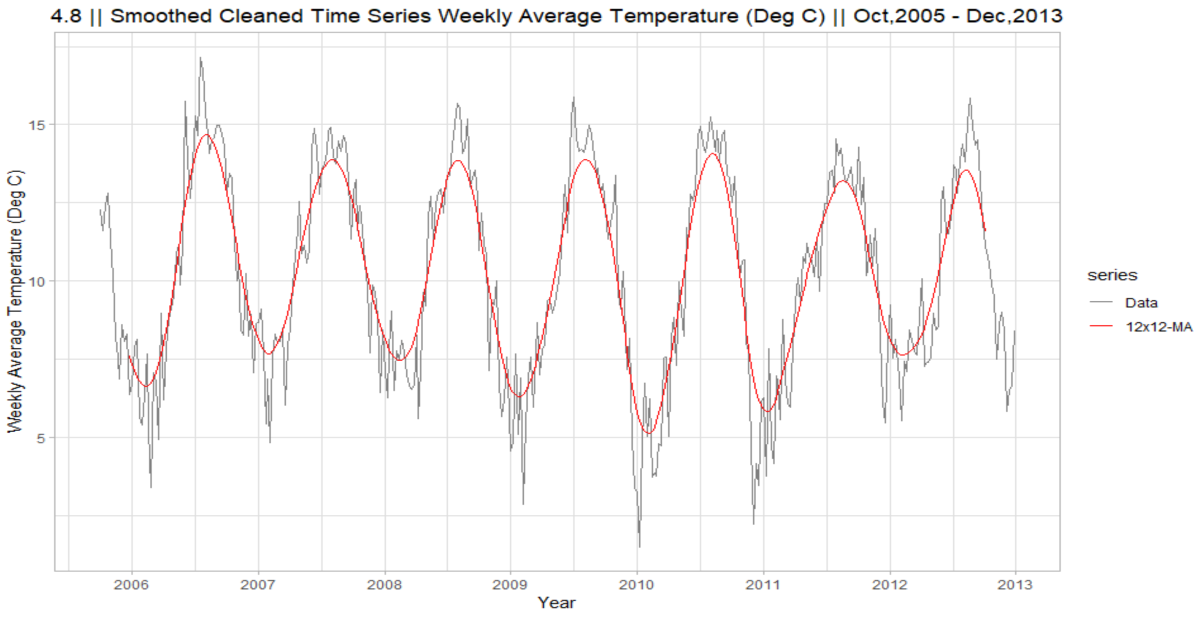


Figure 4.8 Time Series plot of Smoothed Cleaned Weekly Average Temperature data

The above plot is the time series plot of smoothed cleaned Weekly Average Temperature data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the amount of temperature recorded in 'Deg C' in a particular week. The 'red' is the smoothed series and 'grey' is the original series.

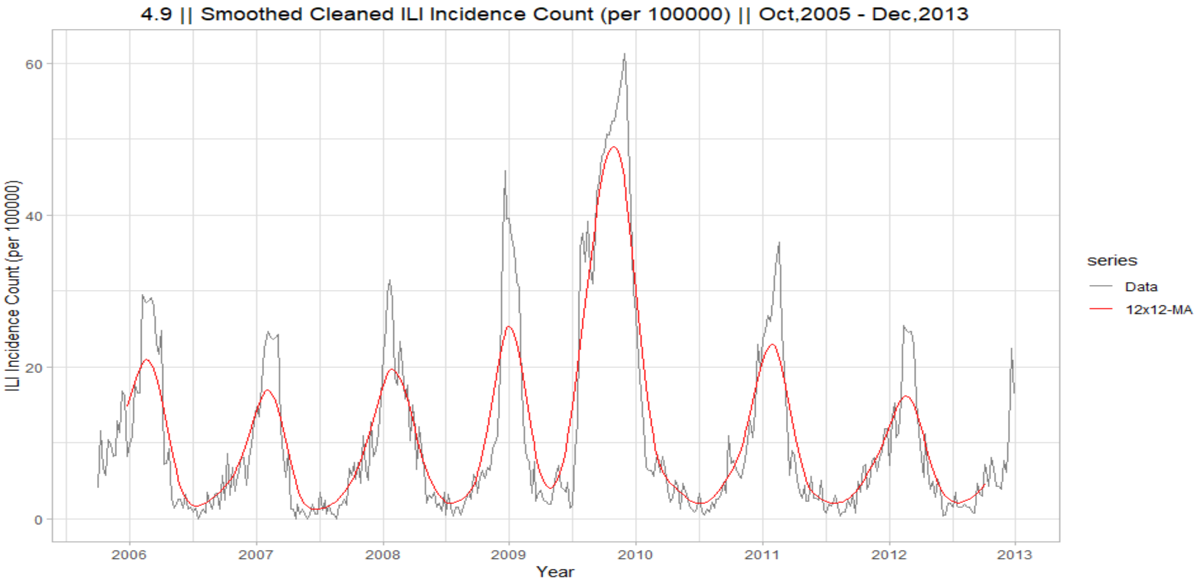


Figure 4.9 Time Series plot of Smoothed Cleaned Weekly ILI Incidence count data

The above plot is the time series plot of ILI incidence count data in which the x-axis represent the year from 40th week of 2005 to 52th week of 2013 and the y-axis represents the number of cases recorded per 100,000 in a particular week. The 'red' is the smoothed series and 'grey' is the original series.

After smoothing, the time series is decomposed further to analyse the seasonality, trend, and cycle. All the three components are understood to capture the historical pattern in the series. By decomposing the series, the behaviour of it can be understood, and it can further help in forming the foundation for building the forecasting model. These three component can be understood as follows:

1. **Seasonal Component**: It refers to the fluctuations in the data according to the cycles with a calendar year. For instance: the number of influenza cases increases during winter months between December to February.
2. **Trend Component**: It refers to the long term increase or decrease in the data. For instance: Are the sales of supermarket increasing or decreasing over time?
3. **Cycle Component**: It refers to the increasing or decreasing patterns that are not fixed with any period. These fluctuation happens due to economic conditions, and are often associated with “business cycle”.
4. **Error**: the part of the time series which cannot be attributed to seasonal, cycle, or trend components.

The process of breaking the time series into these components is known as decomposition. In Rstudio, the time-series can be decomposed with the help of ‘stl()’ function. It calculates the seasonal component of the series using smoothing.

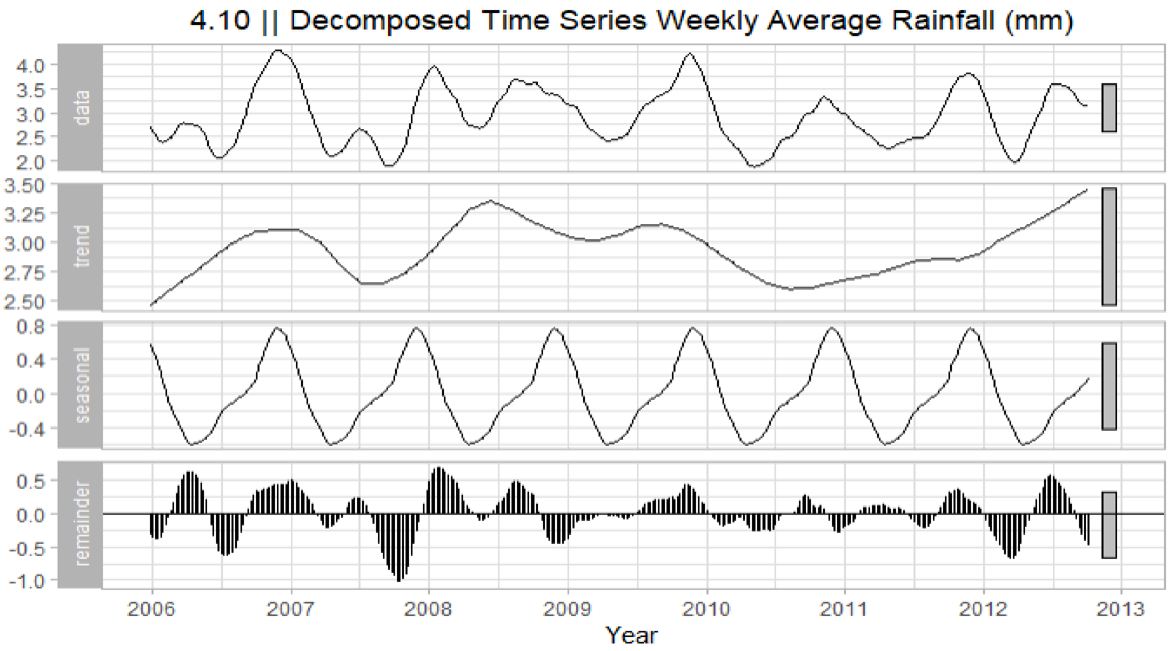


Figure 4.10 Decomposed Time Series plot of Weekly Average Rainfall

The above decomposed plot shows the trend, seasonality and any pattern present in the time series over a given period.

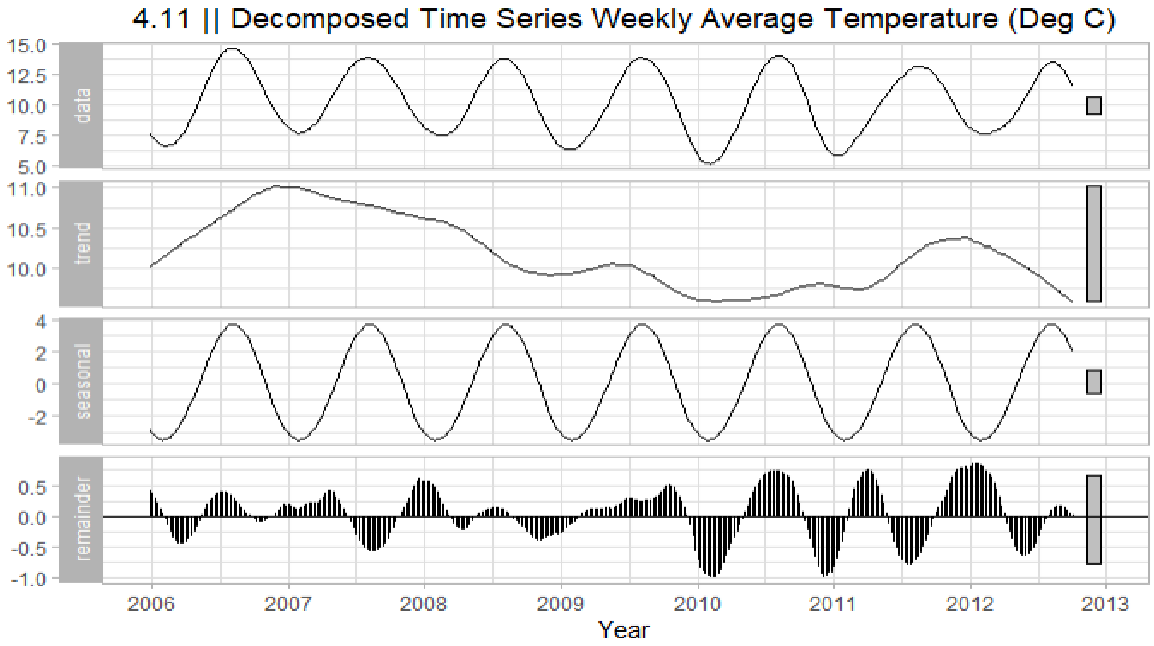


Figure 4.11 Decomposed Time Series plot of Weekly Average Temperature

The above decomposed plot shows the trend, seasonality and any pattern present in the time series over a given period.

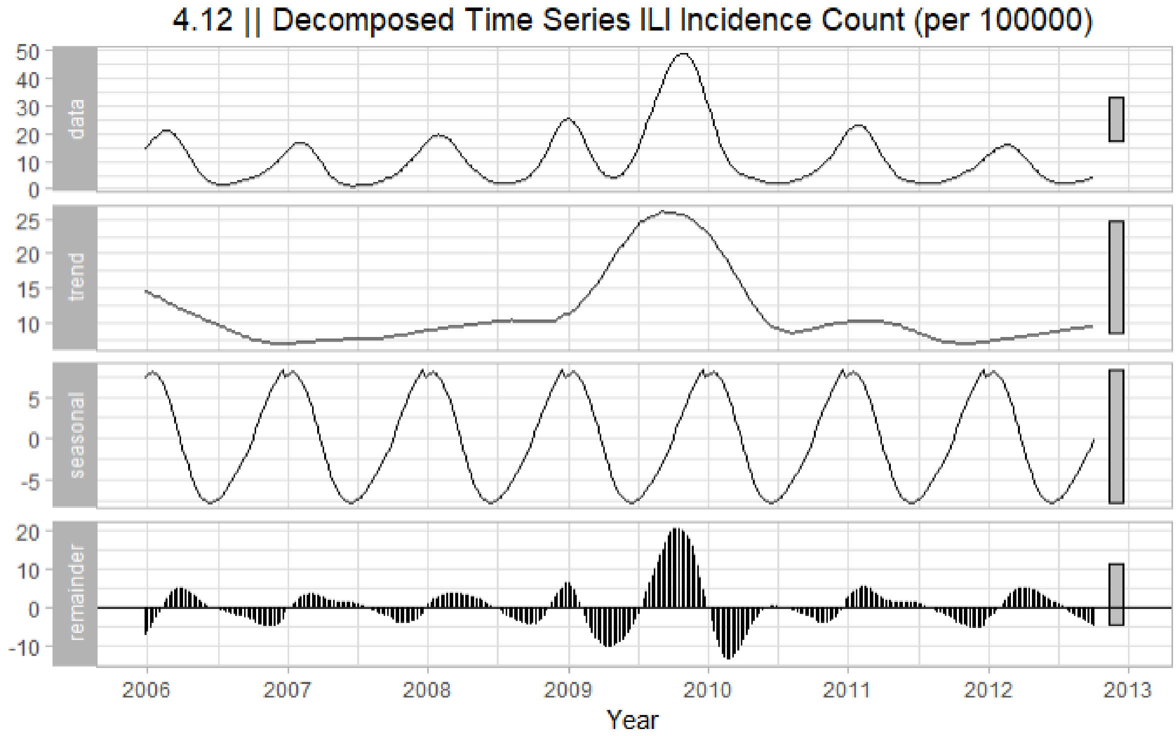


Figure 4.12 Decomposed Time Series plot of Weekly ILI Incidence Count

The above decomposed plot shows the trend, seasonality and any pattern present in the time series over a given period.

After observing the decomposed form of all the three time series (fig. 10, fig. 11 and fig. 12), we can infer that there is a strong seasonality in the rainfall time series (fig. 10) and temperature time series (fig. 11) and not so strong seasonality in the ILI incidence count time series (fig. 12). However, there was no trend or cycle component in all the time series.

Since the time series has to be de-seasoned before fitting in the ARIMA model, we took the log transform of rainfall and temperature time series and applied seasonal differencing to it with a lag of 52 weeks as the data is weekly. In the case of ILI incidence count time series, only seasonal differencing is taken with a lag of 52 weeks. In Rstudio, the seasonal differencing is calculate with ‘diff()’ function of the forecast package.

The deseasonalized time series plot shown below are the time series of Temperature, Rainfall and ILI incidence count with no seasonality component, as per the reason explained of de-seasonalizing in the previous page.

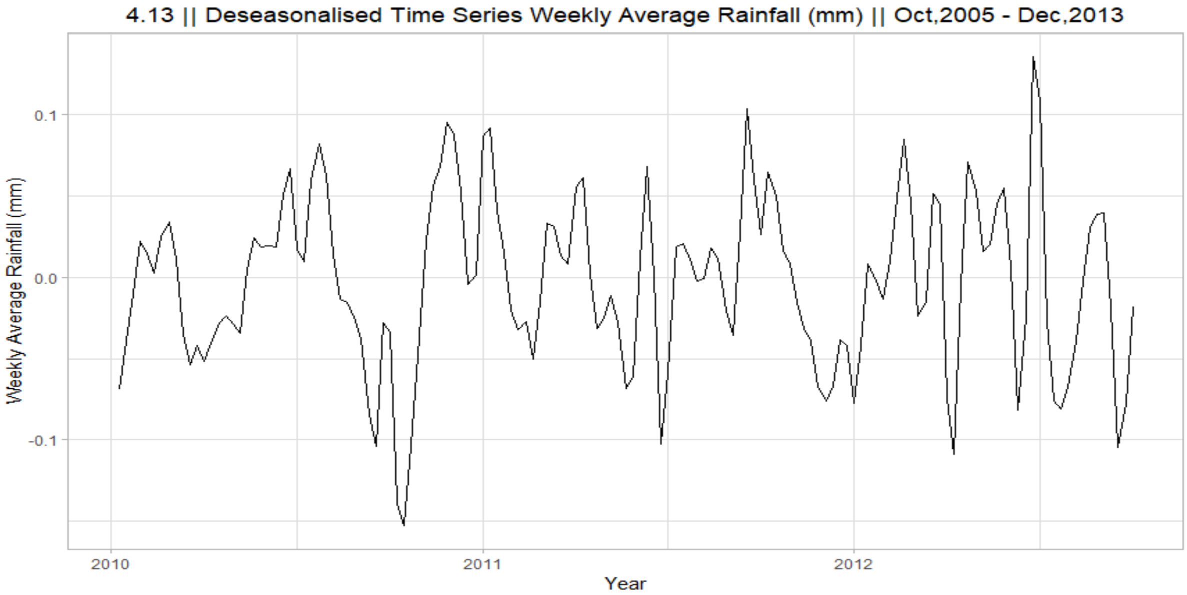


Figure 4.13 Deseasonalised Time Series plot of Weekly Average Rainfall

The time series weekly average rainfall plot over a given time is shown after removing the seasonal component.

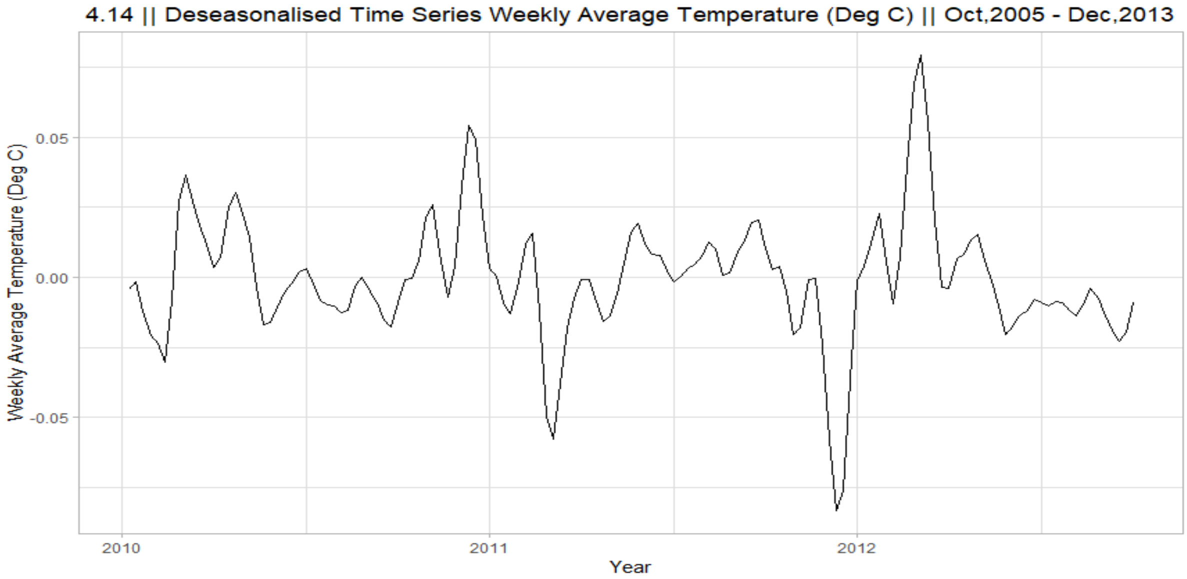


Figure 4.14 Deseasonalised Time Series plot of Weekly Average Temperature

The time series weekly average temperature plot over a given time is shown after removing the seasonal component.

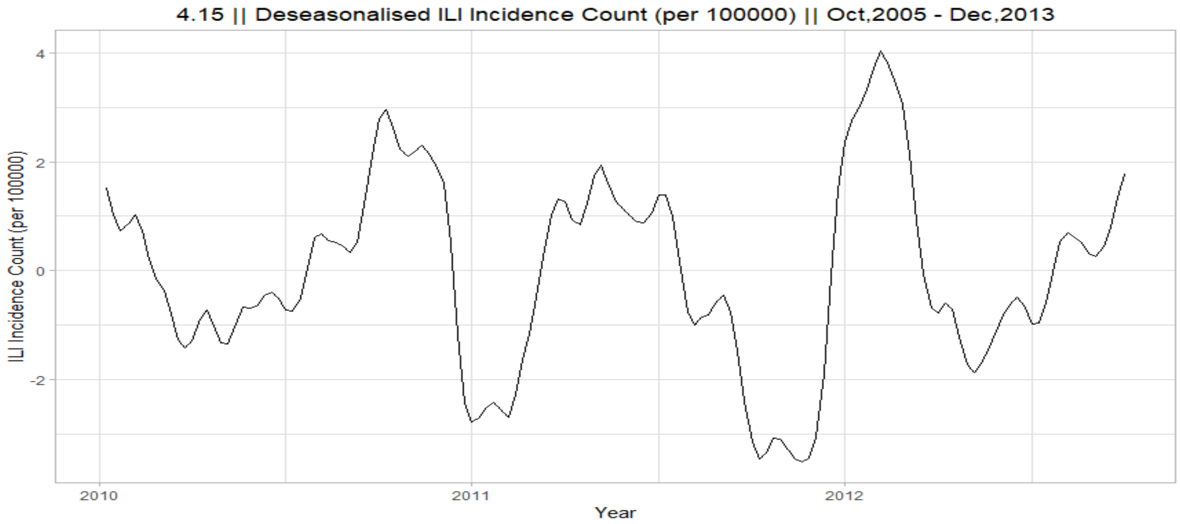


Figure 4.15 Deseasonalised Time Series plot of Weekly ILI Incidence Count

The time series weekly ILI incidence count plot over a given time is shown after removing the seasonal component.

After all the time series is deseasonalized, another requirement of the modelling is that all the time series must be stationarity before fitting into the model. A time series is said to be stationary when its mean, variance, and auto covariance are constant over a period of time. The reason for stationarity can be understood on the reasoning that ARIMA model make use of older lags of the series to model its future behaviour, therefore the modelling of stable time series with constant properties provides less uncertainty.

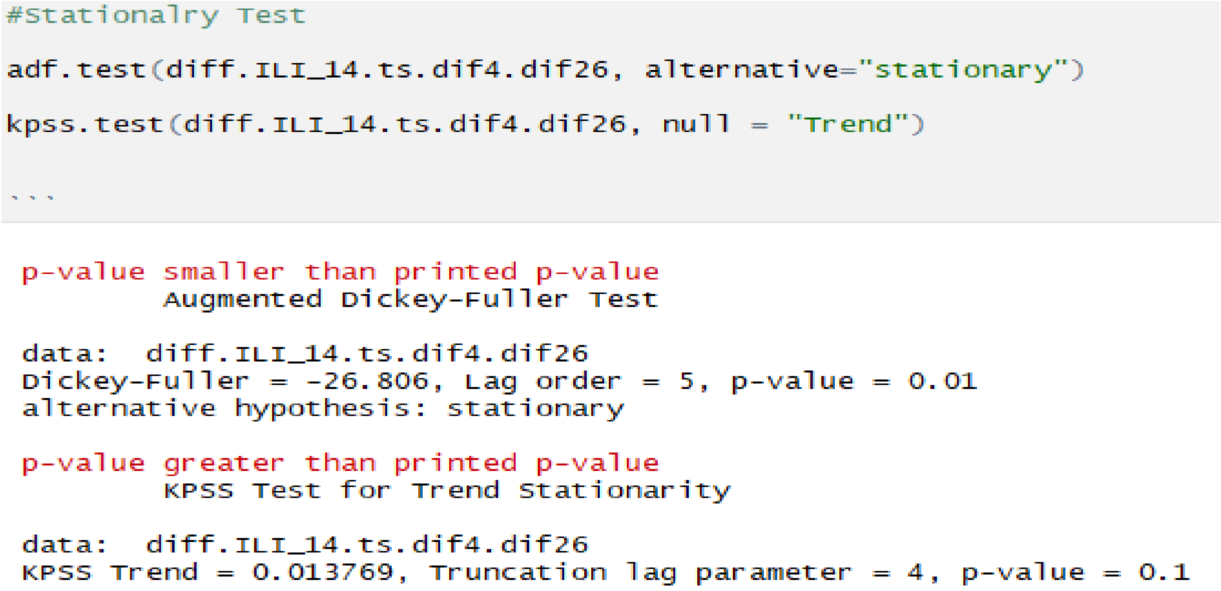
The augmented Dickey-Fuller (ADF) test is one of the statistical test which is used to test stationarity of a time series. The ADF tests whether the lagged value and a linear trend could explain the corresponding change in the value of Y. The null hypothesis states that the time series is non-stationary. If in the change of value of Y is non-significant due to the contribution of lagged value, then the series is non-stationary and the null hypothesis will not be rejected.

The Kwiatkowski–Phillips–Schmidt–Shin (KPSS) is another statistical test which is used to check the stationarity of a time series. It checks if a time series has constant mean or linear trend, or due to a unit root, it is non-stationary. The null hypothesis states that the data is stationary, whereas the alternate hypothesis states that the data is not stationary.

Therefore, the p-value of the ADF test should be less than 0.05, indicating that the null hypotheses is rejected. And the alternate hypothesis is accepted, whereas the p-value of the KPSS test should be greater than 0.05, indicating that the null hypotheses is accepted.

If one or both of the tests are failed, then it would mean that the time series is non-stationary. The non-stationary time series is corrected by differencing. The reason behind differencing is that, if the original time series doesn't have constant mean and variance over time, then the change to another period might have constant properties. If the period also doesn't have stationary time series, then it is further differenced with the same method. This differencing is calculated by subtracting previous year's value from the current year's value.

For all the three time series, I had to apply second order differencing to make the time series stationary after checking the time series multiple times with both the test.



```
#Stationalry Test
adf.test(diff.ILI_14.ts.dif4.dif26, alternative="stationary")
kpss.test(diff.ILI_14.ts.dif4.dif26, null = "Trend")

```
```

```
p-value smaller than printed p-value
        Augmented Dickey-Fuller Test

data:  diff.ILI_14.ts.dif4.dif26
Dickey-Fuller = -26.806, Lag order = 5, p-value = 0.01
alternative hypothesis: stationary

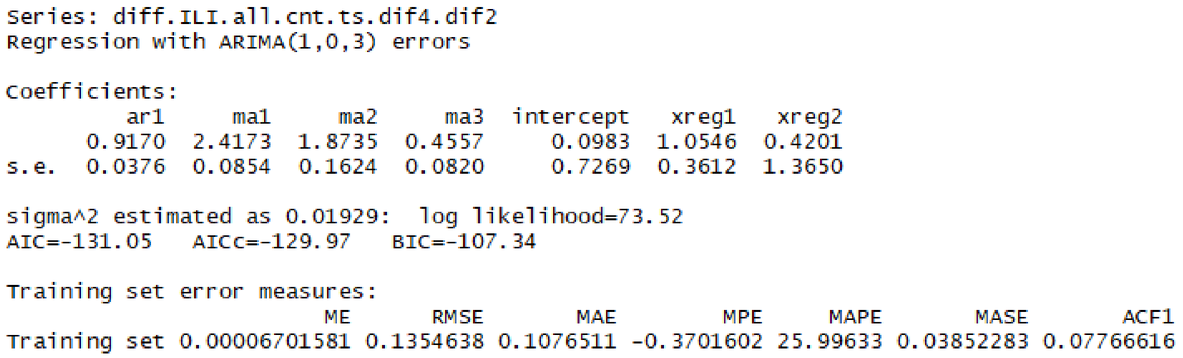
p-value greater than printed p-value
        KPSS Test for Trend Stationarity

data:  diff.ILI_14.ts.dif4.dif26
KPSS Trend = 0.013769, Truncation lag parameter = 4, p-value = 0.1
```

Figure 4.16 Stationarity Test (ADF and KPSS)

The above image shows the results of stationarity test applied to the time series
```

The Auto Correlation function or Autocorrelation plot is also a tool which can help in checking if the series is stationary by visually looking at it. Adding to it, these plots also contribute in selecting the order parameters for ARIMA model. Displaying the correlation between a series and its lags, it also suggests the order of differencing and the order of the MA (q) model. On the other hand, the Partial autocorrelation plots (PACF) helps in suggesting the order of AR (p) model, as well as demonstrating the correlation between a variable and its lags which is not explained by the previous lags.

Since selecting the order for AR (p), MA (q) and differencing (d) could be quite cumbersome, I used the auto.arima() function. The auto.arima() function helps in automatically generating an optimal set of (p,d,q). For this, a lot of criteria are compared to select the optimal parameters for the model fit. Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) are the most widely used parameters. Both of them share an estimate about the amount of information lost, if any particular model is chosen. Therefore, the AIC and BIC values have to be minimised while comparing the models.

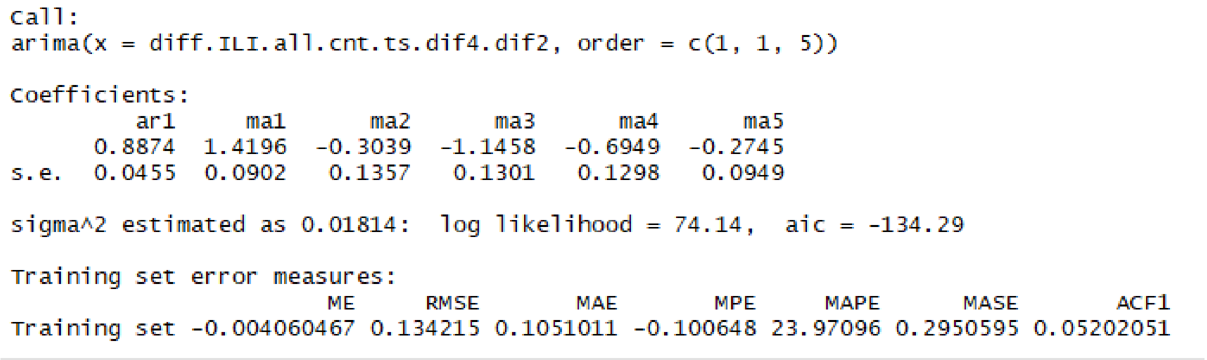
After performing all the above steps, the ARIMA model is fitted to de-seasonalized and smooth data, and the results obtained were as follows:



```
Series: diff.ILI.all.cnt.ts.dif4.dif2
Regression with ARIMA(1,0,3) errors

Coefficients:
         ar1     ma1     ma2     ma3  intercept   xreg1   xreg2
      0.9170  2.4173  1.8735  0.4557     0.0983  1.0546  0.4201
s.e.  0.0376  0.0854  0.1624  0.0820     0.7269  0.3612  1.3650

sigma^2 estimated as 0.01929:  log likelihood=73.52
AIC=-131.05   AICc=-129.97   BIC=-107.34

Training set error measures:
                         ME      RMSE       MAE        MPE     MAPE       MASE       ACF1
Training set 0.00006701581 0.1354638 0.1076511 -0.3701602 25.99633 0.03852283 0.07766616
```

Figure 4.17 ARIMA Model Summary (Multivariate)

The result of fitting the time series data of rainfall, temperature and ILI incidence count in a multivariate ARIMA model using auto.arima() function. The best value of AIC, BIC and RMSE along with AR and MA coefficients of all variables are shown above.



```
Call:
arima(x = diff.ILI.all.cnt.ts.dif4.dif2, order = c(1, 1, 5))

Coefficients:
         ar1     ma1      ma2      ma3      ma4      ma5
      0.8874  1.4196  -0.3039  -1.1458  -0.6949  -0.2745
s.e.  0.0455  0.0902   0.1357   0.1301   0.1298   0.0949

sigma^2 estimated as 0.01814:  log likelihood = 74.14,  aic = -134.29

Training set error measures:
                       ME      RMSE       MAE       MPE     MAPE      MASE       ACF1
Training set -0.004060467 0.134215 0.1051011 -0.100648 23.97096 0.2950595 0.05202051
```

Figure 4.18 ARIMA Model Summary (Univariate)

The result of fitting the time series data of ILI incidence count in a univariate ARIMA model using auto.arima() function, and then using arima() with parameters obtained from auto.arima(). The best value of AIC, BIC and RMSE along with AR and MA coefficients of all variables are shown above.

As per the above results, it can be seen that the best value of (p,d,q) for multivariate ARIMA model is (1,0,3) for the next 52 weeks of the data, and for univariate ARIMA model is (1,1,5) for the next 52 weeks of the data.

After fitting the model, we have to check if the fitted model is appropriate for further forecasting. This is done by examining the ACF and PACF plots of the model residuals for both univariate and multivariate models. The residual plots gives the difference between the observed value and forecasted value. As a thumb rule, if the model order parameters and structure are accurate, then we won’t see any significant auto-correlation present in the residual plots.

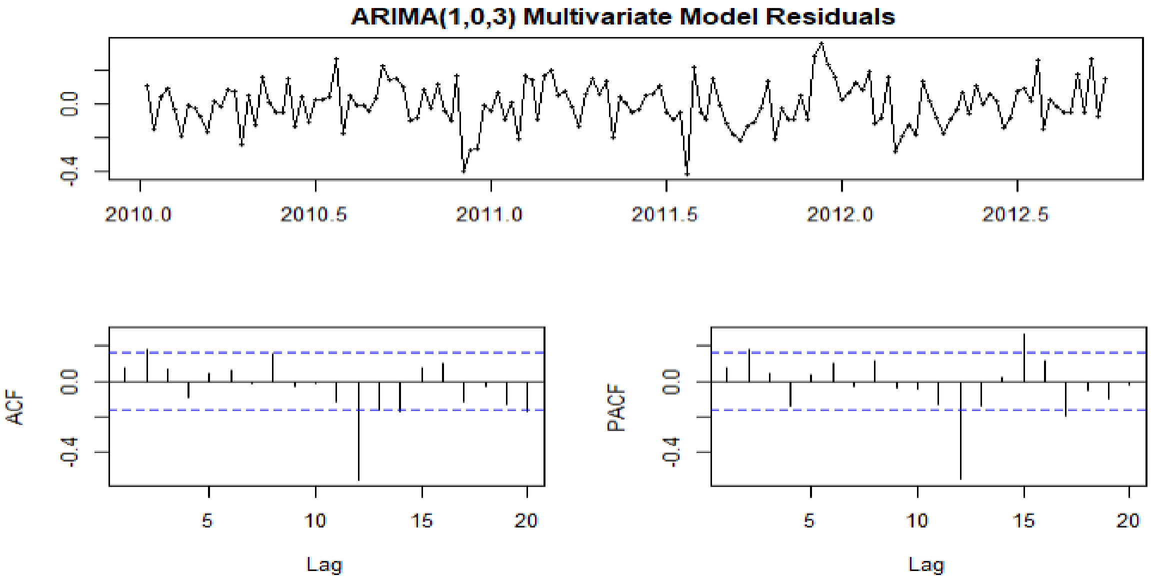


Figure 4.19 ARIMA Residual Plot (Multivariate)

The residual plot above gives the ACF and PACF lags for the residual component of the fitted Multivariate Time Series model

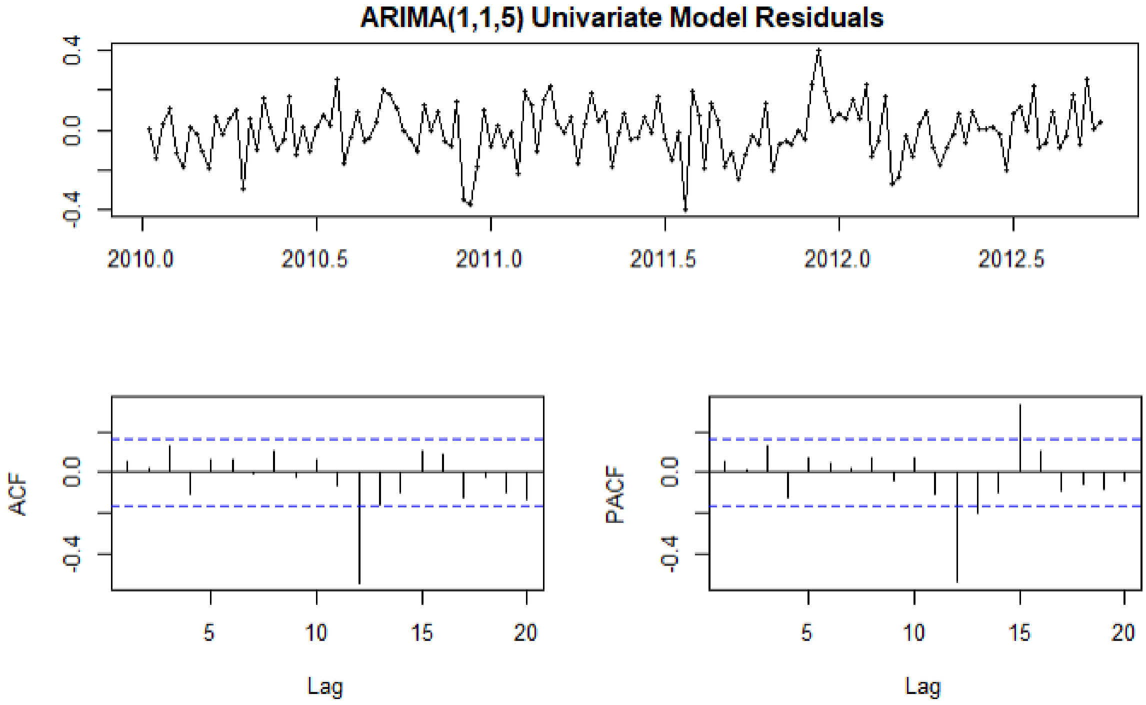
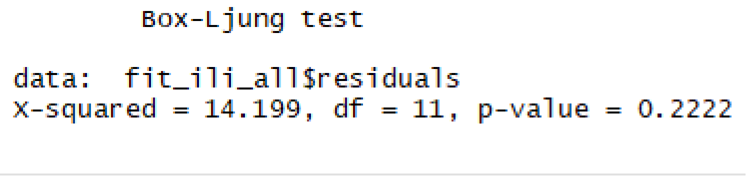


Figure 4.20 ARIMA Residual Plot (Univariate)

The residual plot above gives the ACF and PACF lags for the residual component of the fitted Univariate Time Series model

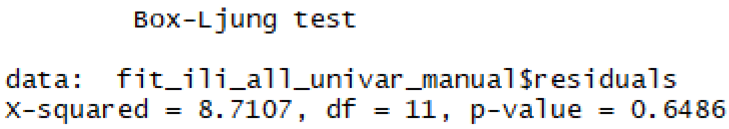
The residuals of the fitted models for both multivariate and univariate models do not have significant auto-correlation present in ACF and PACF plot, i.e., in fig. 4.19 and fig. 4.20, till the first 12 lags, almost all the lags fall inside the 95% confidence bound indicating that the residual appears to be random. Therefore, these models seems to be a better fit for the given time series data. This can be further verified by using statistical Ljung Box test. It is applied on the residuals obtained from both the fitted ARIMA Multivariate and Univariate models. Rather than testing randomness at a particular lag, this test checks the "overall" randomness based upon the number of lags considered. The null hypothesis states that the data is random, whereas the alternate hypothesis states that the data is not random and correlation exists. For the first 11 lag of multivariate model, the autocorrelation among the residuals are zero (p = 0.2222), indicating that the residuals are random, and the model provide adequate fit to the data, whereas for the first 11 lag of univariate model, the autocorrelation among the residuals are zero (p = 0.6486), indicating that the residuals are random, and the model provide adequate fit to the data.



```
                 Box-Ljung test

data:  fit_ili_all$residuals
X-squared = 14.199, df = 11, p-value = 0.2222
```

Figure 4.21 Ljung Box test – ARIMA (Multivariate)

The result obtained by applying Ljung Box test on the residuals of ARIMA Multivariate model



```
                 Box-Ljung test

data:  fit_ili_all_univar_manual$residuals
X-squared = 8.7107, df = 11, p-value = 0.6486
```

Figure 4.22 Ljung Box test – ARIMA (Univariate)

The result obtained by applying Ljung Box test on the residuals of ARIMA Univariate model

After checking the efficiency of the fitted models, we would move to forecast the future values by using the forecast() function in RStudio, and the forecast was as follows:

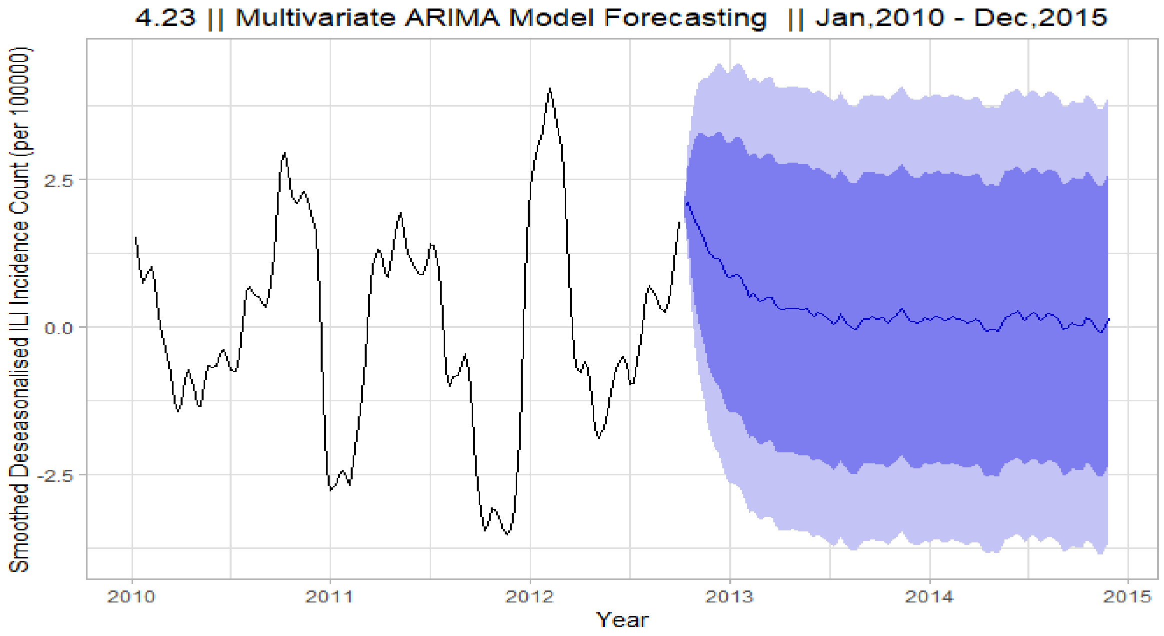


Figure 4.23 Multivariate ARIMA Model Forecasting

The plot above gives the forecasting results for Influenza activity for Multivariate Time Series model with non-zero mean

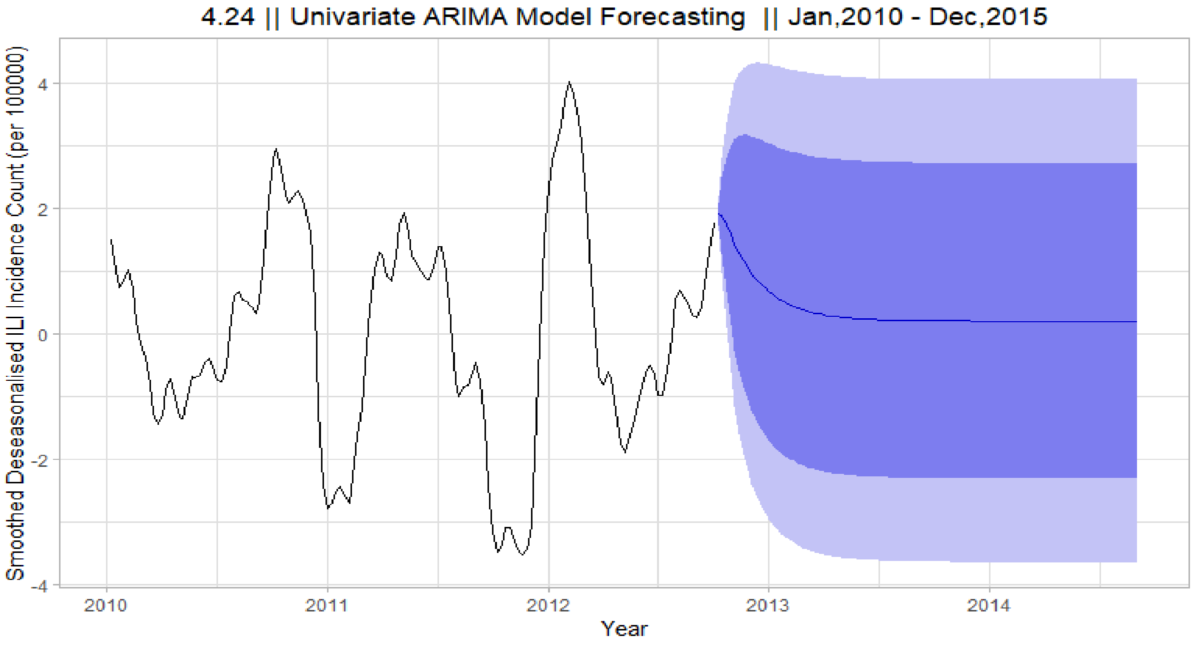
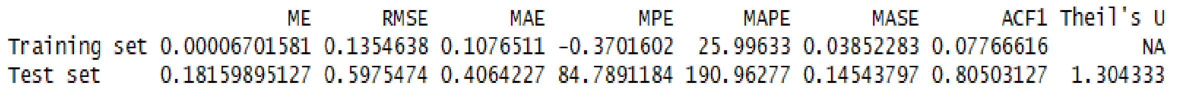


Figure 4.24 Univariate ARIMA Model Forecasting

The plot above gives the forecasting results for Influenza activity for Univariate Time Series model with non-zero mean

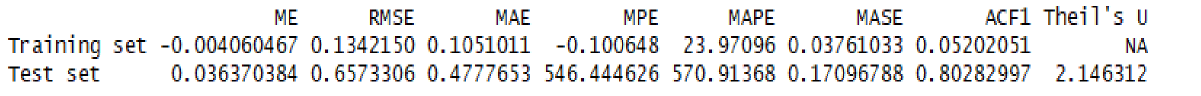
Since we have the ILI incidence count in the test set, we tried to calculate the accuracy of the model, and it was as follows:



```
                       ME      RMSE       MAE        MPE      MAPE       MASE       ACF1 Theil's U
Training set 0.00006701581 0.1354638 0.1076511 -0.3701602  25.99633 0.03852283 0.07766616        NA
Test set     0.18159895127 0.5975474 0.4064227 84.7891184 190.96277 0.14543797 0.80503127  1.304333
```

Figure 4.25 Multivariate ARIMA Model Forecasting Results

These result metrics above share the result of multivariate time series data



```
                      ME      RMSE       MAE        MPE      MAPE       MASE       ACF1 Theil's U
Training set -0.004060467 0.1342150 0.1051011  -0.100648  23.97096 0.03761033 0.05202051        NA
Test set      0.036370384 0.6573306 0.4777653 546.444626 570.91368 0.17096788 0.80282997  2.146312
```

Figure 4.26 Univariate ARIMA Model Forecasting Results

These result metrics above share the result of univariate time series data

4.2 Model Comparison Results

The table below shows both the models and compare them on the basis of RMSE value. The third column gives the RMSE value and helps in selecting the best model based on lower RMSE value.

Table 4.1 RMSE Value Comparison Table

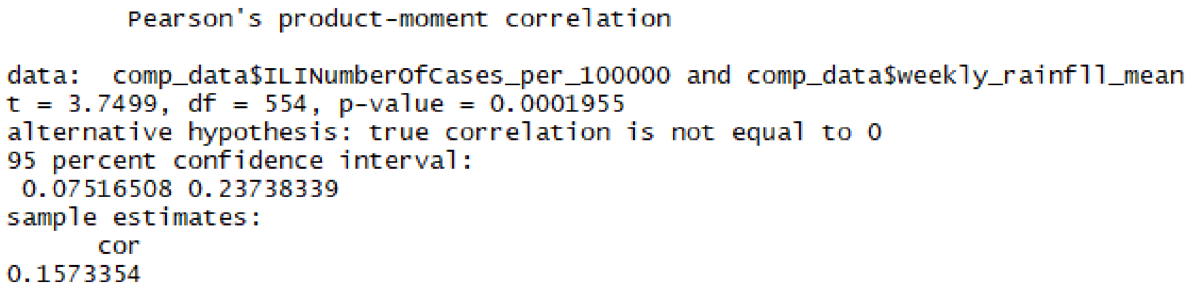
Model Number	Model Details	RMSE value
1	ARIMA Multivariate Time Series	0.597
2	ARIMA Univariate Time Series	0.657

Appendix A

The ILI, Rainfall and temperature data which were used in chapter 4 for building ARIMA model and forecasting results were tested during exploratory analysis. The figures below reveal the correlation analysis in the context of Pearson correlation coefficient and scatter plot, and the interpretation of the findings are as follows.

The Person product-moment correlation coefficient tests the linear association strength between the variables of Interest. It is denoted by the symbol ‘r’. The coefficient ‘r’ can take a value in the range from +1 to -1. A value of 0 means that there is no linear association between the two variables. A value greater than 0 means that there is a positive linear association between two variables, i.e., the increase in the value of one variable leads to the increase in the value of another variable. Similarly, a value less than 0 means that there is a negative linear association between the two variables, i.e., the increase in the value of one variable leads to the decrease in the value of another.

The Pearson’s test between ILI incidence count (per 100,000) and Weekly average rainfall reveals that the p-value (p = 0.0001955) is less than 0.05, which means that ILI incidence count and weekly average rainfall have weak positive correlation with a correlation coefficient (r) of 0.157. The same can be verified from the scatter plot (below) in the fig. A.2.



```
        Pearson's product-moment correlation

data:  comp_data$ILINumberOfCases_per_100000 and comp_data$weekly_rainfll_mean
t = 3.7499, df = 554, p-value = 0.0001955
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.07516508 0.23738339
sample estimates:
      cor
0.1573354
```

Figure A.1 Correlation Test of ILI Incidence Count vs Weekly Average Rainfall

This figure shows the results of Pearson correlation test between ILI Incidence counts and Weekly Average Rainfall

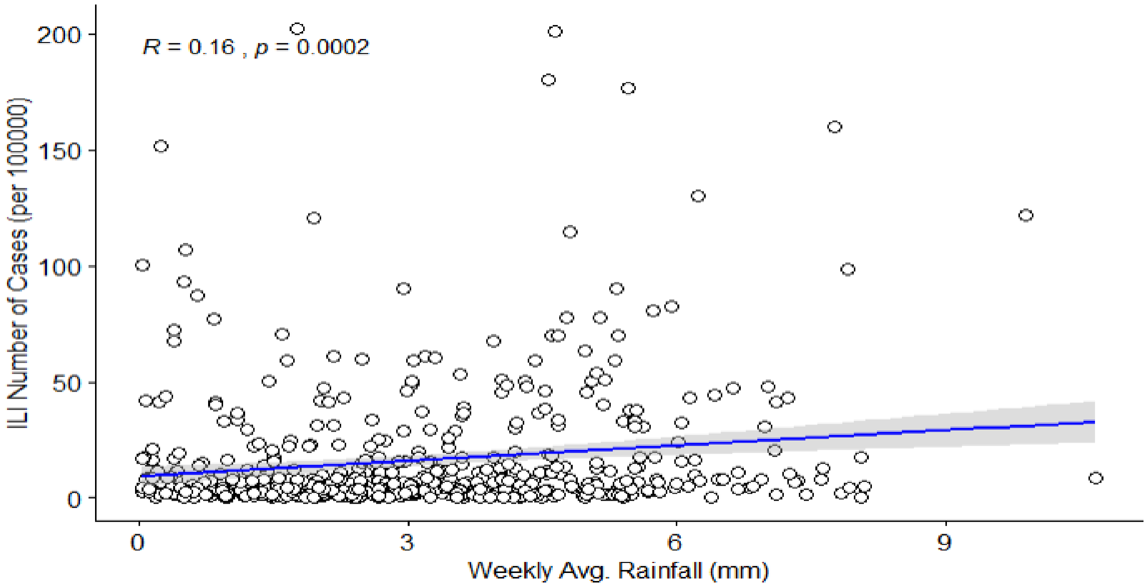
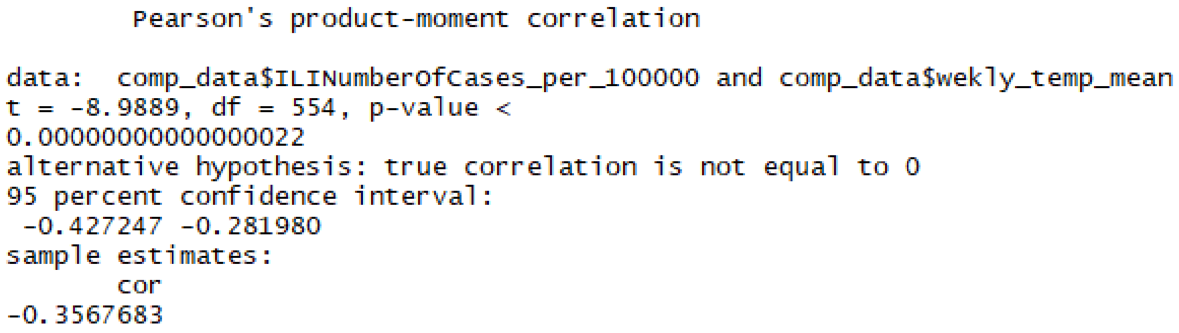


Figure A.2 Scatter Plot of ILI Incidence Count Vs Weekly Average Rainfall

The figure shows the Scatter plot between ILI Incidence counts and Weekly Average Rainfall. The slope shows a slight upward trend which demonstrates it to be weak positive relationship.

The Pearson's test between ILI incidence count (per 100,000) and Weekly average temperature reveals that the p-value (p = 0.0002) is less than 0.05, which means that ILI incidence count and weekly average temperature have moderate negative correlation with a correlation coefficient (r) of -0.357. The same can be verified from the scatter plot (below) in the fig. A.4.



```
        Pearson's product-moment correlation

data:  comp_data$ILINumberOfCases_per_100000 and comp_data$wekly_temp_mean
t = -8.9889, df = 554, p-value <
0.00000000000000022
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.427247 -0.281980
sample estimates:
       cor
-0.3567683
```

Figure A.3 Correlation Test of ILI Incidence Count vs Weekly Average Temperature

This figure shows the results of Pearson correlation test between ILI Incidence counts and Weekly Average Temperature

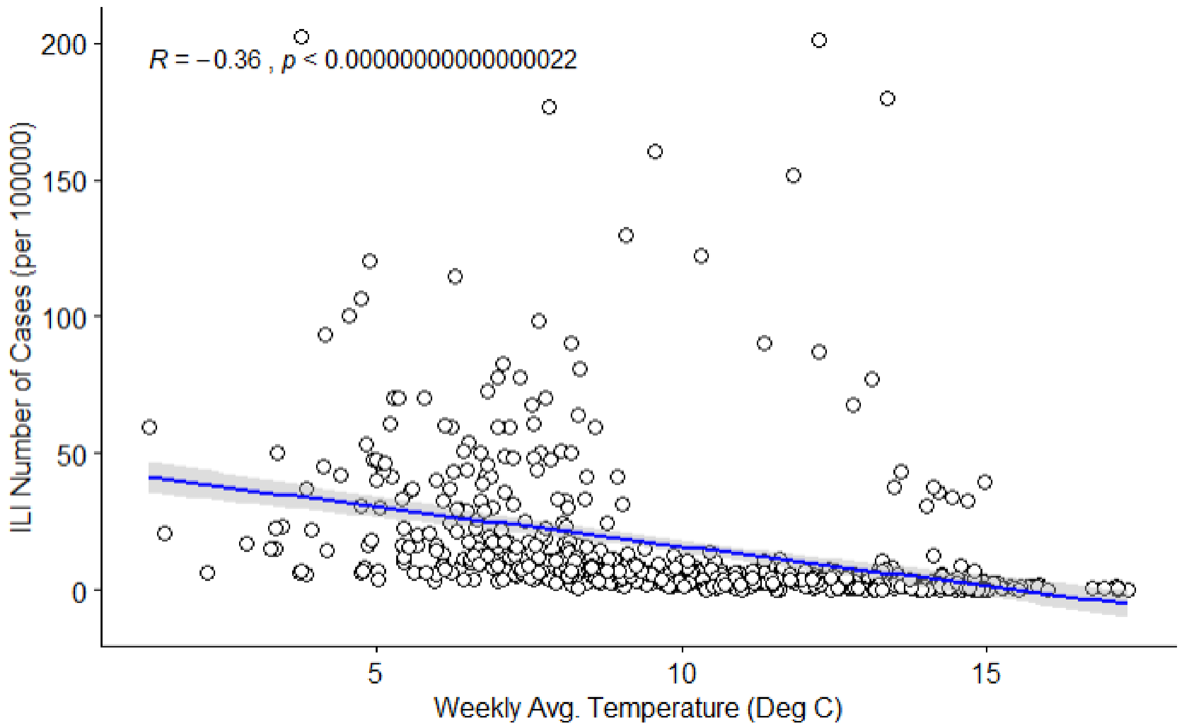


Figure A.4 Scatter Plot of ILI Incidence Count Vs Weekly Average Temperature

The figure shows the Scatter plot between ILI Incidence counts and Weekly Average Temperature. The slope shows a downward trend which demonstrates it to be moderate negative relationship.

Table A.1 Interpretation from Correlation matrix

External Factors	Correlation Coefficient Values	Interpretation
Weekly Average Rainfall	0.157	Weak positive relationship
Weekly Average Temperature	-0.357	Weak negative relationship

Table A.2 Interpretation from Scatter Plot

External Factors	Spearman Slope	Interpretation
Weekly Average Rainfall	Slight Upward slope	Weak positive relationship
Weekly Average Temperature	Moderate Downward slope	Moderate negative relationship