AI6123 - Time Series Assignment 1

G2304225K Chong Wai Kiat

Abstract

Thes main objective of this project is to fit the appropriate ARIMA model to the given dataset 'wwwusage.txt'. ADF and KPSS tests are used to check the stationarity of the data and the data after using differencing methods. First-order differencing and second-order differencing are both conducted for this project including the observations and results achieved from the prediction models proposed in this report. ACF and PACF are applied and analyzed to find the best suited model for the data. The models are further analyzed with diagnostic tests to ensure the adequacy of the model. Lastly the models are fitted with 90% (90 days) for training and 10%(10days) for validation. WE observed that the ARIMA(1,1,1) is the best model for this data with the lowest RMSE score and MAPE percentage. The ARIMA(3,1,0) might be overfitted to the training data due to its model's complexity and the ARIMA(2,2,0) is due to over-differencing the series data.

1. Dataset

In this project, we are going to fit an ARIMA model for the 'wwwusage.txt' dataset. Before we fitting the model, we perform a basic data explatory on the dataset. The dataset is retrieved using **read.csv()** method in R.

The overall mean of this dataset is 137.8, the maximum value is 228 and the minimum value is 83. We can observe that the data has a varying (increasing) trend component without a seasonal effect. Hence, the mean changes on different gaps of time. For instance, the mean value between time 20 and time 40 is different from the mean value between time 40 and time 60.

To determine the stationarity of the data, the Autocorrelation Function (ACF) can show the linear relationship between the points in the time series data separated by different time lags. In Figure 2, the acf value dies down extremely slowly which indicates that this time series is considered non-stationary. To fit the ARIMA model to this time series data, we need to determine the mean and variance of the time series are remained constant. Hence, we need to use the Differencing method to transform the data into stationary form. To ensure the stationarity of the time series, the Augmented Dickey-Fuller Test (ADF Test) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS Test) are used to test for the null hypothesis to observe the stationarity.

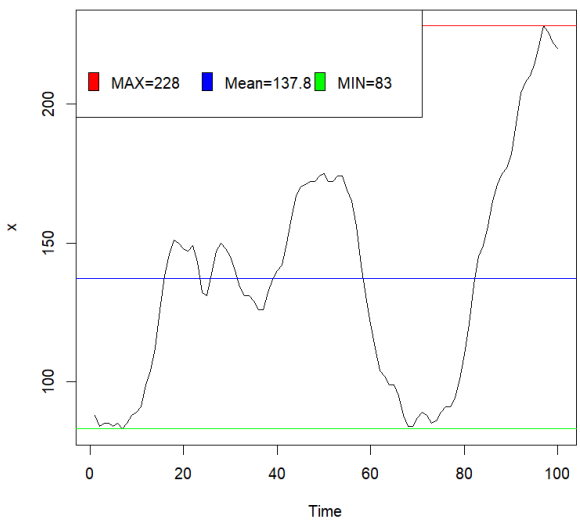

Figure 1: Original Dataset

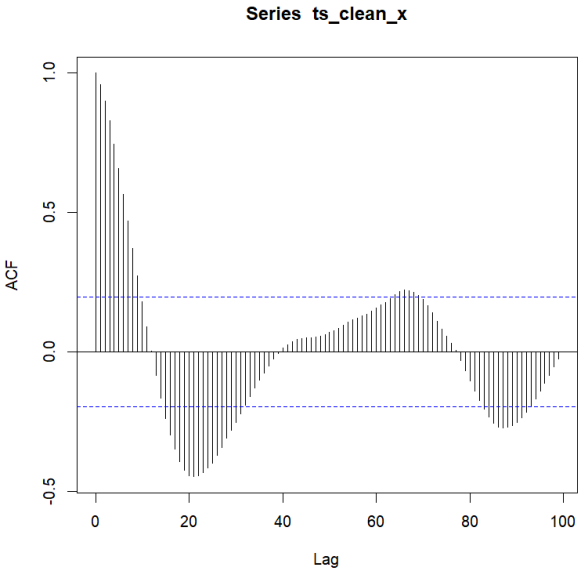

Figure 2: ACF of original dataset

ADF suggests finding the stationary time

```
adf.test(ts_clean_x,c("stationary"),k=1)
adf.test(diff(ts_clean_x,differences = 1),c("stationary"),k=1)
adf.test(diff(ts_clean_x,differences = 2),c("stationary"),k=1)

kpss.test(ts_clean_x,null="Level",lshort=TRUE)
kpss.test(diff(ts_clean_x,differences=1),null="Level",lshort=TRUE)
kpss.test(diff(ts_clean_x,differences=2),null="Level",lshort=TRUE)
```

Figure 3: ADF and KPSS setup in R

series while KPSS suggests finding the non-stationary time series. Therefore in ADF, when the p-value is less than 0.05, the time series is considered stationary. On the other hand, when the p-value in KPSS is less than 0.05, the time series is considered non-stationary. In this project, the ADF and KPSS in library(tseries) are used for the measurements. The ADF setup is shown in Figure 3, where the **k** in ADF and the **lshort** in KPSS are the lags.

Data	ADF	KPSS
Original	0.4088	0.0538
First-Order Differencing	0.01	0.1
Second-Order Differencing	0.01	0.1

Table 1: P values on tseries ADF and KPSS

Based on Table 1, we observed that the original data and first-order differenced data in ADF are both greater than 0.05 p-value. In KPSS, both of the original data and first-order differenced data are lower than 0.05 p-value. On the other hand, Second-order differenced data has both 0.01 ADF's (lower than 0.05) and 0.1 KPSS's (higher than 0.05) p-value. Hence, the time series data need to perform 2 times differencing to make the data to be stationary.

Based on Table 1, we observed that the original data has a greater than 0.05 p-value in the ADF test while a less than 0.05 p-value in KPSS test. This indicates that the original data is non-stationary. However, after First and Second-Order Differencing, we observed that both have a higher than 0.05 p-value in the ADF test and a less than 0.05 p-value in the KPSS test. hence, both differencing methods have successfully changed the data to stationary form.

2. First Order Differencing

After First-Order Differencing, we observed that the data became stationary and fluctuated around the mean (mean=1.33). The maximum value of the data is 14 while the minimum value of the data is -14. This shows that the Differencing method has reduced and balanced the distance between the maximum and minimum values in

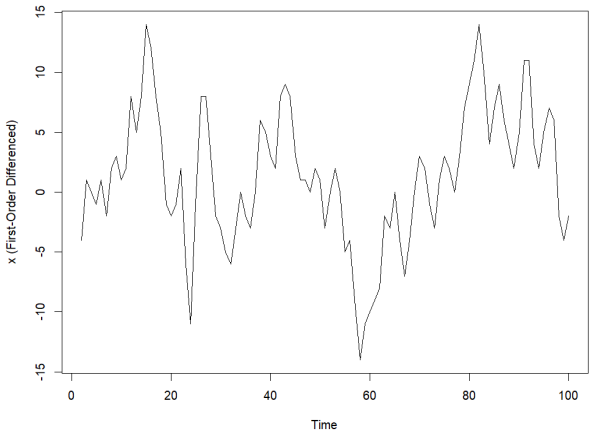

Figure 4: First-Order Differencing

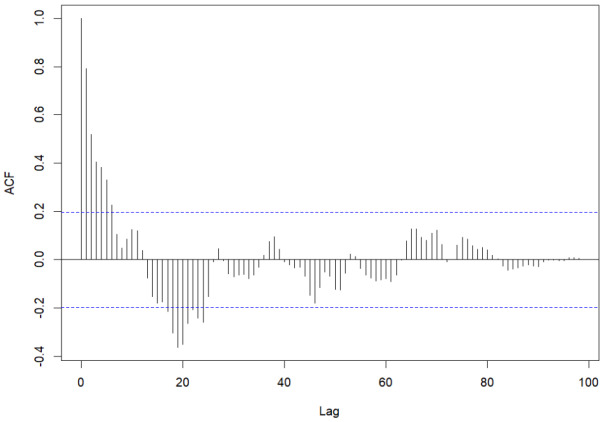

Figure 5: ACF of First-Order Differencing

the data which shows that the variance remains constant in different time lags.

Before fitting the ARIMA model, ACF and PACF are conducted to find whether the AR, MA, or ARMA model is more suitable for this data. In Figure 5, it dies down and cuts off after time lag 24 which suggests that the model should

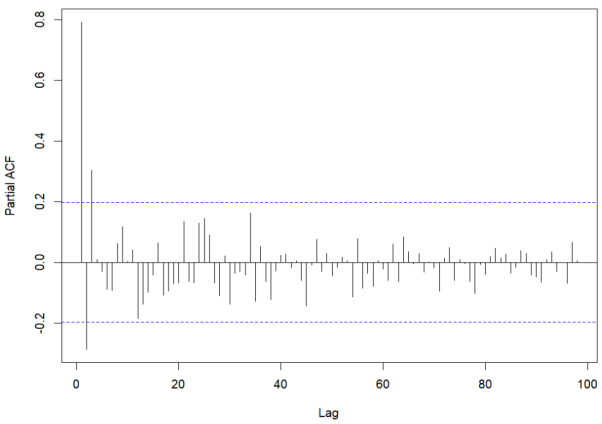

Figure 6: PACF of First-Order Differencing

```
ts.yw <- ar.yw(diff_x, order.max=5)
ts.yw
```

Figure 7: Yule Walker setup example in R

```
Coefficients:
      1        2        3
 1.1060  -0.5957   0.3029

Order selected 3  sigma^2 estimated as  10.32
```

Figure 8: Yule Walker Function of First Order Differencing

be MA(24). However, in Figure 6, it cuts off after the time lag 3 which suggests that the model should be AR(3). We observed that the PACF was cut off earlier than the ACF which was after a time lag of 24. This scenario suggests that we should implement AR for this data instead of MA or ARMA. However, since both of the ACF and PACF dies down, we may try the lower order of ARMA model for fitting this data.

On the other hand, to find the best order for the AR model, the Yule-Walker function is fitted to find the best order of coefficient of the model AR for this data. Figure 7 shows the setup of using Yule-Walker in R. Figure 8 shows that the best order is 3 for this First-Order Differenced data. Based on the result from the Yule-Walker Function in Figure 8 and the PACF result in Figure 6, the suggested model for this Second-Order Differencing series data is AR(3). Hence, the ARIMA(3,1,0) which represents the AR(3) model and ARIMA(1,1,1) which represents the ARMA(1) are built to fit the original data. Figure 9 shows an example setup of the ARIMA model (ARIMA(3,1,0))in R where the **3** represents AR(3), **1** represents first-order differencing and **0** represents MA(0).

Before fitting the model for prediction and comparison, we need to conduct diagnostic tests for the models mentioned to ensure the model is adequate. **tsdiag(model)** is used in R to generate the diagnostic test result of the model. Figure 10 shows a basic diagnostic test for the ARIMA(3,1,0) model. From a standardized residuals perspective, the ARIMA(3,1,0) model's residuals are random without any bias pattern. The ACF of the residuals cut off after lag 0. For Ljung-Box statistic, **Box.test(model$residuals, type** ¯¨Ljung-Box¨) is used to check the p-value. The p-value for Ljung-Box statistic is 0.9749 which is far greater than 0.05, as shown in the graph. In Figure 11, the residuals fluctuate randomly along the horizontal axis, indicating that the model's prediction is unbiased. From the Histogram in Figure 12, we can check the normality of the residuals that the residuals are fitted approximately a normal distribution. Hence, based on the conditions above mentioned, the ARIMA(3,1,0) is an adequate model and its AIC is 511.99.

```
model <- arima(ts_clean_x, order=c(3,1,0))
model
```

Figure 9: Example of ARIMA model setup in R (ARIMA(3,1,0))

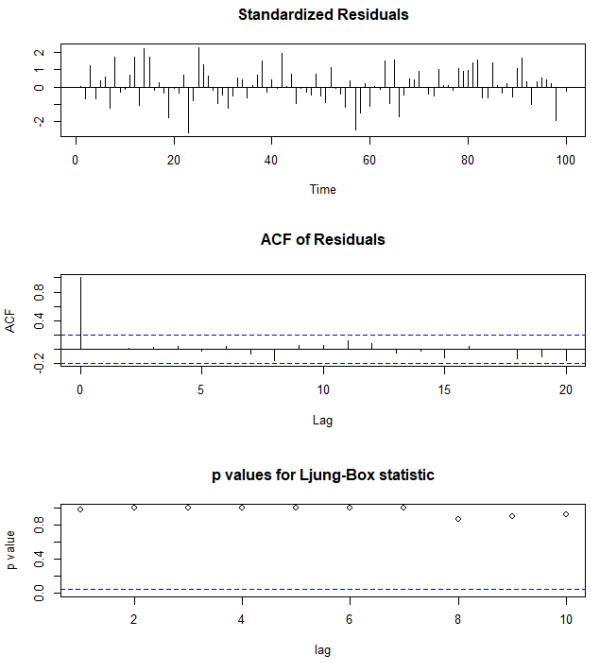

Figure 10: Diagnostic Test of ARIMA(3,1,0)

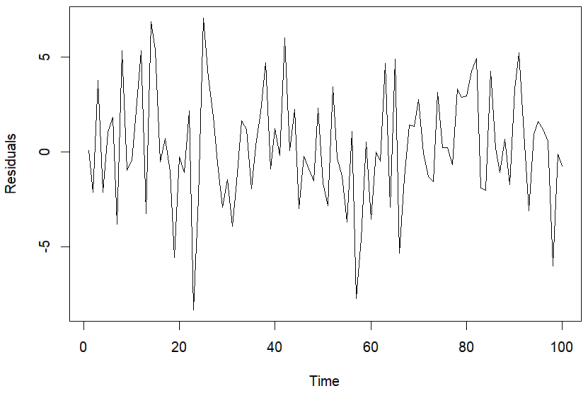

Figure 11: ARIMA(3,1,0) Residuals

For every model, we will determine whether the model is adequate based on the conditions aforementioned and test the ARIMA(3,1,0) model before fitting it for prediction and com-

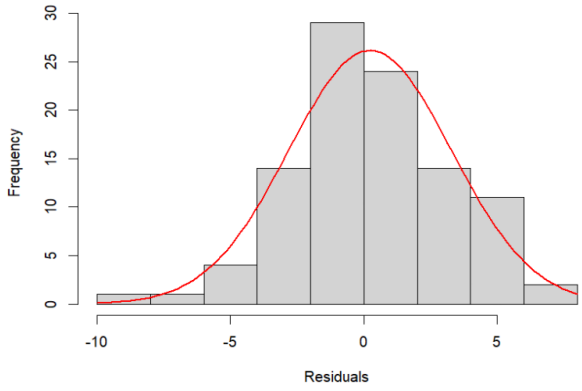

Figure 12: Histogram of residuals of ARIMA(3,1,0)

parison. The ARIMA(1,1,1) model's standardized residuals are random enough which did not show any bias pattern, as shown in Figure 13. The ACF cuts off after lag 0 and the p-value for the Ljung-Box statistic is 0.8618 which is highly above the p-value of 0.05. In Figure 14, the residuals are fluctuating along the horizontal axis which represent the model's prediction is unbiased. In Figure 15, the histogram is approximately a normal distribution to show the normality of the residuals. Hence, the ARIMA(1,1,1) is also an adequate model and its AIC is 514.3.

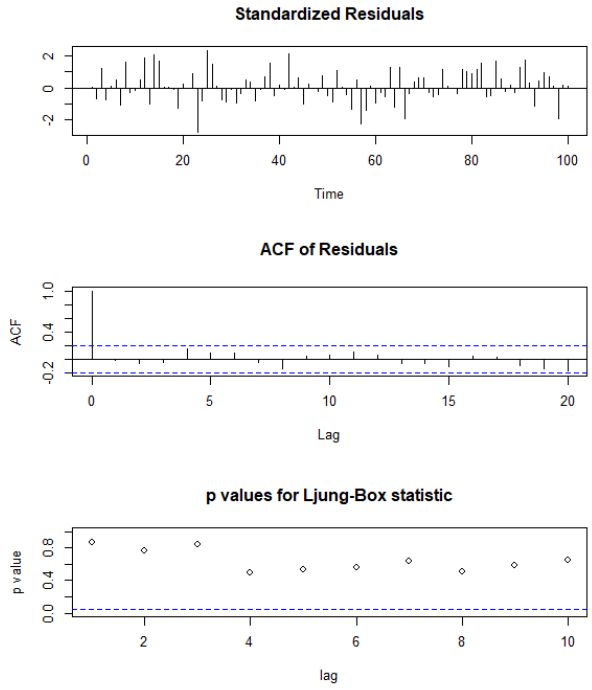

Figure 13: Diagnostic Test of ARIMA(1,1,1)

Since both of the models are adequate model, we can conduct predictions on both models to compare their accuracies. Before conducting the prediction, we split our data in a ratio of 90:10

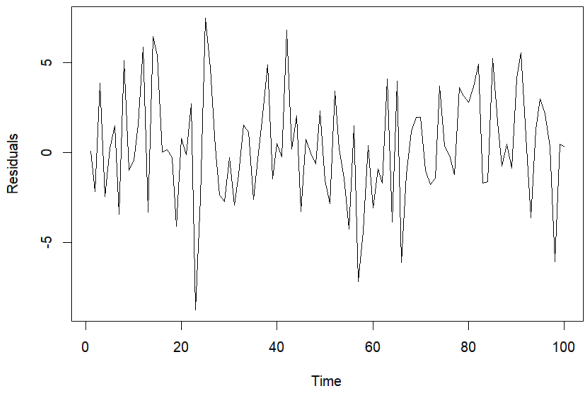

Figure 14: ARIMA(1,1,1) Residuals

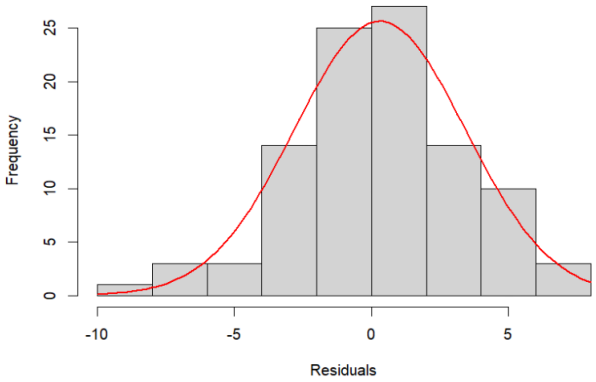

Figure 15: Histogram of residuals of ARIMA(1,1,1)

```
modeltrain <- arima(ts_clean_x[0:91], order=c(3,1,0))
modeltrain
forecast <- predict(modeltrain,n.ahead=10)
forecast
```

Figure 16: Example of training and prediction setup

with 90% for training data and 10% for testing data. The 90% training data (90 days) will fit into the model for training while the 10% testing data (10 days) is used to validate the model's accuracy. Figure 16 shows an example of the setup for training and validation of the model. Figure 17 shows the prediction of the ARIMA(3,1,0) model on the next 10 days compared to the actual data of the next 10 days. Figure 18 shows a better prediction from the ARIMA(1,1,1) model compared to the ARIMA(3,1,0) model.

3. Second Order Differencing

In First-Order Differenced result data shown in Figure 4, we observed that although the ADF and KPSS result shows that the data is station-

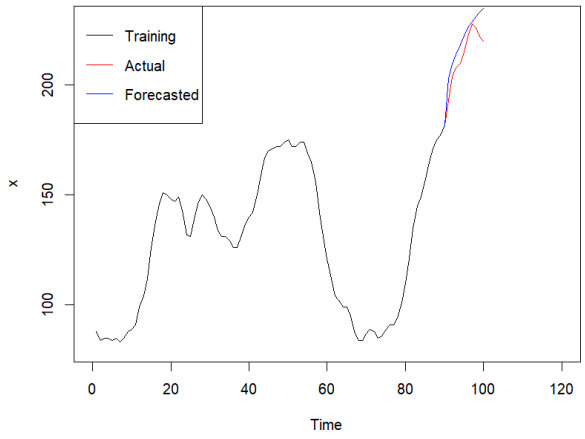

Figure 17: ARIMA(3,1,0) Forecasting 10 days ahead

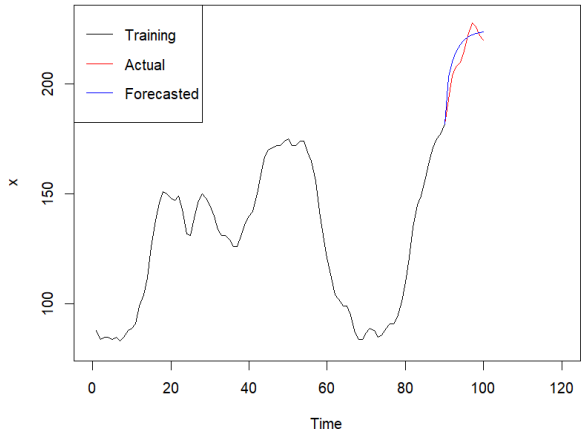

Figure 18: ARIMA(1,1,1) Forecasting 10 days ahead

ary, the data seems to be not stationary. Therefore, Second-Order Differencing is conducted for comparison to find the best ARIMA model.

When applying Second-Order Differencing, we observed that the data became more stationary and fluctuated around the mean (mean=0.2041). The maximum value of the data is 10 while the minimum value of the data is -8. This shows that Second-Order Differencing has a reduced and better-balanced distance between the maximum and minimum values in the data compared to First-Order Differencing.

Subsequently, we check for the ACF and PACF for selecting the AR, MA or ARMA model. In Figure 20, it dies down and cuts off at time lag 46 which suggests the MA(46) for the data. However, in Figure 21, it cut off after time lag 2 which suggests the AR(2) for the

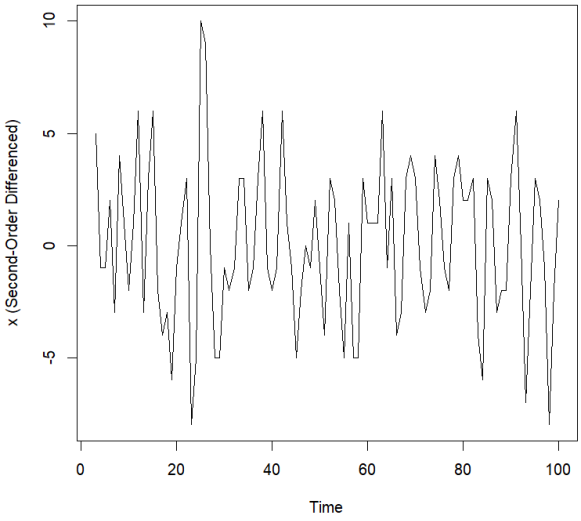

Figure 19: Second-Order Differencing

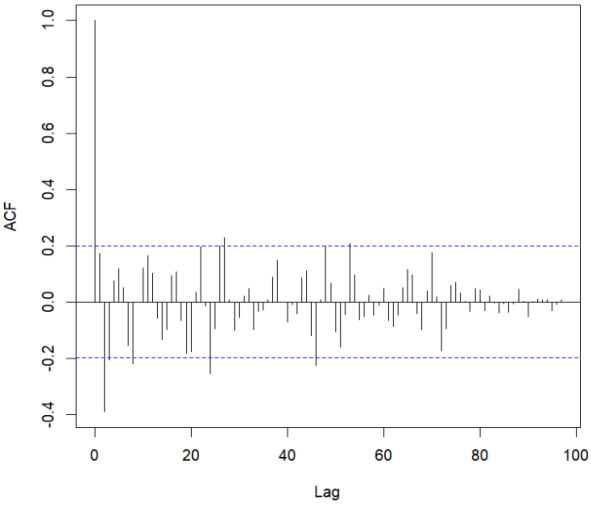

Figure 20: ACF of Second-Order Differencing

data. Hence, based on the ACF and PACF, we conclude that the ACF dies down quickly and PACF cuts off at lag 2. This scenario suggests that we should implement AR for this data instead of MA or ARMA.

On the other hand, to find the best order for the AR model, the Yule-Walker function is fitted to find the best order of coefficient of the model AR for this data. It shows that the best order is 2 for this data, shown in Figure 22. Based on the result from the Yule-Walker Function in Figure 22 and the PACF result in Figure 21, the suggested model for this Second-Order Differencing series data is AR(2). Hence, the ARIMA(2,2,0) which represents the AR(2) model, is built to fit the original data.

Figure 23 shows a basic diagnostic test on

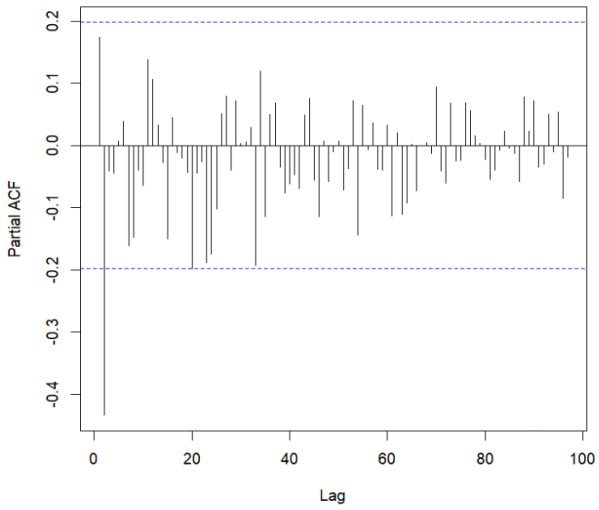

Figure 21: PACF of Second-Order Differencing

```
Coefficients:
      1        2
 0.2489  -0.4341

Order selected 2  sigma^2 estimated as  10.56
```

Figure 22: Yule-Walker Function of Second-Order Differencing

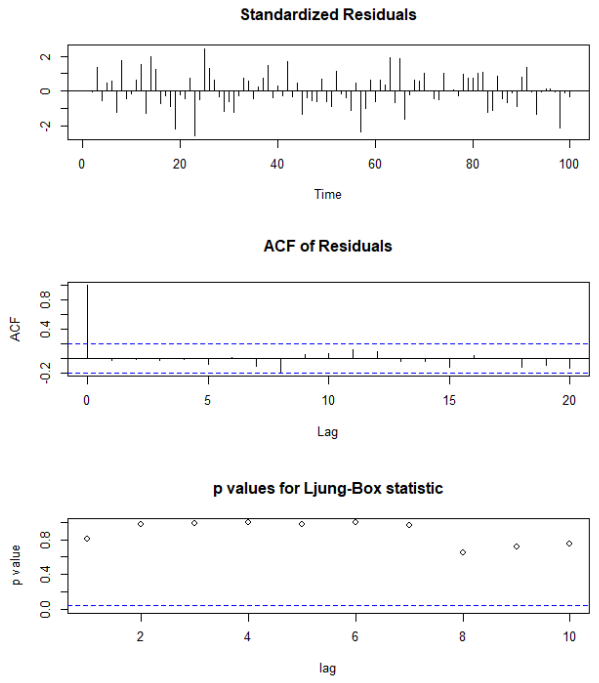

Figure 23: Diagnostic Test of ARIMA(2,2,0)

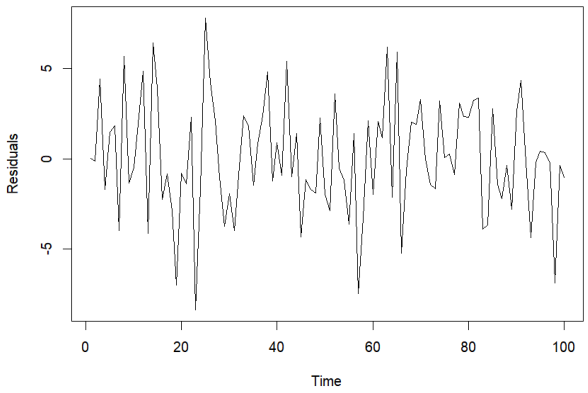

Figure 24: ARIMA(2,2,0) residuals

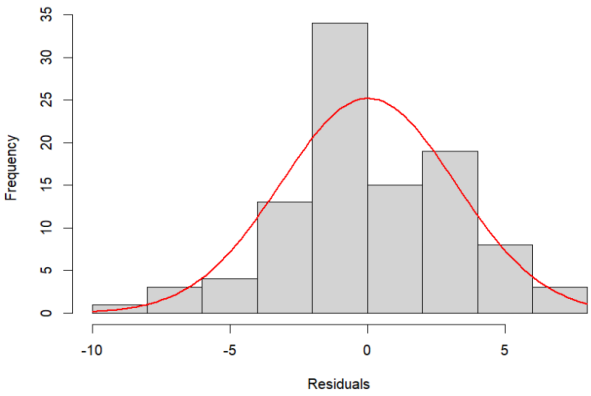

Figure 25: Histogram of residuals of ARIMA(2,2,0)

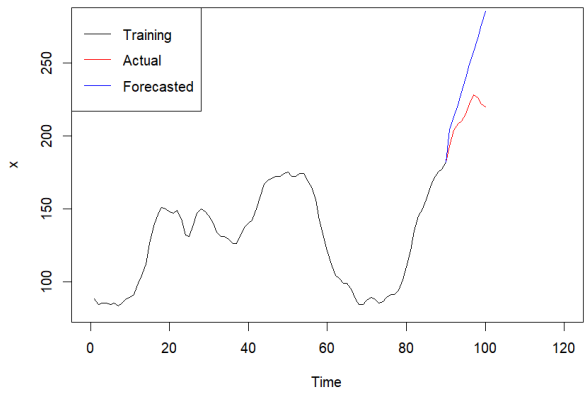

Figure 26: ARIMA(2,2,0) Forecasting 10 days ahead

the ARIMA(2,2,0) model. The ARIMA(2,2,0) model's standardized residuals are random enough and did not show any bias pattern, as shown in Figure 23. The ACF cuts off after lag 0 and the p-value for the Ljung-Box statistic is 0.8111 which is highly above the p-value of 0.05. In Figure 24, the residuals fluctuate along the horizontal axis which represents the model's prediction is unbiased. In Figure 25, the histogram is approximately a normal distribution to show the normality of the residuals. Hence, the ARIMA(1,1,1) is also an adequate model and its

AIC is 511.46. Figure 26 shows the result of the ARIMA(2,2,0) model predicting 10 days ahead compared to the actual data.

4. Comparison of Different Models

In Table 2, we observed that the best model is ARIMA(1,1,1) which has the lowest RMSE score (3.2447) and MAPE percentage (2.4063%) of errors. Secondly, the ARIMA(3,1,0) model is the second best model in this project with a 7.59 RMSE and 3.4704% MAPE of errors. However, during the training, the ARIMA(3,1,0) had 0.0496 fewer errors compared to ARIMA(1,1,1). Hence, the ARIMA(3,1,0) probably had overfitted to the training data causing bad generalization on new data. This is due to the ARIMA(3,1,0) is a more complex model (3 variables) compared to the ARIMA(1,1,1) model (2 variables where 1 for AR and 1 for MA). On the other hand, the ARIMA(2,2,0) model has the worst performance among all other models. This might be due to overdifferencing the time series data leading to negative auto-correlation between the time lags. For instance, the current value is negatively correlated to the previous value, leading to a decreasing pattern although should be an increasing pattern. Comparing the AIC between three different models in Table 3, we observed that the ARIMA(2,2,0) has the lowest AIC which should be chosen for the prediction. However, the best overall performance is the ARIMA(1,1,1) model, although the AIC value is the highest. This might further proven that the ARIMA(3,1,0) has overfitted the training data and the ARIMA(2,2,0) is due to overdifferencing as the AIC value is close to the ARIMA(3,1,0) model.

ARIMA model	**(Train)RMSE**	**RMSE**	**MAPE**
ARIMA(3,1,0)	**3.1155**	7.3900	3.4704
ARIMA(1,1,1)	3.1651	**3.2447**	**2.4063**
ARIMA(2,2,0)	3.2222	29.6081	13.5364

Table 2: Accuracies of different models

ARIMA model	**AIC**
ARIMA(3,1,0)	511.99
ARIMA(1,1,1)	514.3
ARIMA(2,2,0)	**511.46**

Table 3: AIC of different models