TIME SERIES

Ricardo Chiang Cornejo
Muhammad Naufal Aniq Bin Khairol Amali
Adam Green

INTRODUCTION

Forecasting Airport Traffic with Time Series

	Date	Traffic
1	1998/01	234.2
2	1998/02	211.1
3	1998/03	228.2
4	1998/04	244.6
5	1998/05	225.3
6	1998/06	231.8
7	1998/07	257.0

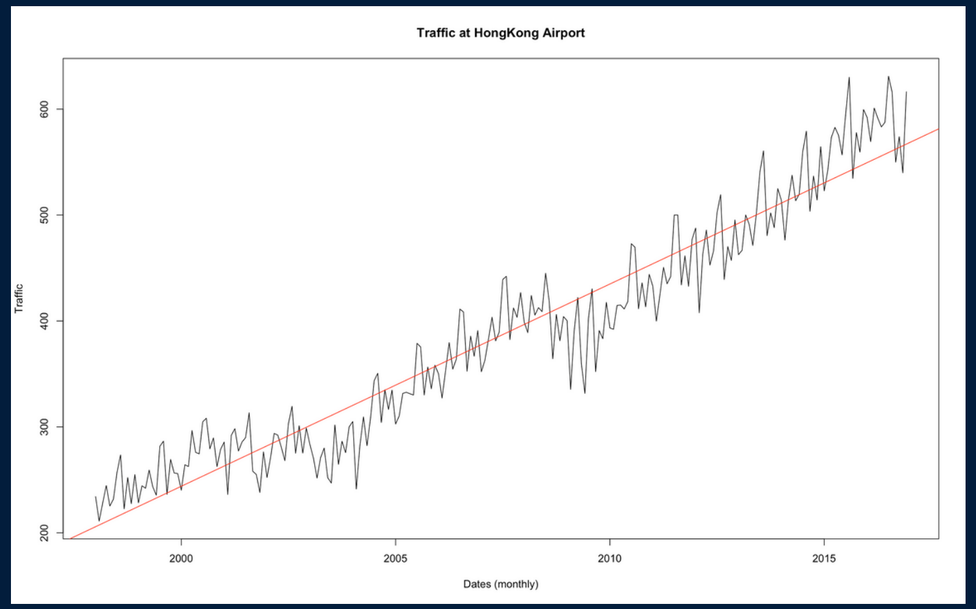
Overview

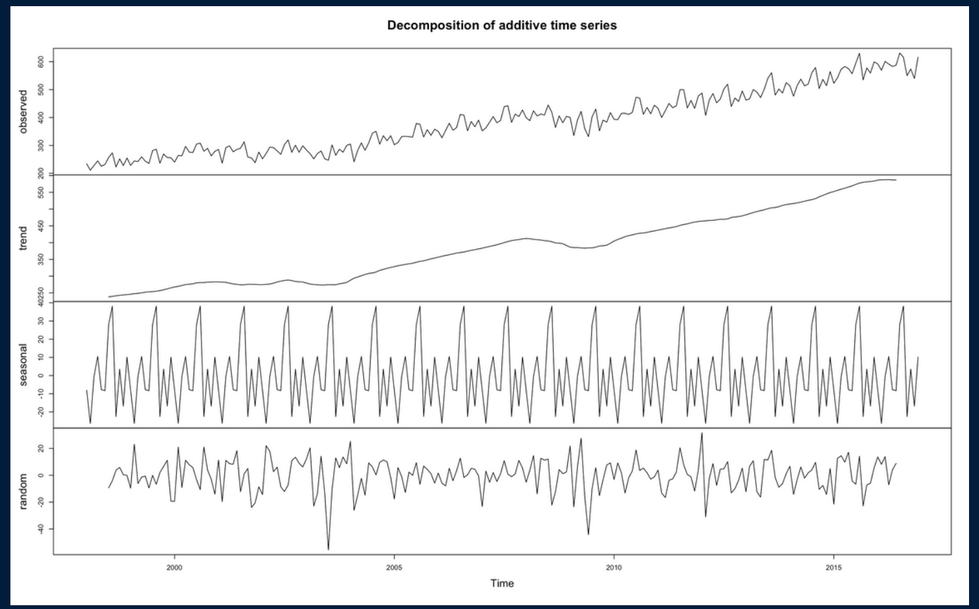
This presentation focuses on forecasting airport traffic at Hong Kong Airport from 1998 to 2016 starting with using the Box-Jenkins methodology. Following this we apply different time series statistical techniques to create a predictive model. This techniques and processes leading up to our final model will be seen in the upcoming slides.

Importance

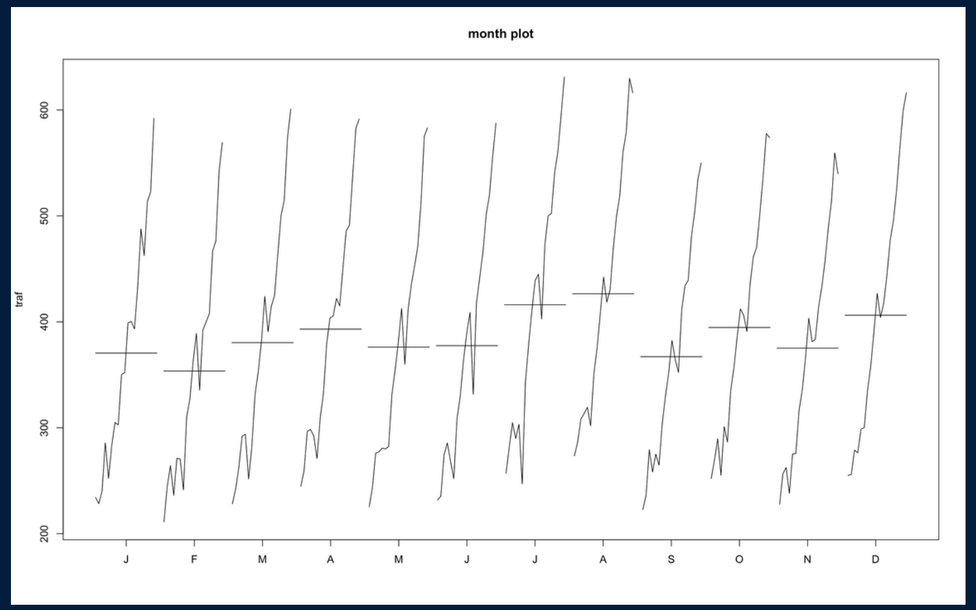
Understanding and predicting airport traffic trends is vital for efficient resource allocation, operational planning, and strategic management at airports.

Dataset Overview: Starting Point





Data from 1998 to 2016

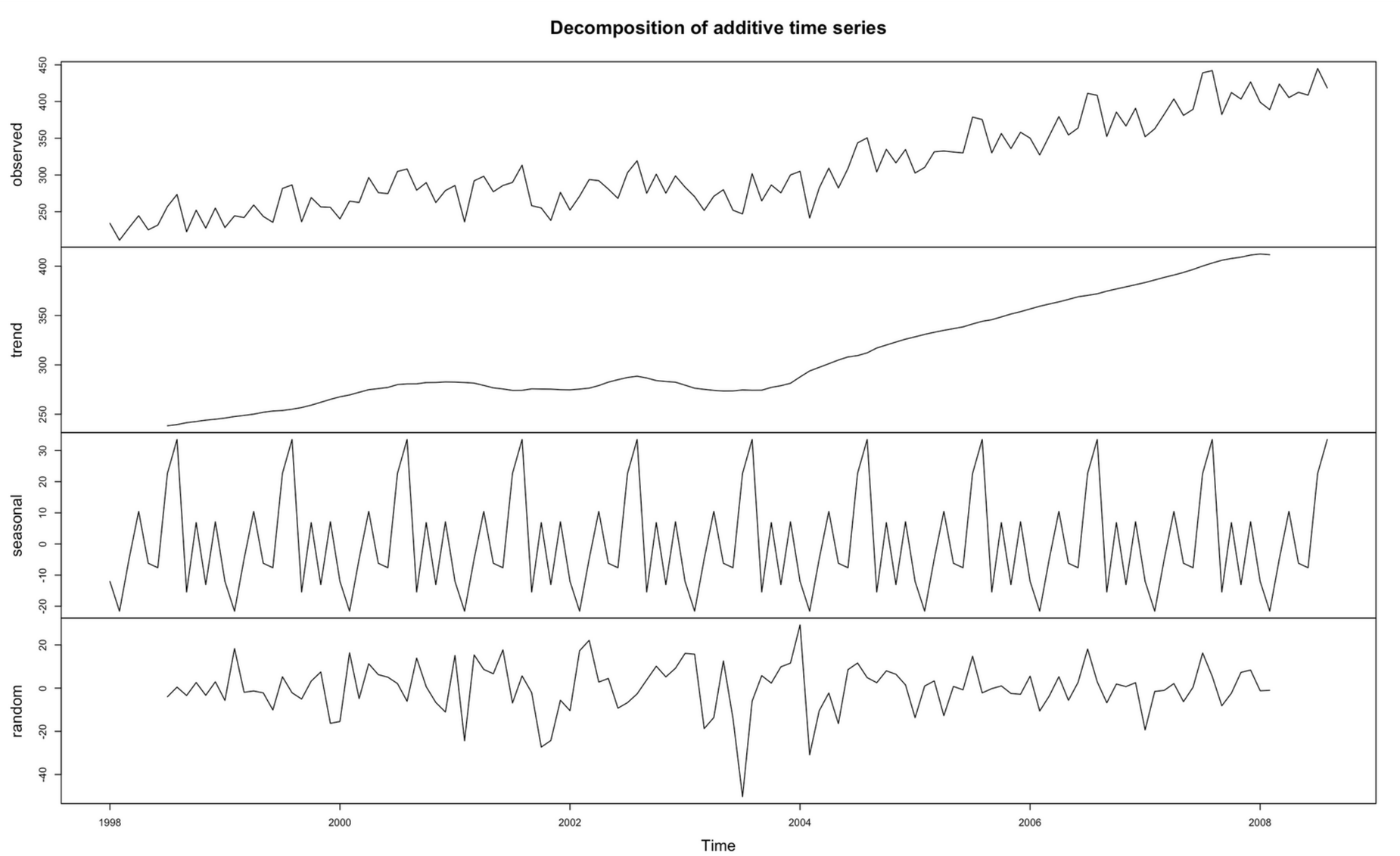


Key Take aways:

- The series is not stationary: trend, seasonal effects
- We decide to split the sample around 2008, January to avoid the change of trend

Split Data:

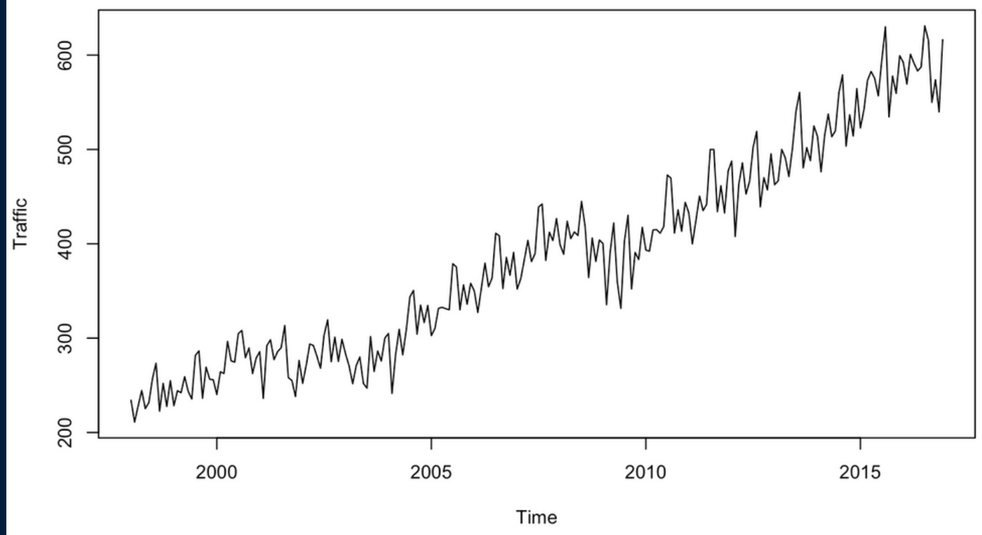
Data from 1998 to 2008

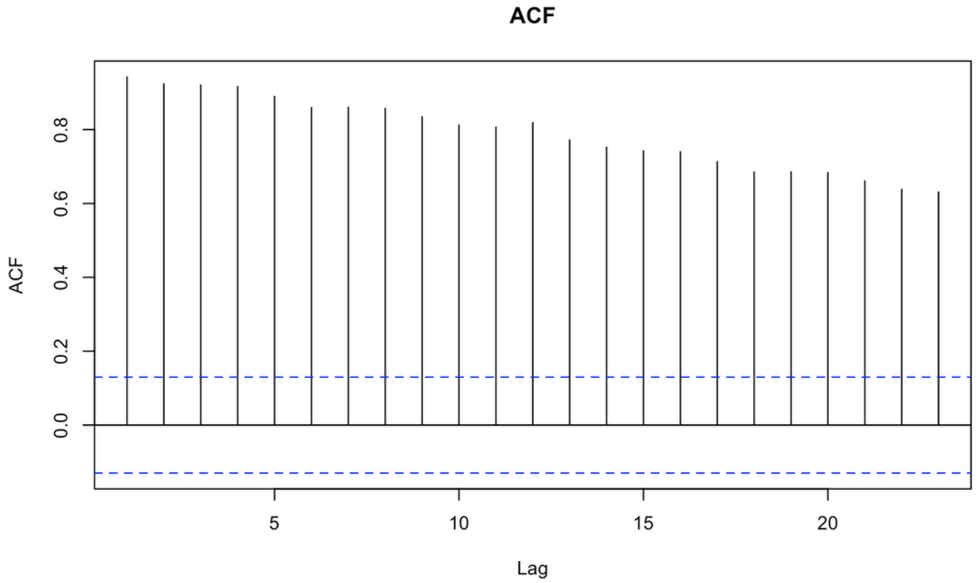


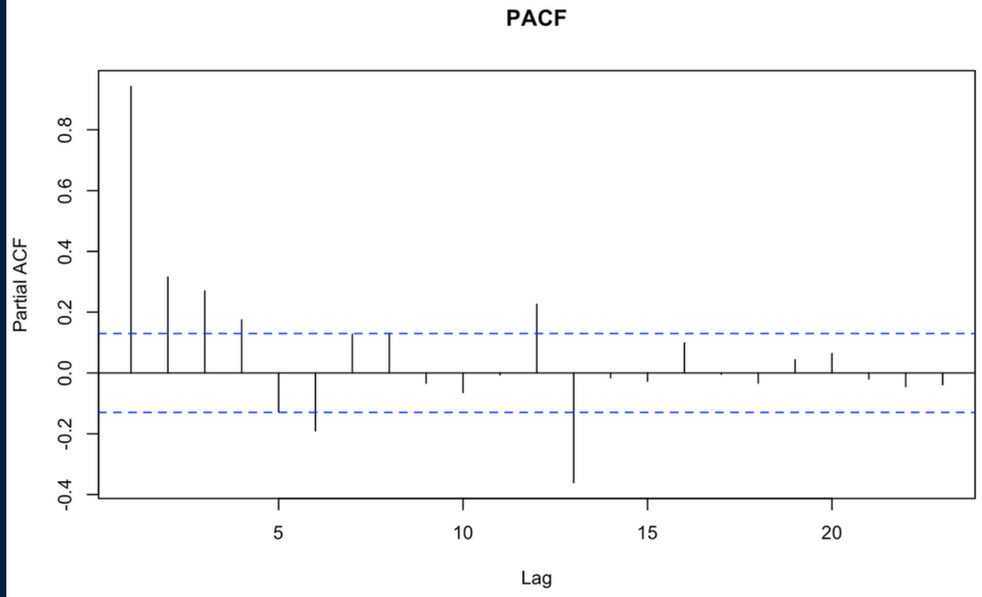
we split the observed graphs, we can see that the graph has:

- trend
- seasonality
- randomness

What we notice after Split:







Key Take aways:

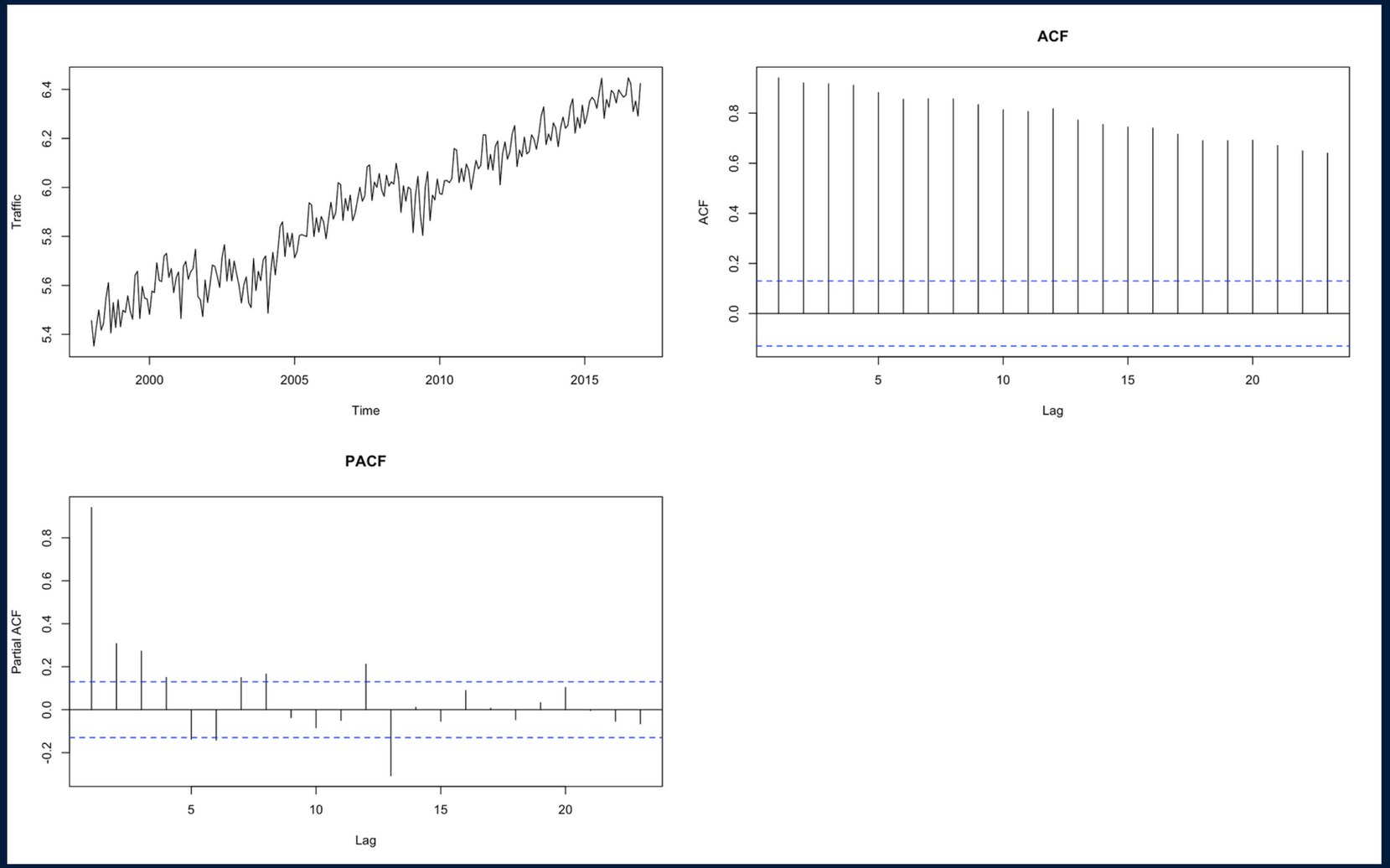
- As said in the previous slide, the series is non stationary: we observe a trend and seasonal effects
- The ACF has stochastic trend setting

Make Data Stationairy
(Log of Data, First Order Difference of Log & Removing Seasonality):

Part 1:

Log of Data Results:

- Thus stabilizing variance
- We do this to make the data more interpretable
- Allows later modelling

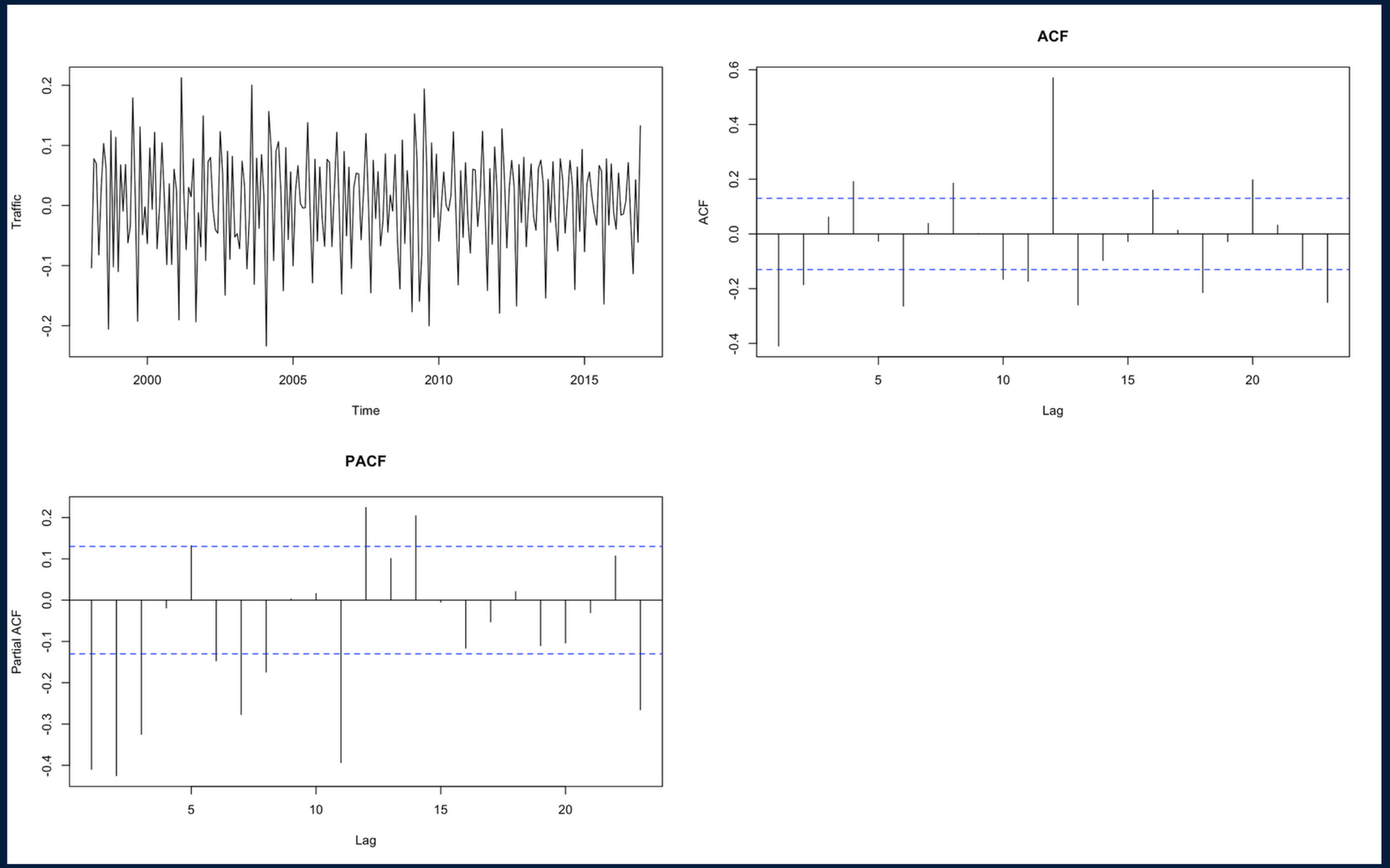


Make Data Stationairy
(Log of Data, First Order Difference of Log & Removing Seasonality):

Part 2:

First Order Difference Results:

- We see we have successfully removed the trend

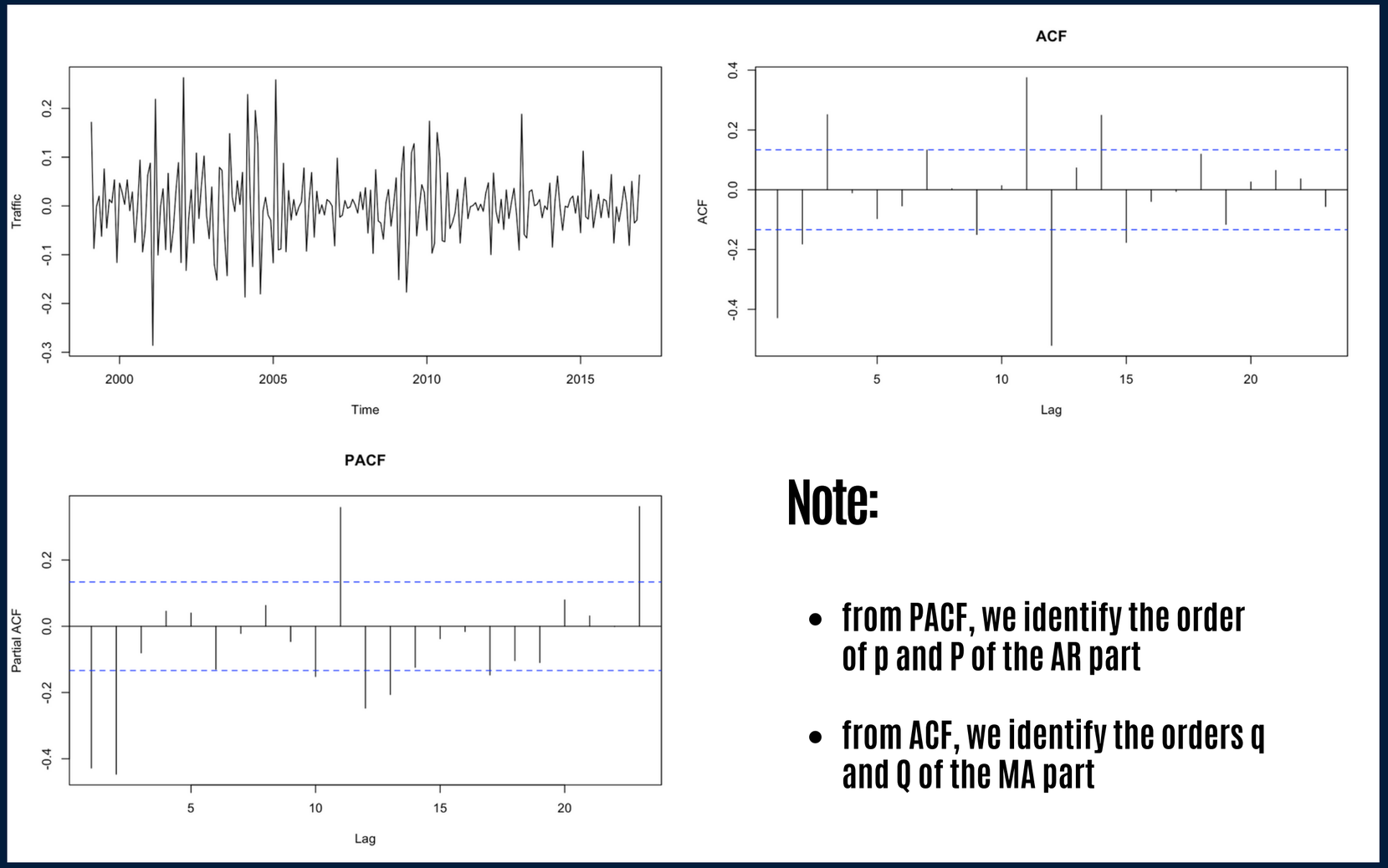


Make Data Stationairy
(Log of Data, First Order Difference of Log & Removing Seasonality):

Part 3:

Removing Seasonality Results:

- We see we have successfully removed seasonality
- Data is stationairy as its decaying fast to zero



Note:

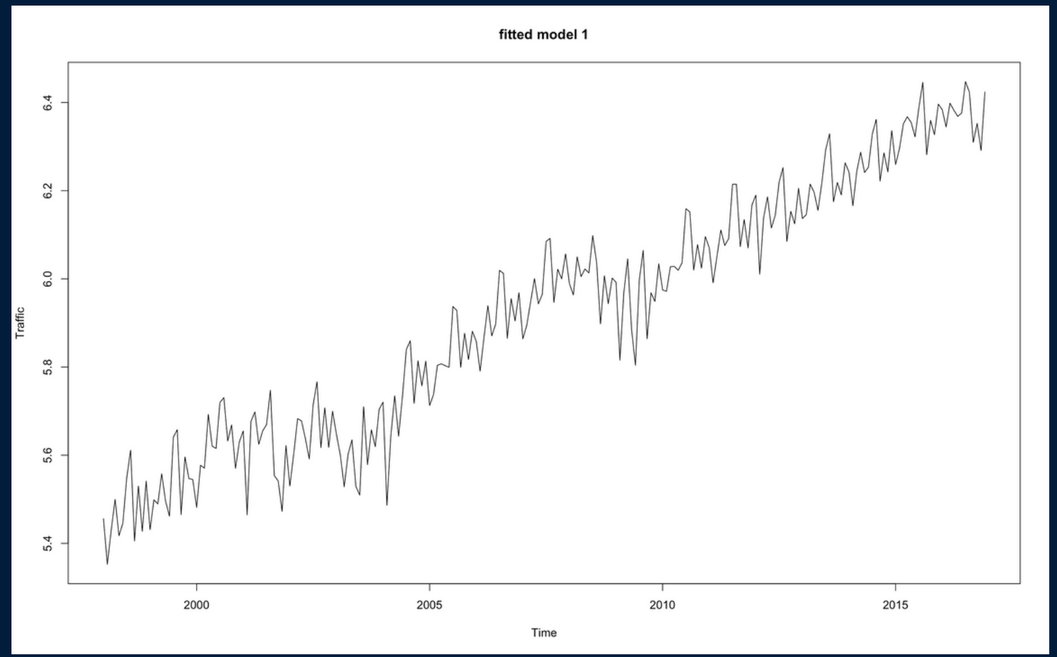
- from PACF, we identify the order of p and P of the AR part
- from ACF, we identify the orders q and Q of the MA part

Estimation of First multiplicative SARIMA model:

```
Call:
arima(x = ltraf, order = c(2, 1, 1), seasonal = list(order = c(1, 1, 1), period = 12),
    method = "ML")

Coefficients:
          ar1      ar2      ma1     sar1     sma1
      -0.3668  -0.2892  -0.2234  -0.0739  -0.9977
s.e.   0.1799   0.1037   0.1864   0.0720   0.3531

sigma^2 estimated as 0.001757:  log likelihood = 358.44,  aic = -706.88
> # AIC=-706.88
```

Key Take aways:

- AIC is pretty strong but can definitley be better!
- This is a fair starting point for a first model
- Remember we split at 2008 -->



Validation Step Meta Overview:

Methodology:

- First we evaluate how well the model fits to the data coefficients significance; residuals; parameters of how well model fits to data and then do an evaluation of fit based off the AIC.

- Second we evaluate how well the model predicts new values. We use build confidence intervals around our model and we will check if the observed data is between confidence interval for our in-sample / out-of-sample analysis

Validation & Significance of Coefficients:

Methods:

- Residuals analysis: Gaussian white noise
- Used to infer prediction accuracy
- We iterate this process 3 times with different p,P,q,Q values to get the lowest AIC score
- Can be viewed in our code

Key Take Away:

Model 3 is our best since AIC is smaller

Results model 1: (p=2, P=1, q=1, Q=1)

```
#         ar1         ar2         ma1         sar1         sma1
# 0.041439865 0.005267540 0.230601043 0.304787131 0.004721735
```

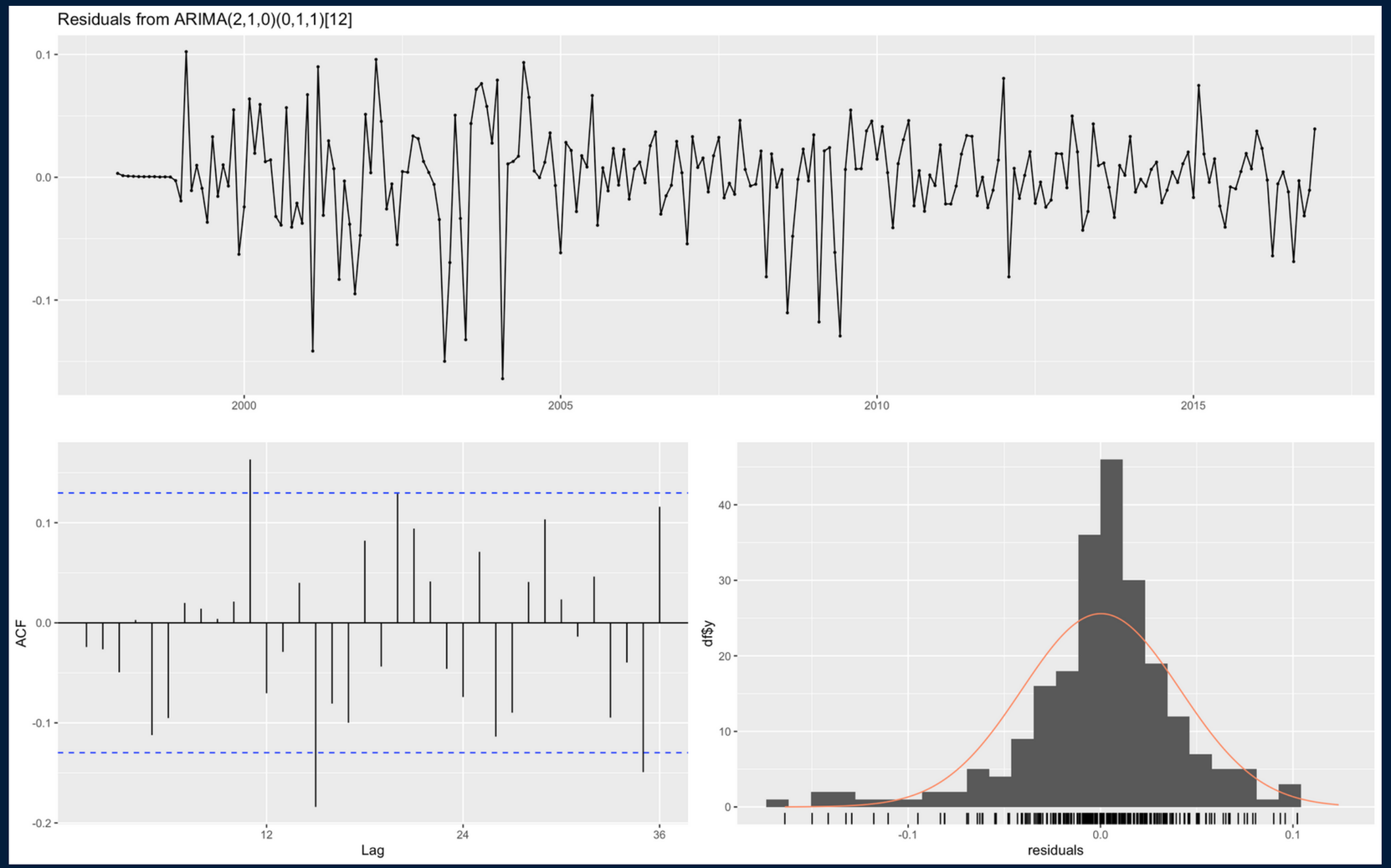
Results model 2: (p=2, P=0, q=1, Q=1)

```
#         ar1         ar2         ma1         sma1
#0.017920714 0.001735534 0.230381894 0.000000000
```

Results model 3: (p=2, P=0, q=0, Q=1)

```
#         ar1         ar2         sma1
#0.000000e+00 2.078052e-09 3.552714e-15
# pvalues for AR1, AR2 and SMA1 are smaller than 5%
```

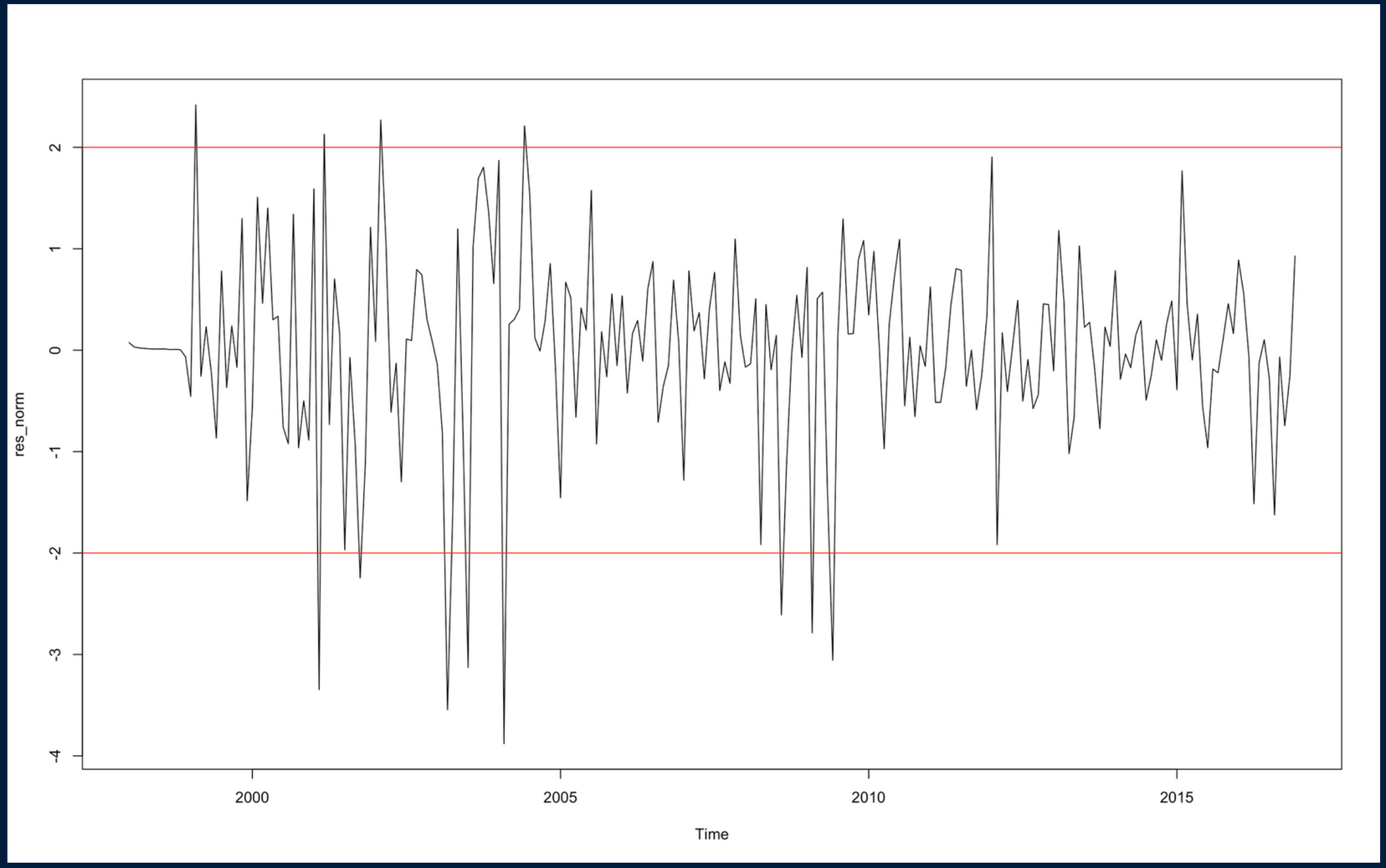
Residuals Analysis:



What we Know:

- The ACF of the residuals has no significant coefficient
- The residuals behave like a white noise
- The histogram has bell shape close to a normal distribution

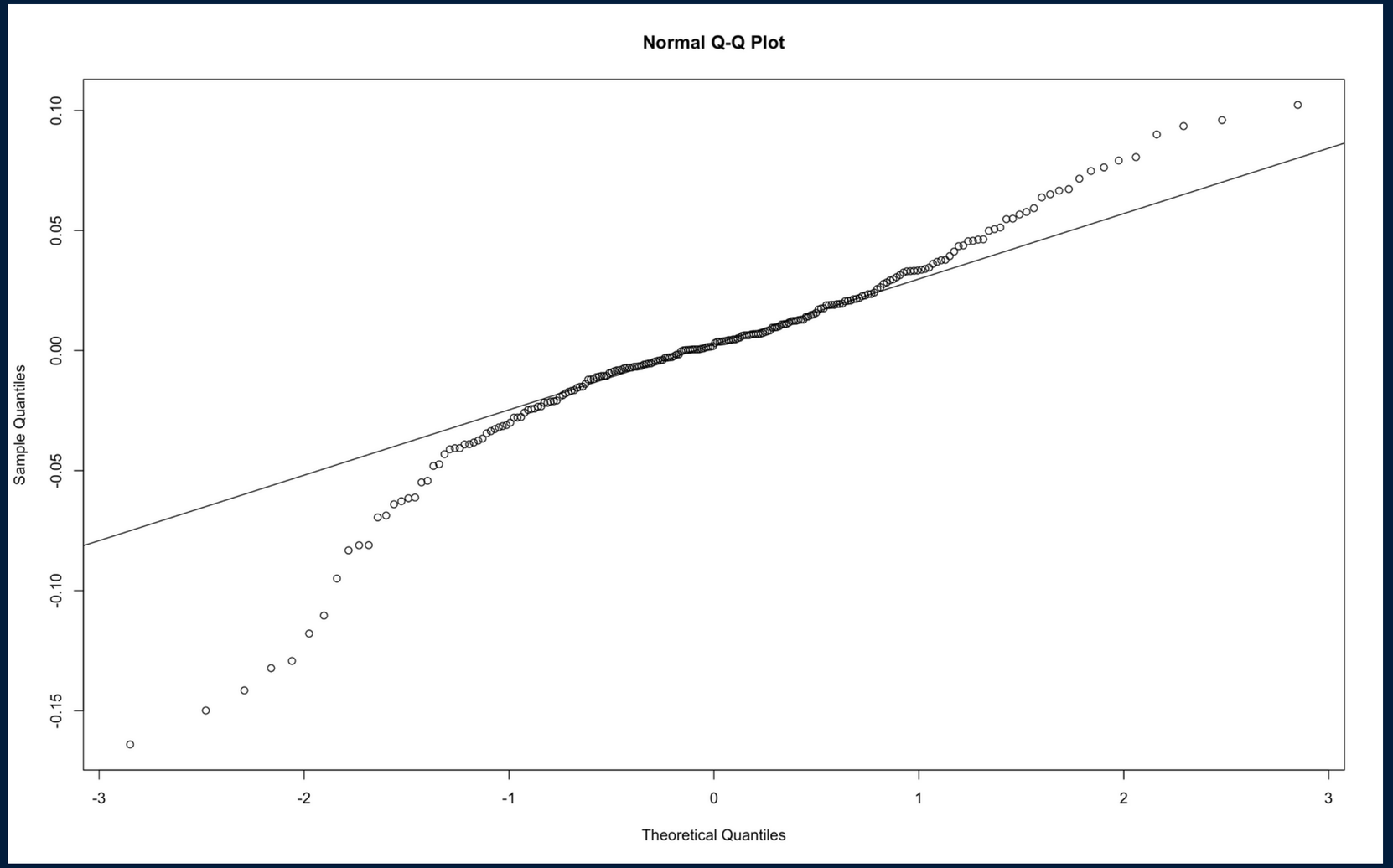
Standardize Residuals:



We see:

- There are four points outside the interval,
- we will only use 1 dummy variable because the model already fits

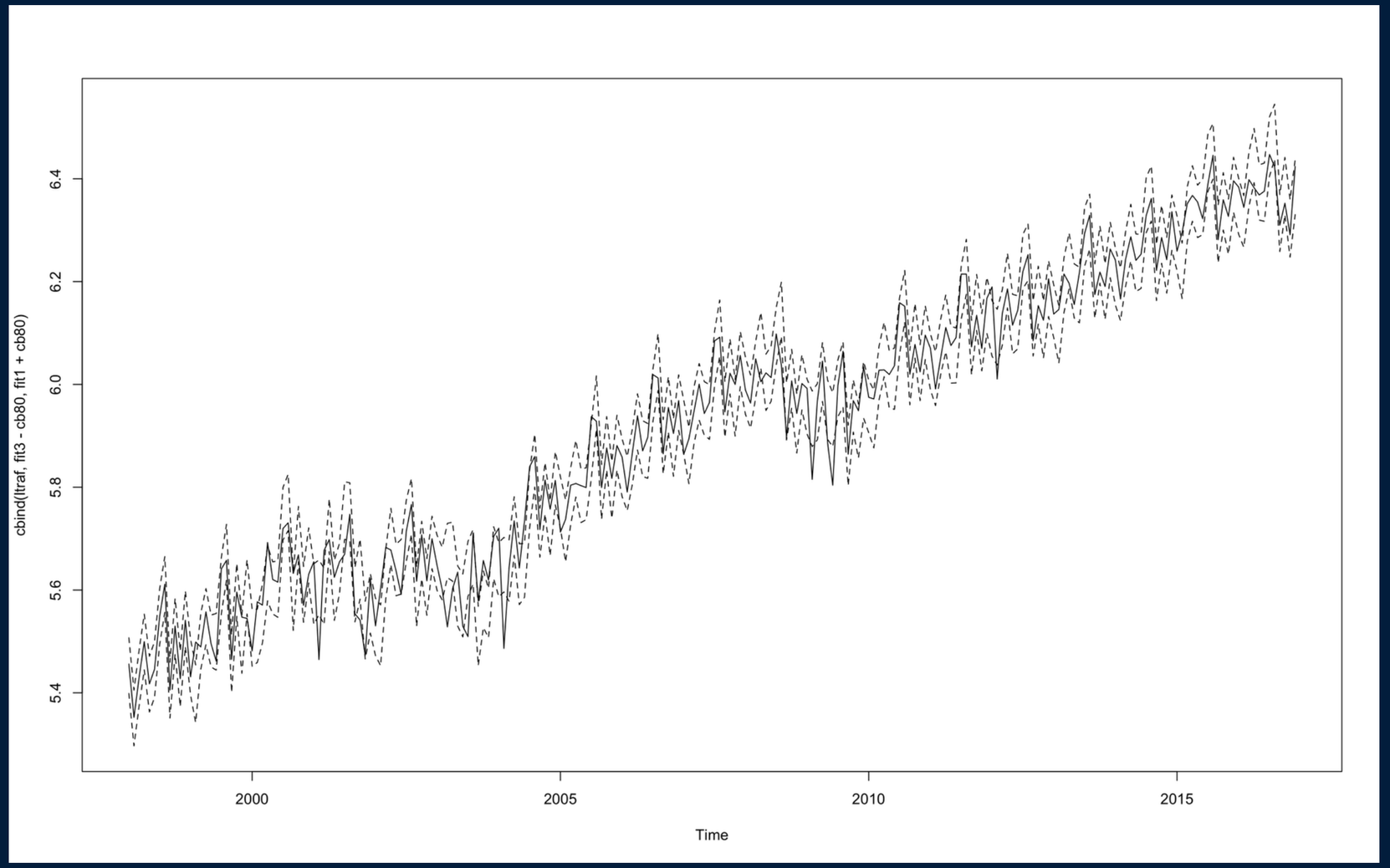
QQ Plot of Residuals:



We see:

- The distribution of residuals is very close to a Gaussian distribution
- We use this to check for normality
- The fit can be considered as good as points are close to line

Confidence bounds around the fitted value



Here we are fitting a model and with a 90% confidence interval around the fitted values.

Then we plotted these intervals along with the original data, and calculated the percentage of the original data points that fall within these intervals.

Making our first Predictive Model:

Step 1: Split Train Data & Test Data

```
data.train=window(ltraf,start=c(1998,1), end=c(2014,1))
str(data.train) #193 observations

data.test = window(ltraf, start=c(2014,2), end=c(2016,12))
str(data.test) #35 obs
mod3.train = arima(data.train, c(2,1,0), seasonal=list(order=c(0,1,1), period=12), method="ML")
```

Step 2: Run a predictive model

```
pred3.test = predict(mod3.train, n.ahead=31)
accuracy(pred3.test$pred, data.test)
```

Step 3: Results

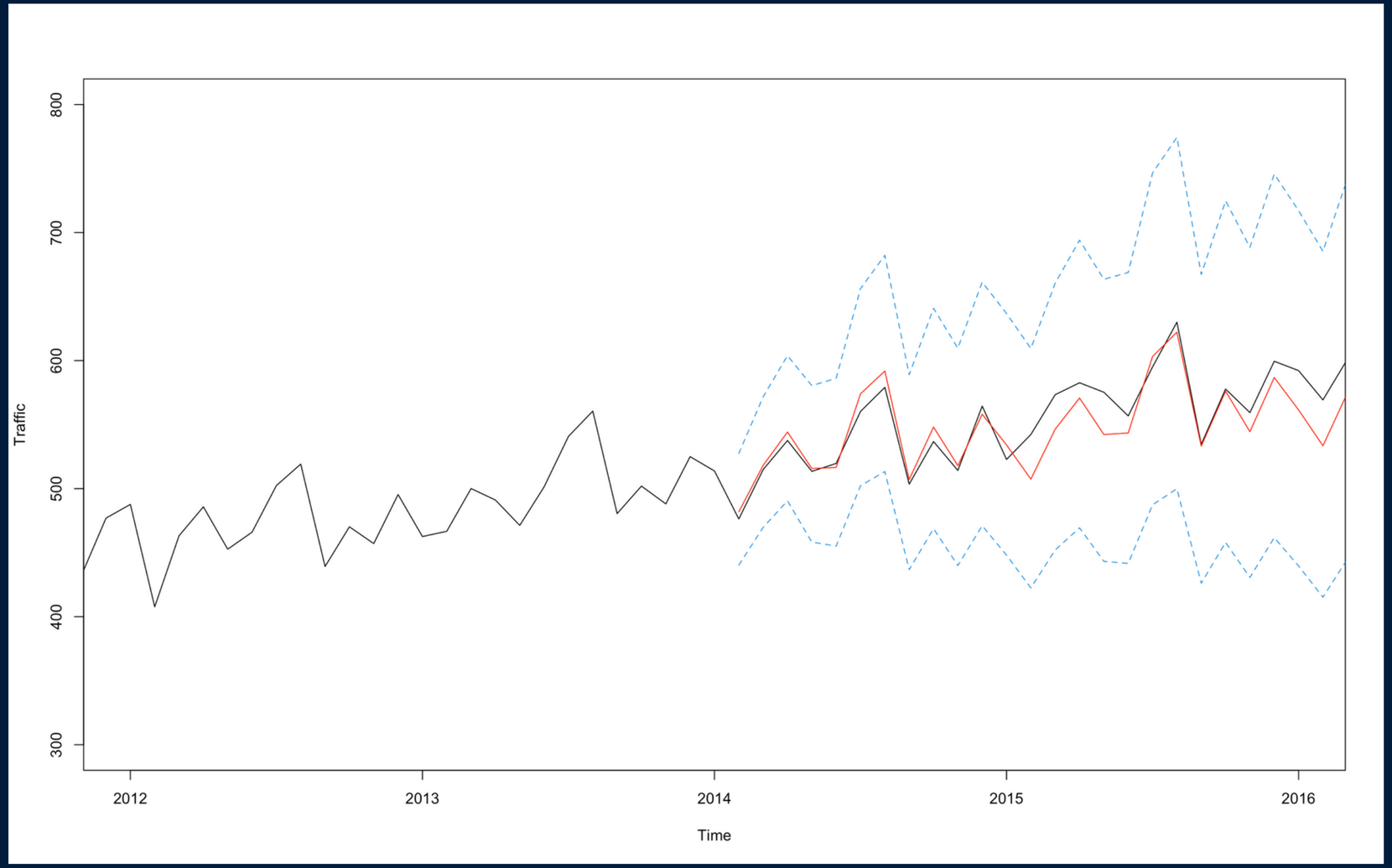
```
                    ME       RMSE        MAE       MPE      MAPE      ACF1 Theil's U
Test set 0.008558763 0.03068442 0.02401781 0.1343126 0.3784212 0.3816599 0.4902544
```

What we interpret:

- On average, we observe 38% of error with respect to the true value (MAPE result)
- The best model would be the model with the lowest value for all these parameters

Making our first Predictive Model:

Step 3: Plotting the model with In Sample Analysis



What we interpret:

- The model seems to quite closely predict what did actually happen in the test data based off what it inferred from the train data
- This was a successful start point but we try to improve regardless.

Improving our Model:

Outlier manipulation

Step 1: Outlier Identification

```
out1= which(mod3$residuals < -0.1) #identification of outlier
out1
```

Result

```
[1]   38   63   67   74 128 134 138
```

What we found:

- the outlier corresponds to the observation 50 in the dataset

Improving our Model:

Outlier manipulation

Step 2: Outlier Date Identification

```
library("zoo")
index(mod3$residuals)[out1]
```

Result

```
[1] 2001.083 2003.167 2003.500 2004.083 2008.583 2009.083 2009.417
```

What we found:

- The corresponding dates of the given observations

Improving our Model:

Outlier Manipulation

Step 3: Dummy Variable

```
HKtraf$dum1=0
HKtraf$dum1[out1]=1

mod4 =arima(ltraf, c(2,1,0), seasonal=list(order=c(0,1,1), period=12), method="ML",
            xreg = HKtraf$dum1)
mod4
```

Result

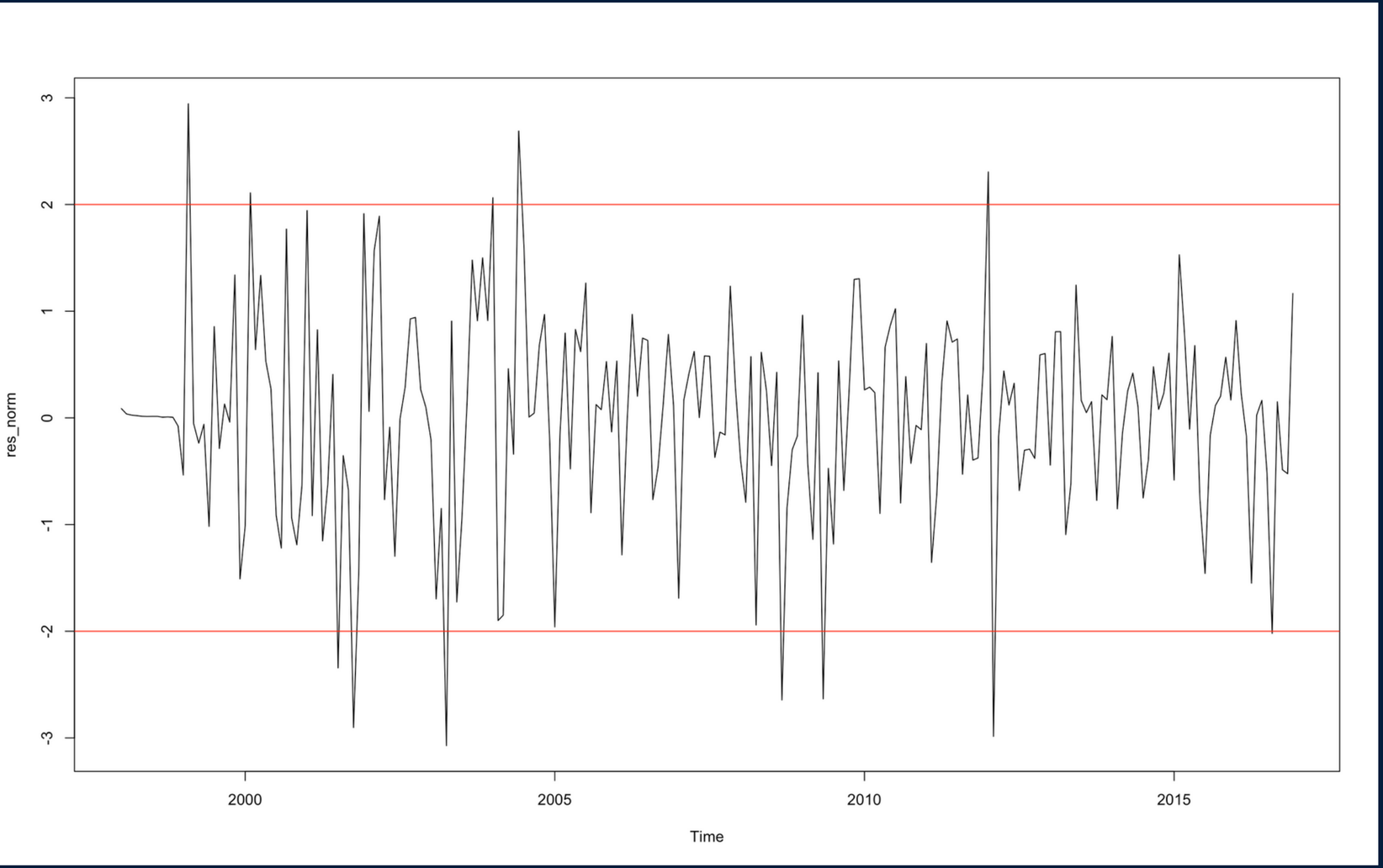
```
Call:
arima(x = ltraf, order = c(2, 1, 0), seasonal = list(order = c(0, 1, 1), period = 12),
    xreg = HKtraf$dum1, method = "ML")

Coefficients:
          ar1      ar2     sma1     xreg
      -0.6197  -0.2550  -0.9230  -0.1255
s.e.   0.0678   0.0684   0.0757   0.0126

sigma^2 estimated as 0.001294:  log likelihood = 398.46,  aic = -788.92
```

What we did:

- We added a external dummy variable X=fitting a SARIMAX model
- The graph had many peaks so we decided to isolate one
- We still got a good result
- AIC = -788.92
- This is better than before!

Improving our Model:

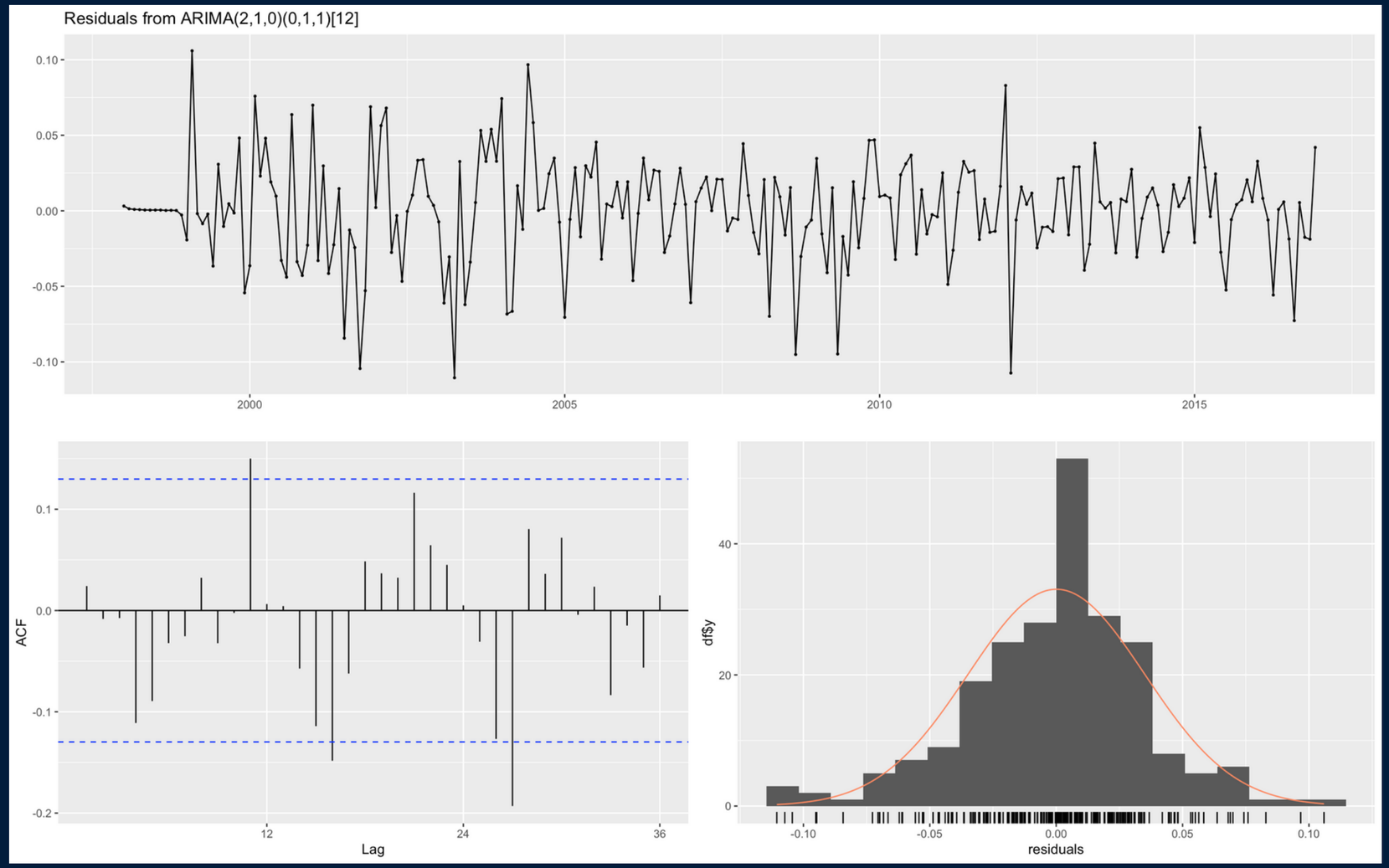
Plot the Residuals



What we interpret:

- We notice that the residuals that fall outside the red lines are much closer to the red lines than before
- We count this as a small victory

Residuals Analysis:



What we Know:

- The ACF of the residuals has no significant coefficient
- The residuals behave like a white noise
- The histogram has bell shape close to a normal distribution

New Model Predictive Analysis:

Code

```
pred4.test = predict(mod4.train, n.ahead=30, newxreg = 0)
accuracy(pred4.test$pred, data.test)
```

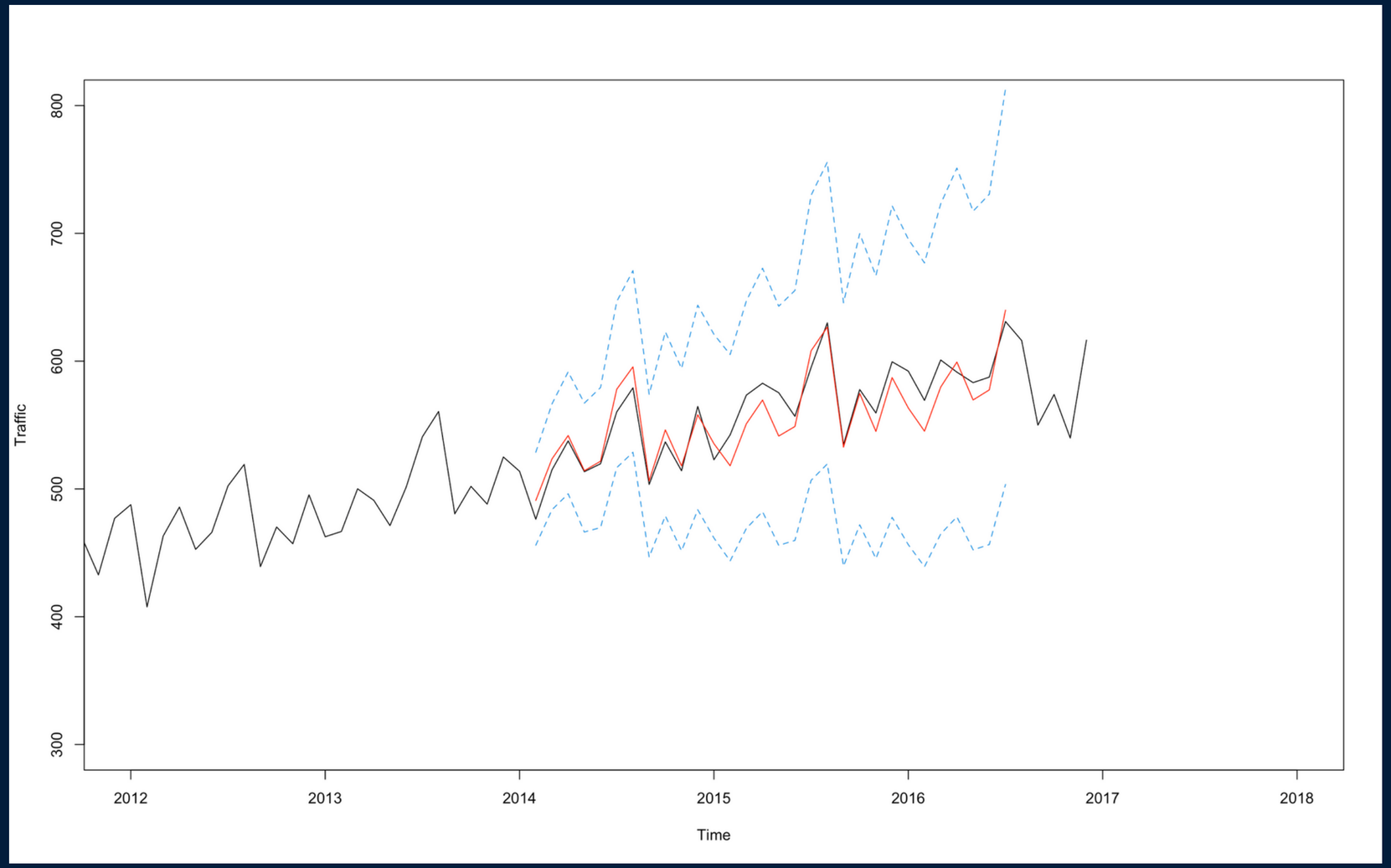
Results

```
                    ME       RMSE       MAE         MPE      MAPE       ACF1 Theil's U
Test set 0.006144721 0.02612397 0.021551 0.09560801 0.3401587 0.4265925 0.4037102
```

What we Interpret:

- MAPE = 34% instead of 38%, the predictive power of the model has improved!

New Model Plot (In Sample Analysis):



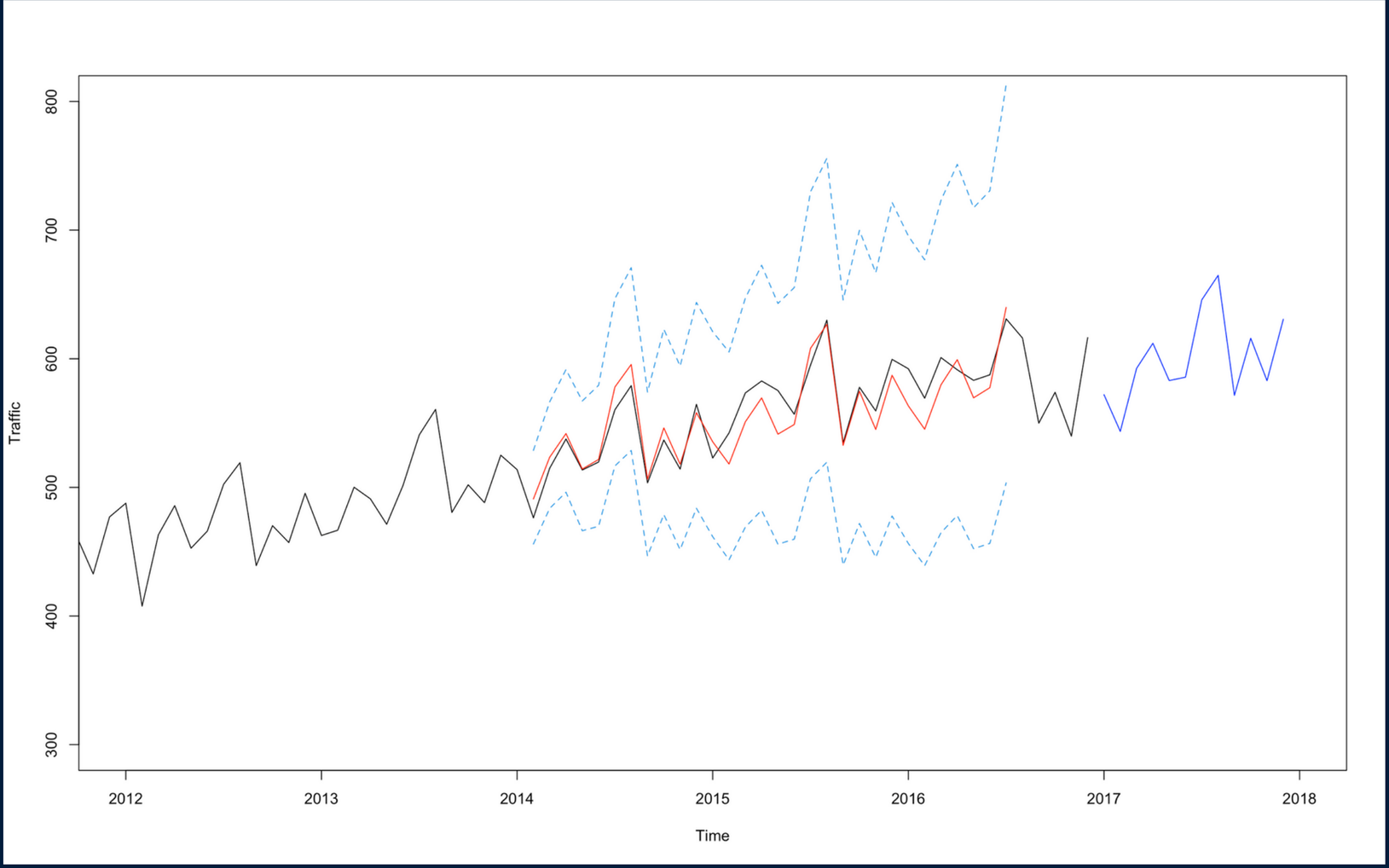
What we interpret:

- The model very closely predicts what did actually happen in the test data based off what it inferred from the train data
- This is a big improvement (visually) from before
- We now go on to use this model to predict past the test data and into the future

Out of Sample Analysis

Predicting the next 3 periods:

Graph 1



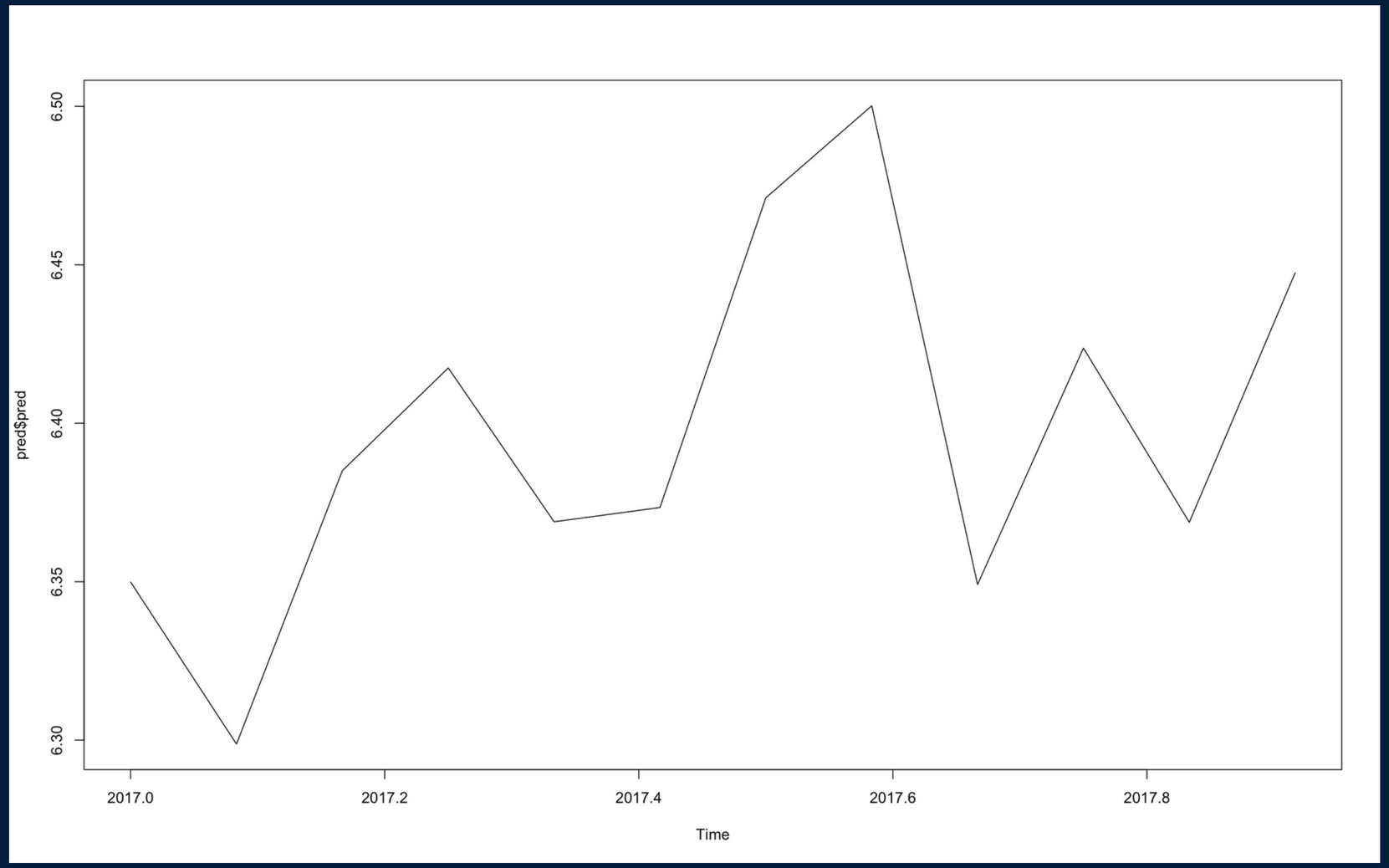
What we found:

- The model shows us the prediction based off our test and training data, which we can see the blue line on the graph
- The graph predicts Hong Kong's airport traffic for the next 10 periods
- The blue line extends beyond our data which ends in December 2016

Out of Sample Analysis

Predicting the next 3 periods:

Graph 2



- Graph 2 represent a closer look based on Graph 1 on the previous slide.
- By looking at the pattern, we can say that overall, the prediction will contain a trend and also peak moments such as 2017.2 while also have some drops like after 2017.6

THANK YOU

FOR YOUR ATTENTION