[1]

PLDT-Smart Broadband Traffic Prediction using Auto ARIMA Model

author : Nicole John Mortel
e-mail: 22-09584@g.batstate-u.edu.ph
namortel@pldt.com.ph
njamortel@gmail.com

Batangas State University
The National Engineering University

Abstract— This study aims to develop a model to predict broadband traffic of PLDT-Smart, Internet Service Providers (ISP) in the Philippines. By leveraging data science approaches, using the ARIMA model to predict broadband traffic and use it as reference for the analysis of network performance and provide warnings to optimize network infrastructure, allocate bandwidth, and provide quality of service and customer experience.

Keywords – Telecom Broadband Network Performance, Internet Traffic, Forecasting

I. INTRODUCTION

In the Philippines, ISP like PLDT-Smart face challenges in managing network infrastructure due to processing big data and unpredictable broadband traffic.

This study proposes a solution to this problem by developing a predictive model that can predict broadband traffic using the ARIMA model. This will provide a comprehensive analysis of broadband traffic, which is vital for understanding and improving the telecom infrastructure. By examining traffic and usage, we could be able assess how efficiently network resources are being used.

Through this analysis, we aim to predict the broadband traffic that can inform strategies to enhance network performance and service quality. The findings of this research will be particularly relevant for telecom operators, policymakers, and consumers who are navigating the challenges and opportunities presented by the evolving telecom landscape in the Philippines.

II. RESEARCH METHODOLOGY

The methodology used in this analysis is the Agile Method in Data Analytics Life Cycle. Data Collection, Data Preprocessing, Data Modeling, Data Models Evaluation and Validation, and Communicate Results and Insights.

Agile Methodology

Figure 1. Agile Method in Data Analytics Life Cycle

III. DATA PREPARATION

In the Agile approach of data analysis (see in figure 1), data gathering, and EDA is part of data preparation for analysis.

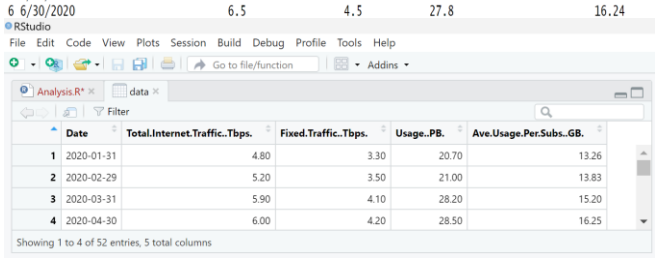
Data Gathering: Gather the needed data requirements for the analysis from the internal partners or stakeholders. The datasets used for this analysis are big and need to process that big data with the use of the tools and Network Management System (NMS). We need to extract the needed data features from the data source:

+ Date
+ Traffic

Data Source

The datasets used in this research are from internal systems Network Management System under Performance Module. The data set needs to be stored in a .csv file by extracting it from these data sources for preparing the data features.

```
> # Load necessary libraries
> library(ggplot2)
> library(forecast)
> library(lubridate)
> library(imputeTS)
> # Load the dataset
> data <- read.csv('C:/Users/njamo/OneDrive - PLDT/Study/BatState-U/MSDS/MSDS 511 - Time Series Data Analysis
and Forcasting/Broadband Traffic Prediction/Traffic and Usage.csv')
> head(data)
      Date Total_Internet_Traffic.Tbps Fixed_Traffic.Tbps Fixed_Usage.PB Fixed_Ave_Usage_Per_Subs.GB
1 1/31/2020                      4.8                3.3           20.7                      13.26
2 2/29/2020                      5.2                3.5           21.0                      13.83
3 3/31/2020                      5.9                4.1           28.2                      15.20
4 4/30/2020                      6.0                4.2           28.5                      16.25
5 5/31/2020                      6.1                4.3           27.8                      16.25
6 6/30/2020                      6.5                4.5           27.8                      16.24
```



Data Cleaning and Transformation

After loading the data we need to check if there are missing values. There are null values or missing values in the data we collected.

```
> # Check for NA values in 'Total.Internet.Traffic.Tbps' column
> sum(is.na(data$Total_Internet_Traffic.Tbps))
[1] 0
> sum(is.na(data$Fixed_Traffic.Tbps))
[1] 0
> sum(is.na(data$Usage.PB))
[1] 1
> sum(is.na(data$Ave_Usage_Per_Subs.GB))
[1] 1
```

We can remove the missing data but since this is a time series and we are predicting then we will interpolate to impute the missing values.

```
> # Interpolate the missing data
> data$Usage.PB <- na_interpolation(data$Usage.PB)
> data$Ave_Usage_Per_Subs.GB <- na_interpolation(data$Ave_Usage_Per_Subs.GB)
> # Recheck missing value if still null after imputation
> sum(is.na(data$Usage.PB))
[1] 0
> sum(is.na(data$Ave_Usage_Per_Subs.GB))
[1] 0
```

The date needs to be transformed to a date object and set as a time series object.

```
> # Convert 'Date' to a date object and set it as a ts object
> data$Date <- mdy(data$Date)
> View(data)
```

The values parameters are also need to be transformed to as a time series object.

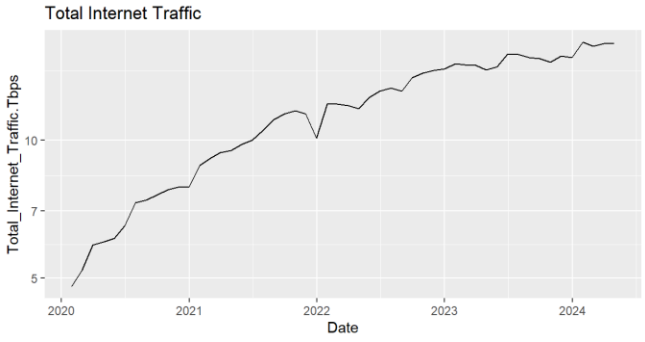
```
> # Create time series objects
> total_traffic_ts <- ts(data$"Total_Internet_Traffic.Tbps", start=c(2020,1), frequency=12)
> fixed_traffic_ts <- ts(data$"Fixed_Traffic.Tbps", start=c(2020,1), frequency=12)
> fixed_usage_ts <- ts(data$"Fixed_Usage.PB", start=c(2020,1), frequency=12)
> fixed_ave_usage_per_subs_ts <- ts(data$"Fixed_Ave_Usage_Per_Subs.GB", start=c(2020,1), frequency=12)
```

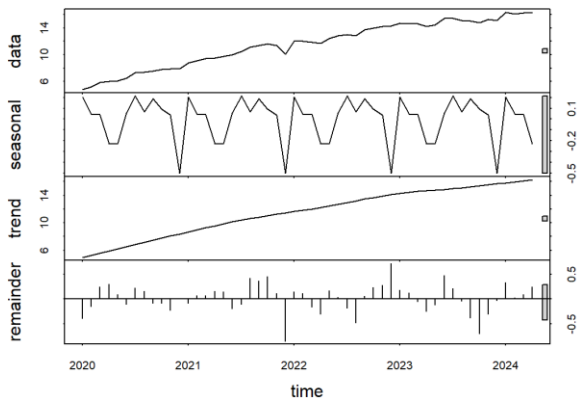
Exploration and Data Analysis (EDA): Explore the structure and summary of the dataset.

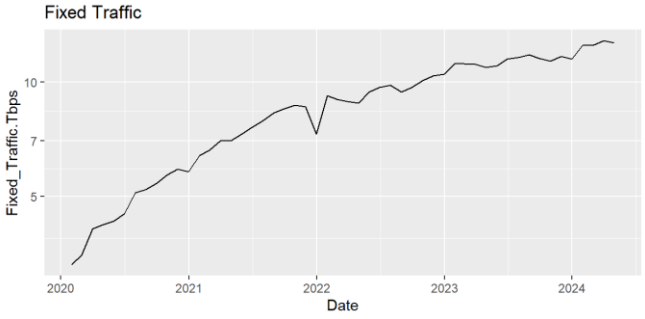
```
> # Explore the structure and summary of the dataset
> str(data)
'data.frame':   52 obs. of  5 variables:
 $ Date                       : Date, format: "2020-01-31" "2020-02-29" ...
 $ Total_Internet_Traffic.Tbps: num  4.8 5.2 5.9 6 6.1 6.5 7.3 7.4 7.6 7.8
...
 $ Fixed_Traffic.Tbps         : num  3.3 3.5 4.1 4.2 4.3 4.5 5.1 5.2 5.4 5.7
...
 $ Usage.PB                   : num  20.7 21 28.2 28.5 27.8 27.8 30.7 32.9 3
3.4 34.3 ...
 $ Ave_Usage_Per_Subs.GB      : num  13.3 13.8 15.2 16.2 16.2 ...
```

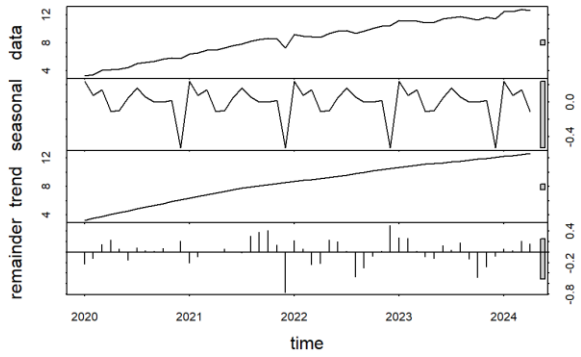
```
> summary(data)
      Date            Total_Internet_Traffic.Tbps Fixed_Traffic.Tbps
 Min.   :2020-01-31   Min.   : 4.800              Min.   : 3.300
 1st Qu.:2021-02-21   1st Qu.: 9.025              1st Qu.: 6.550
 Median :2022-03-15   Median :11.950              Median : 8.950
 Mean   :2022-03-16   Mean   :11.568              Mean   : 8.658
 3rd Qu.:2023-04-07   3rd Qu.:14.588              3rd Qu.:11.165
 Max.   :2024-04-30   Max.   :16.350              Max.   :12.860
    Usage.PB     Ave_Usage_Per_Subs.GB
 Min.   :20.70   Min.   :13.26
 1st Qu.:40.85   1st Qu.:18.98
 Median :61.25   Median :20.41
 Mean   :56.22   Mean   :20.09
 3rd Qu.:71.04   3rd Qu.:21.58
 Max.   :83.13   Max.   :24.39
```

Traffic and usage trends will help discern long-term patterns and fluctuations. This visualization facilitates a comprehensive understanding of the historical evolution of traffic, aiding in strategic analysis and decision-making.









One of the key features noticed is a sudden drop in traffic around the last month of 2021, which coincides with a typhoon. After the typhoon, we can see that the traffic increased. This visual representation facilitates a comprehensive understanding of the historical evolution of traffic, aiding in strategic analysis and decision-making.

Here I decompose a series of data points to determine the seasonality of trend. It shows the trend, seasonality and the remainder or the random/irregular component or residual of a time series.

IV. DATA MODELING AND VALIDATION OF RESULTS

Here I split my dataset into a training set and validation set to fit in my model. I fitted my data model using the auto ARIMA model. with the order (0,1,1) (0,0,1)[12] with drift.

```
> # Create a training set and a validation set
> train <- window(total_traffic_ts, end = c(2023,12))
> validation <- window(total_traffic_ts, start = c(2024,1))

> # Fit an ARIMA model
> model <- auto.arima(train)

> # Print the model summary
> summary(model)
Series: train
ARIMA(0,1,1)(0,0,1)[12] with drift

Coefficients:
          ma1    sma1   drift
      -0.3408  0.2738  0.2157
s.e.   0.1655  0.1576  0.0495

sigma^2 = 0.189:  log likelihood = -26.52
AIC=61.04   AICc=61.99   BIC=68.44

Training set error measures:
                     ME      RMSE       MAE        MPE
Training set -0.006434113 0.4162605 0.3001675 0.2132041
                  MAPE      MASE       ACF1
Training set 2.745736 0.1099291 0.04682534
```

ARIMA (0,1,1) - is the non-seasonal part of my model. The 3 numbers correspond to the AR (autoregressive) order, differencing (integrated) order, and the MA (moving average) order. in this case, i have no AR, I have differenced the series once to make it stationary, and I have 1 MA term.

the seasonal part of my model (0,0,1)[12]. Again the 3 numbers also correspond to AR, differencing/ integrated, and the MA order. In this case I have no seasonal AR terms, no seasonal differencing, and 1 seasonal MA term. the number in brackets 12 indicates that the seasonality is annual (12 months).

with drift - this indicates that a constant term is included in the model.

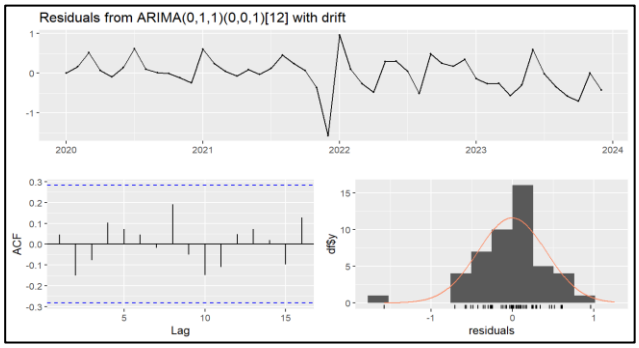
The coefficients section provides the estimated values for the coefficients of the MA1 term, the seasonal MA1 term, and the drift term, along with their standard errors (s.e.).

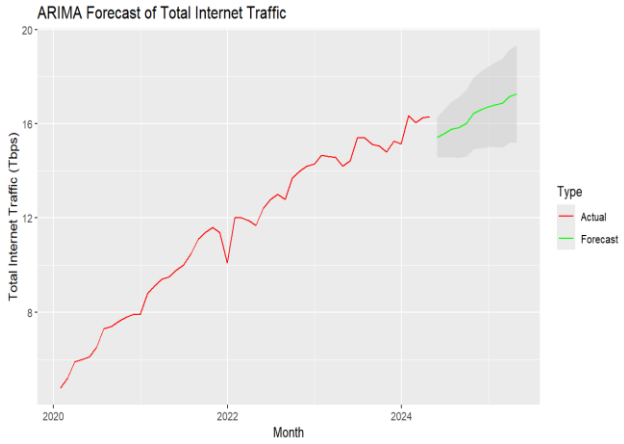
The sigma^2 value is the estimated variance of the residuals in the model.

The log likelihood, AIC, AICc, and BIC are all measures of the goodness-of-fit of the model. The log-likelihood is the logarithm of the likelihood function, which measures how likely it is to observe the data given the model. The AIC (Akaike Information Criterion), AICc (corrected AIC), and BIC (Bayesian Information Criterion) all penalize models for complexity (i.e., having more parameters), so lower values are generally better.

The training set error measures section provides various error measures calculated on the training set, including the mean error (ME), root mean squared error (RMSE), mean absolute error (MAE), mean percentage error (MPE), mean absolute percentage error (MAPE), mean absolute scaled error (MASE), and the autocorrelation of the first lag of residuals (ACF1).

Visualization and Analysis of Results: I used the check residual function to test my model, if the residuals from the ARIMA model are independently distributed.





VI. CONCLUSIONS AND RECOMMENDATIONS

The output of the test shows a p-value of 0.6018 which is greater than 0.05. This is a good sign as it suggests that the model has captured the underlying correlations in the data.

Error Metrics: The Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) are calculated to evaluate the accuracy of the forecasts.

The MAE, RMSE, and MAPE are all measures of the average error in the forecasts. Lower values indicate more accurate forecasts.

In your case, the MAE is 0.5775902, the RMSE is 0.6106889, and the MAPE is 0.03553176. These values can be used to compare the accuracy of this model with other models.