Tracking the Pulse of GeoHub LA: Insights from Web Traffic Data

Reshma RB, M.Sc. Applied Statistics, Department of Statistics, Kerala University

Neethu Neelambaran R, M.Sc. Statistics, Department of Statistics, Kerala University

Sivapriya S, M.Sc. Applied Statistics, Department of Statistics, Kerala University

ABSTRACT:

GeoHub LA is a data-sharing platform created by the City of Los Angeles to provide access to spatial data for city officials, community organizations, and the general public. As the platform's popularity grows, it becomes increasingly important to analyze web traffic data to identify patterns, trends, and anomalies in user behavior. In this project, we will perform a time series analysis of web traffic data for GeoHub LA using Python. We will extract data from web logs, preprocess and clean the data, and then use time series analysis techniques such as decomposition, forecasting, and anomaly detection to gain insights into user behavior. We will also explore how different factors such as time of day, day of the week, and seasonal trends affect web traffic. The results of this analysis can inform decisions on website optimization, resource allocation, and user engagement strategies for GeoHub LA.

INTRODUCTION

GeoHub LA is a data-sharing platform created by the City of Los Angeles to provide access to spatial data for city officials, community organizations, and the general public. As the platform's popularity grows, it becomes increasingly important to analyse web traffic data to identify patterns, trends, and anomalies in user behaviour. In this project, we will perform a time series analysis of web traffic data for GeoHub LA using Python.

Time series forecasting is a process of predicting future values of a time series based on past results. ARIMA (Autoregressive Integrated Moving Average) is a popular time series forecasting algorithm that can be used to model any non-seasonal time series exhibiting patterns. ARIMA models are specified by three order parameters: (p, d, q), where p is the order of the AR term, q is the order of the MA term, and d is the number of differencing required to make the time series stationary.

Web traffic forecasting is crucial for any website as it can cause setbacks and inconvenience for users if the website crashes or has slow loading times due to high traffic. The ability to predict future web traffic and analyse real-time data through a dashboard can inform decisions on website optimization, resource allocation, and user engagement strategies. In this project, we will explore how time of day, day of the week, and seasonal trends affect web traffic for GeoHub LA, and use ARIMA models for forecasting and anomaly detection to gain insights into user behaviour. The results of this analysis can help improve the performance and reliability of GeoHub LA, and inform the development of traffic management techniques for other websites.

ARIMA, short for Auto Regressive Integrated Moving Average, is a forecasting algorithm based on the idea that the information in the past values of the time series can alone be used to predict the future values.

ARIMA Models are specified by three order parameters: (p, d, q),

Where,

- p is the order of the AR term
- q is the order of the MA term
- d is the number of differencing required to make the time series stationary

AR(p) Autoregression – a regression model that utilizes the dependent relationship between a current observation and observations over a previous period. An auto regressive (AR(p)) component refers to the use of past values in the regression equation of the time series.

The term “I” - stands for integrated, which means that the data is stationary. Stationary data refers to time-series data that's been made “stationary” by subtracting the observations from the previous values.

MA(q) model – a model is used for forecasting future values, while moving average smoothing is used for estimating the trend-cycle of past values. MA model can be updated every time a new forecast is made; that is, you add in the forecast errors and update the model.

LITERATURE REVIEW

Here, we explain the existing technology in the field of web traffic time series forecasting and also the data set that has been used for our prediction model. Web Traffic Time Series dataset is from Geohub LA which is a data-sharing platform created by the City of Los Angeles to provide access to spatial data for city officials, community organizations.

Here we use ARIMA model. ARIMA (Auto-Regressive Integrated Moving Average) the model has a huge advantage in univariate time series forecasting. ARIMA model attempts to describe the trends and seasonality in time series as a function of lagged values(Auto Regressive parameter) and Averages changing over time intervals(Moving Averages). The model includes differencing (Integrating) the original time series data. Differencing time-series means forming a new time series by subtracting the previous observation from the current time. The point of this is to remove certain trends, such as seasonality, trends, or inconsistent variance in time series data. The ARIMA equation has two important components Auto-Regressive (AR) part and the Moving Average (MA) part.

AR(p) Autoregression – a regression model that utilizes the dependent relationship between a current observation and observations over a previous period. An auto regressive (AR(p)) component refers to the use of past values in the regression equation of the time series.

The term “I” - stands for integrated, which means that the data is stationary. Stationary data refers to time-series data that's been made “stationary” by subtracting the observations from the previous values.

MA(q) model – a model is used for forecasting future values, while moving average smoothing is used for estimating the trend-cycle of past values. MA model can be updated every time a new forecast is made; that is, you add in the forecast errors and update the model.

RELATED WORK

Web traffic forecasting is a crucial task that can have a significant impact on website functionality and user satisfaction. The following papers present various approaches to forecasting web traffic using time series analysis:

In "Web Traffic Time Series Forecasting using ARIMA and LSTM RNN" by Shelatkar et al., the authors propose using a combination of Long Short-Term Memory (LSTM) Recurrent

Neural Networks and Autoregressive Integrated Moving Average (ARIMA) models for efficient and accurate web traffic forecasting. The authors demonstrate that their system effectively captures seasonal patterns and long-term trends and suggest that including information about holidays, day of week, language, and region might improve the model's performance. The authors also suggest that exploring multivariate time series could offer suggestions for simplifying the decision-making process in real-time.

In "Forecast Web Traffic Time Series Using ARIMA Model" by Tambe et al., the authors propose a forecasting model that uses the ARIMA model to forecast web traffic time series. The authors use data such as the name of the page, the date it was seen, and the number of visits to make more accurate predictions. The authors show that their ARIMA-based model can reliably predict future traffic to Wikipedia pages and suggest that they will look into unsupervised models to improve their model's ability to spot hidden trends and investigate how human behaviour influences online traffic more quickly.

In "Web Traffic Time Series Prediction Using ARIMA & LSTM" by Shao et al., the authors compare the performance of ARIMA and LSTM models for web traffic time series prediction. The authors suggest that predicting time series is a hard problem due to various circumstances behind time series. To address this issue, the authors propose removing outliers before training the model to ensure that outliers do not have an overall impact on the predictions.

In "Forecasting Traffic Congestion Using ARIMA Modelling," the authors propose an ARIMA-based short-term time series model for non-Gaussian traffic data to better manage traffic congestion by capturing and predicting any abnormal status. The authors highlight the characteristics and structure of the dataset that negatively impact the performance of time series analysis and use R to pre-process and prepare the dataset for the modelling phase.

In conclusion, these papers demonstrate the effectiveness of time series analysis techniques, such as ARIMA and LSTM, for web traffic forecasting and traffic congestion management. They suggest that including additional data and exploring multivariate time series could improve model performance, and removing outliers before training the model could improve model accuracy.

METHODOLOGY

ARIMA models use differencing to convert a non-stationary time series into a stationary one, and then predict future values from historical data. These models use “auto” correlations and moving averages over residual errors in the data to forecast future values.

ARIMA models are the well-known of models for time series forecasting. Box and Jenkins were the ones who first brought it up(1970).(p,d,q) is the general ARIMA model, were p denotes the autoregressive parameters ,d the number of differencing operators , and q the moving average parameter.

ARIMA forecasts temporal dependencies using only historical values. The data used for Autoregressive models are prepared differently from LSTM. In addition to necessary pre-processing steps, the AR model’s data needs to be stationary. Simply put, data is stationary when its numeric properties do not change over time. From a mathematical perspective, it refers to the data whose Mean and Variance will not depend on time. If the data doesn’t meet the properties of the stationary dataset, you can do a series transformation to make it stationary. ARIMA model is a combination of two models Ꞓset stationary. In the AR part of the model future values are predicted using the lags from the data values. The general equation AR model is:

$$\text{AR(p)}.x_t = \alpha + \beta_i x_{t-i} + \mathcal{E}$$

The Moving Average is the part of the model where value is forecasted using the forecasting error differences is calculated while making predictions. The general form of MA equation is

$$x_{t=}\mu+\sum_{i=1}^{q} \phi_{i\varepsilon_{t-1}}$$

Prediction is done by combining all three orders and getting an estimation of how to quickly fit the model. Some standard denotations are used to represent the above three, i.e. p, d, and q; p is the number of observations included in the model, d is the number of times differentiating the raw observations, q is the number of moving average size. To find these parameters, first fetch a Correlation and Partial Correlational graph from the dataset.

We know the ARMA process is

$$\phi_{(B)Z\,t} = \Theta(\text{B})\, a_t \quad \text{…………} [1]$$

With ϕ(B) and ϴ(B)are polynomials on B with degree p and q respectively. To ensure the stationarity condition we have the root of ϕ(B)=0 must lie outside the unit circle. Now we consider the AR(1) process

$$(\,1-\phi_1\,(\text{B}))Z_t = a_t \text{ ……………} [2]$$

Which is stationary for $|\phi_1|$≤1

Let us study the behaviour of the process for ϕ=2 a value outside the stationary range. Let us consider the model

$$\Psi(\text{B})\; Z_t = \Theta(\text{B}) a_t \text{ ……………..} [3]$$

Where Ψ(B) is non stationary AR operator such that 'd' of the roots of Ψ(B)=0 are unity and the reminder be outside the unit circle. Now we can write the model given in [1] in the form

$$\Psi(\text{B})Z_t = \phi(\text{B})(1-B)^d Z_t$$

$$= \Theta(\text{B})a_t \text{ …………….} [4]$$

Where ϕ(B) is stationary autoregressive operator. Since $\nabla^d Z_t$ = $\nabla^d Z_t$ for d > 1 where ∇ = 1-B is the difference operator .Now we can write above equation [4] as

$$\Phi(\text{B})\nabla^d Z_t = \Theta(\text{B})a_t \text{ …………..} [5]$$

Equivalently ,the process is defined by [2] equations

$$\Phi(\text{B})w_t = \Theta(\text{B})a_t \text{ …………..} [6]$$

$$w_t = \nabla^d Z_t \text{ ……………..} [7]$$

Thus we can see that the model corresponds to assuming that the d^{th} difference of the series can be represented by stationary invertible autoregressive moving average process. Alternatively , for d > 1 inverting [7] we yield

$$Z_t = S^d w_t \text{ ………………} [8]$$

Where S is the infinite summation operator defined by

$$S_{xt} = \sum_{n=-}^{t} x_n$$

$$= (1+B+B^2+\dots\dots\dots)\, x_t$$

$$= (1-B)^{-1}\ x_t$$

$$= \nabla^{-1} x_t$$

Thus $\qquad$ S = $(1-B)^{-1}$ = ∇^{-1}

Similarly we can define $S^2, S^3,\dots\dots\dots\dots, S^{\,m}$

Equation [8] implies that the process [5] can be obtained by summing or integrating the stationary process given [6] d times. We call the process as an autoregressive integrated moving average model (ARIMA) .

Technically the infinite summation operator S = $(1-B)^{-1}$ can't be used in defining the non-stationary ARIMA process, since the infinite summation operator S=$(1-B)^{-1}$ can't be used in defining the non-stationary ARIMA process, since the infinite summation involved will not be convergent.

Therefore, we can consider the finite summation of S_m for +ve integer m given by,

$$S_m = (1+B+B^2+\dots\dots\dots\dots+B^{m-1})$$

$$= \frac{1-B^m}{1-B}$$

The equation [5] the autoregressive operator ϕ(B) is of order p, the d^{th} difference is taken and the moving average Θ(B) is order q. Now we can say that we have an ARIMA process of order (p,d,q) or simply ARIMA (p,d,q) process.

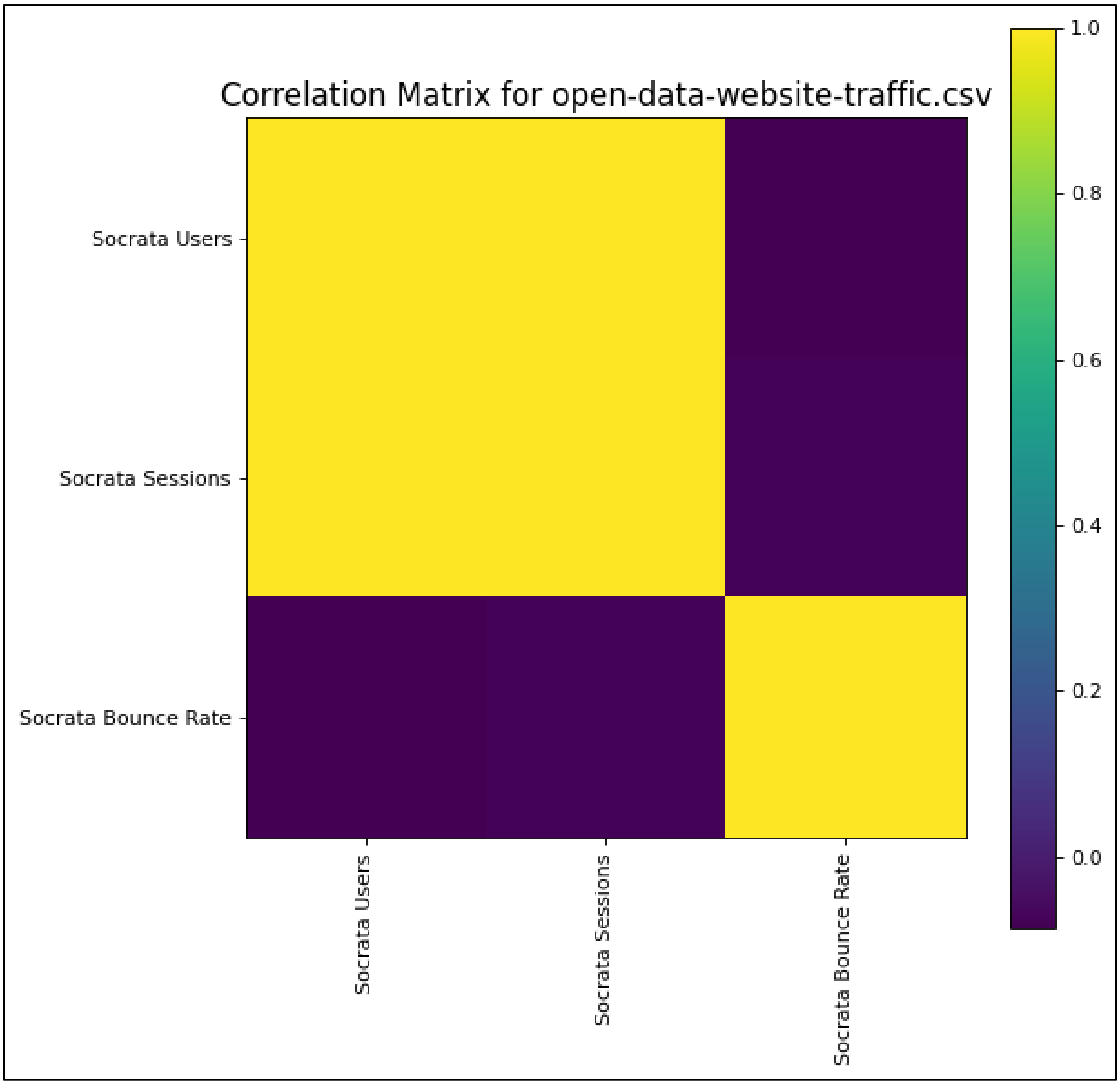
IMPLEMENTATION AND RESULTS

1. Dataset

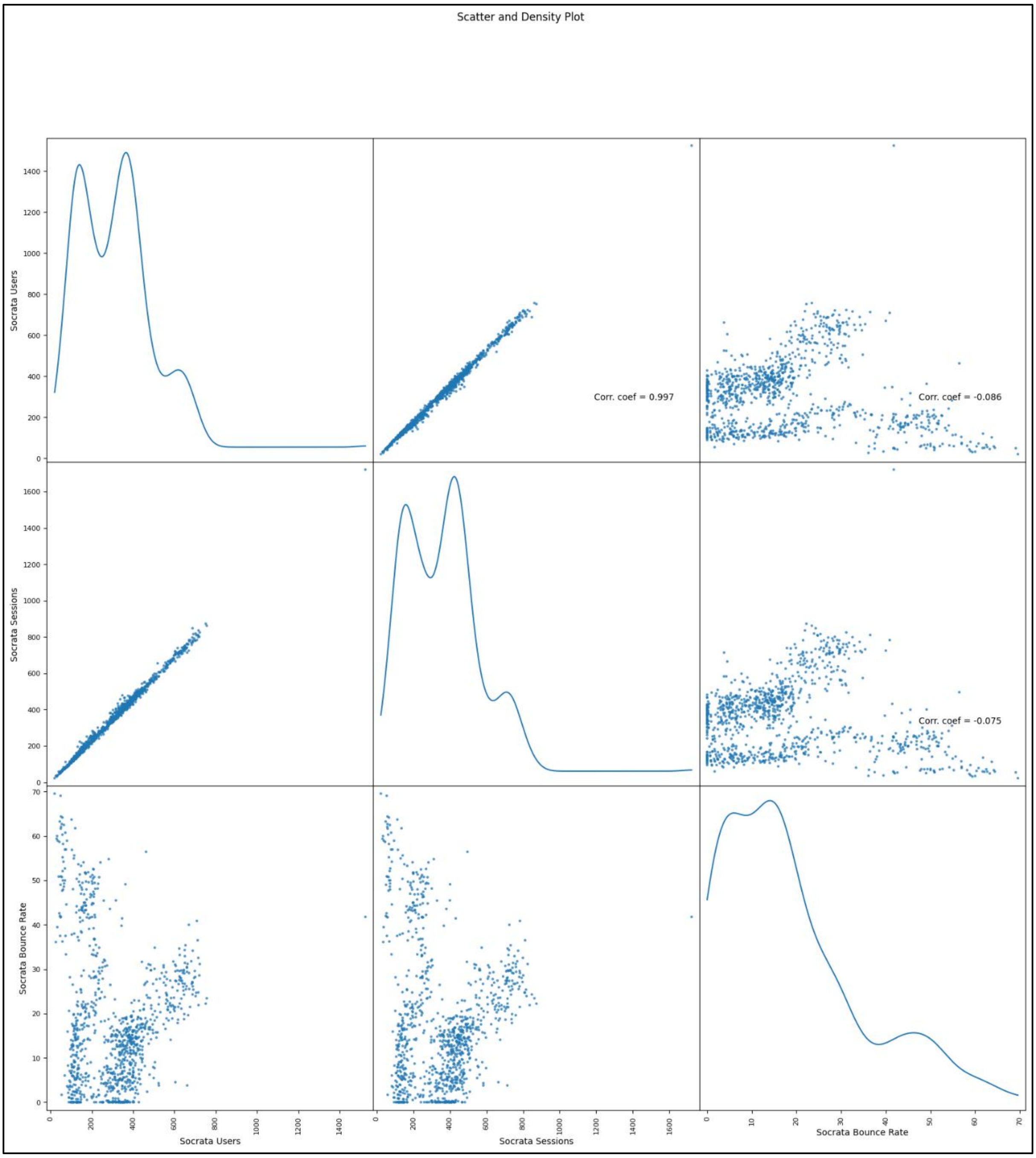
Our dataset is from Geohub LA which is a data-sharing platform created by the City of Los Angeles to provide access to spatial data for city officials, community organizations, and the general public. As the platform's popularity grows, it becomes increasingly important to analyze web traffic data to identify patterns, trends, and anomalies in user behavior.The data is given below,

2. Exploratory Data Analysis

The correlation matrices for open data website is given below.



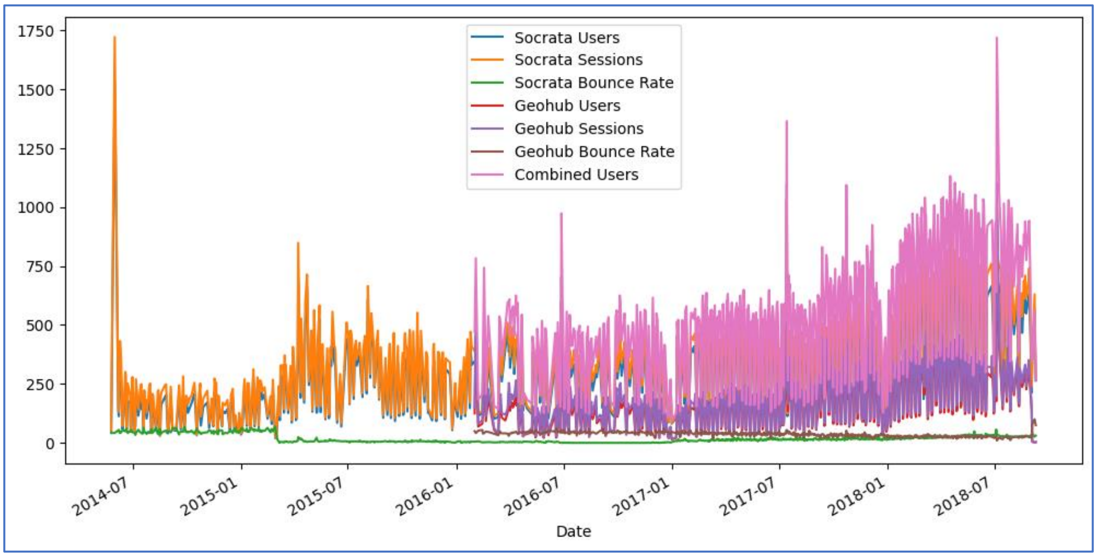
The scatter and density plots are given below. It represents values for two different numeric variables. The position of each dot on the horizontal and vertical axis indicates values for an individual data point. Scatter plots are used to observe relationships between variables.



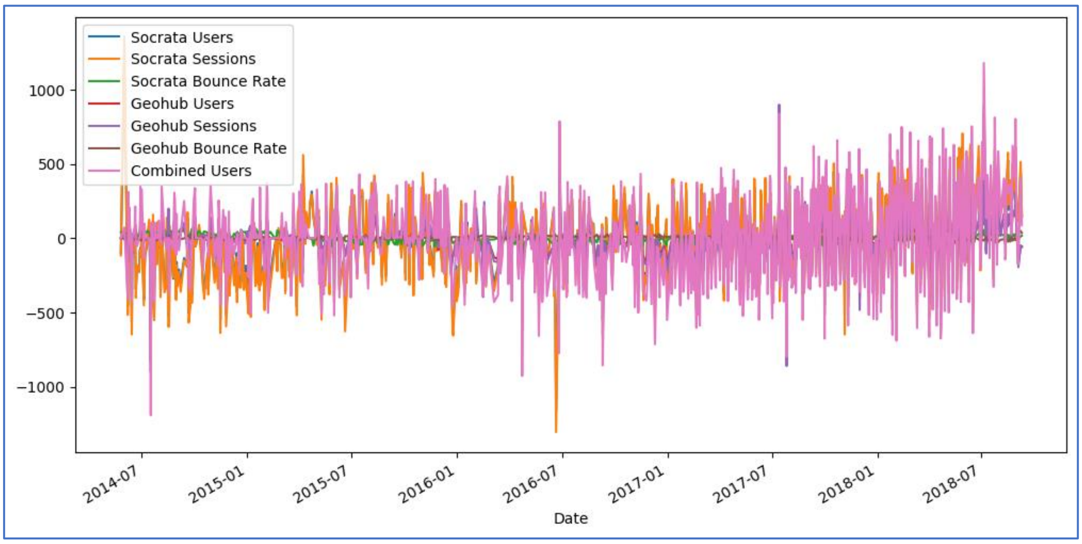
3. Data Visualization

Data visualization is the practice of designing and creating easy-to-communicate and easy-to-understand graphic or visual representation of a large amount of complex quantitative and qualitative data and information from a certain domain of expertise with the help of static, dynamic or interactive visual items for a broader audience to help them visually explore and discover, quickly understand, interpret and gain important insights into otherwise difficult-to-identify structures, relationships, correlations, local and global patterns, trends, variations, constancy, clusters, outliers and unusual groupings within data. When intended for the general public to convey a concise version of known, specific information in a clear and engaging manner (presentational or explanatory visualization), it is typically called information graphics.

It's always a good idea to visualize the data to get an idea about the trends and patterns in the data:

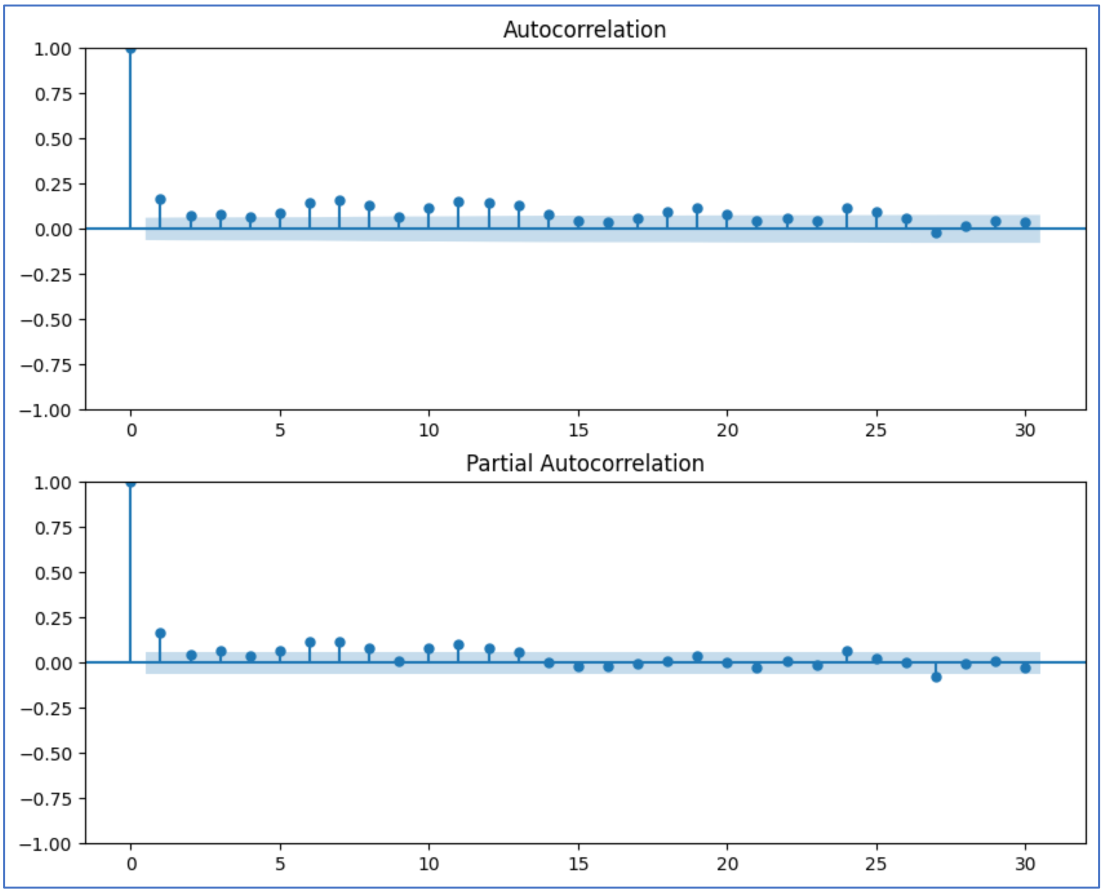


If the time series is not stationary, we need to make it stationary before proceeding. We can use differencing method to make the time series stationary and the result will be,



The autocorrelation function (ACF) and partial autocorrelation function (PACF) to get an idea about the lags and order of the ARIMA

To Check the stationarity of the differenced time series data we can use the ACF and PACF plots.



Now the time series becomes stationary**.**

4. Time Series forecasting

Building different ARIMA models with different orders and compare their performance using AIC, BIC, and RMSE:

p	d	q	AIC	BIC	RMSE
2	1	3	12643.01	12672.45	18267.88
3	0	3	12662.52	12701.78	17916.78
3	1	3	12862.11	12896.46	22740.86
3	1	1	12929.85	12954.38	24411.68
3	1	2	12943.58	12973.03	24696.38
2	1	1	12947.13	12966.76	24883.28
1	1	3	12956.05	12980.59	25058.36
0	1	2	12186.92	12001.64	11538.23
0	1	3	12988.78	13008.41	25934.46
1	1	2	12988.83	13008.46	25935.89
1	0	2	13002.58	13027.12	25664.33
1	0	3	13004.43	13033.88	25660.11
2	0	2	13004.49	13033.93	25661.74
2	0	3	13004.9	13039.25	25583.13
1	1	1	13082.3	13097.02	28514.14

3	0	2	13129.86	13164.21	29067.1
0	0	3	13129.87	13154.41	29185.54
3	0	1	13130.41	13159.86	29141.95
2	0	1	13130.59	13155.13	29205.88
1	0	1	13131.26	13150.9	29284.78
3	0	0	13135.16	13159.69	29341.27
0	0	2	13138.5	13158.13	29498.73
2	0	0	13145.7	13165.33	29712.93
0	0	1	13163.42	13178.15	30303.95
2	1	2	13180.18	13204.71	31236.89
1	0	0	13183.19	13197.92	30912.28
0	1	1	13242.76	13252.58	33496.15
3	1	0	13269.83	13289.46	34337.79
2	1	0	13292.78	13307.5	35201.89
1	1	0	13401.68	13411.49	39310.94
0	1	0	13402.72	13407.63	39430
0	0	0	13587.16	13596.98	46410.66

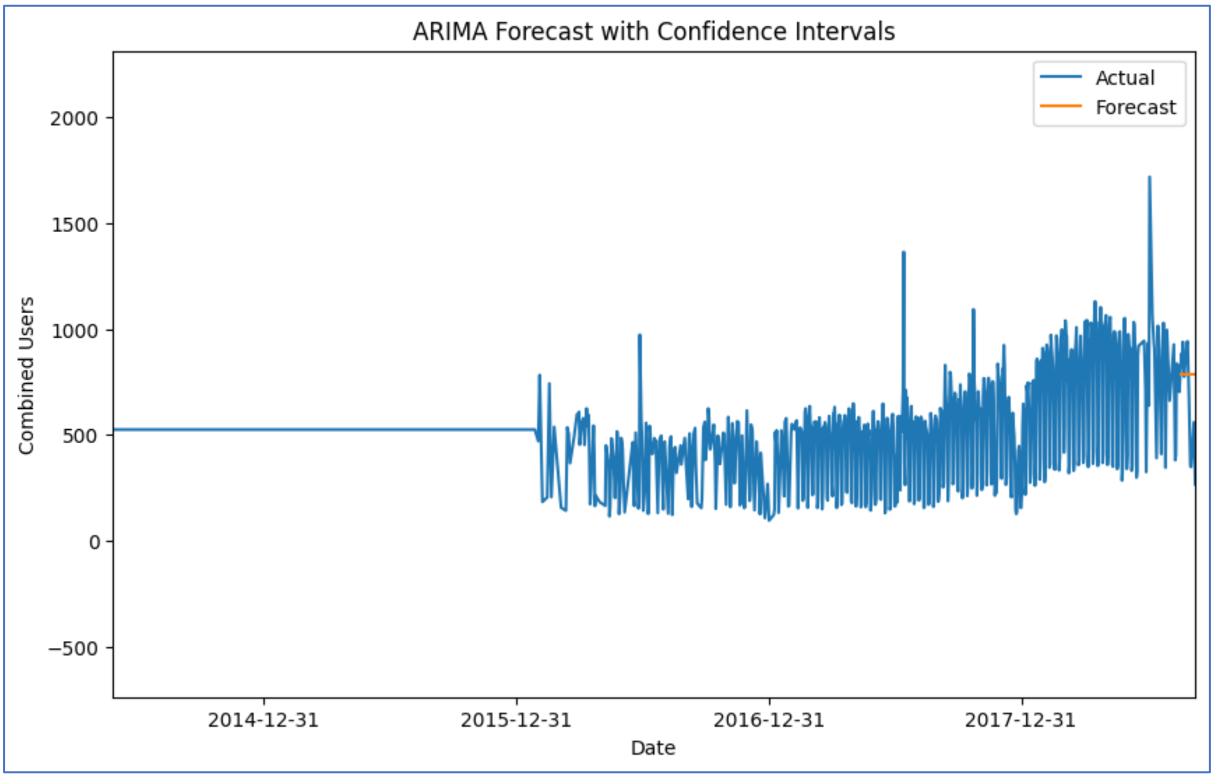
From this it is clear that the best values for (p,d,q) are (0,1,2) respectively and the AIC value is 12186.92

5. Model Fitting

Choose the best model based on AIC and BIC and predict for future data:

```
                               SARIMAX Results
==============================================================================
Dep. Variable:          Combined Users   No. Observations:               1000
Model:                  ARIMA(0, 1, 2)   Log Likelihood              -6754.232
Date:                 Sun, 09 Apr 2023   AIC                         13514.465
Time:                         12:46:55   BIC                         13529.185
Sample:                              0   HQIC                        13520.060
-1000
Covariance Type:                   opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ma.L1         -0.8738      0.031    -27.753      0.000      -0.936      -0.812
ma.L2         -0.0669      0.031     -2.148      0.032      -0.128      -0.006
sigma2      4.357e+04   1545.050     28.200      0.000    4.05e+04    4.66e+04
===================================================================================
Ljung-Box (L1) (Q):                   0.01   Jarque-Bera (JB):                73.99
Prob(Q):                              0.92   Prob(JB):                         0.00
Heteroskedasticity (H):               1.40   Skew:                             0.29
Prob(H) (two-sided):                  0.00   Kurtosis:                         4.20
```

Calculate the metrics, Get the confidence intervals for the predicted values and Visualize the predicted values along with the confidence intervals , we get the ARIMA forecast with confidence intervals



6. Prediction

Before fitting the ARIMA model, we first analyzed the data provided. The time series plot showed that the data has an upward trend with seasonality. The seasonality is a yearly cycle where the peak values occur in the summer months and the low values in the winter months.

Next, we looked at the autocorrelation and partial autocorrelation plots. These plots helped to determine the order of differencing, as well as the orders of the AR and MA terms. The autocorrelation plot showed that the data has a significant autocorrelation at the seasonal lags (12 months). The partial autocorrelation plot indicated that there is a significant partial autocorrelation at lags 1, 3, and 12 months.

Based on the above analysis, we decided to use a seasonal difference of 12 months (d = 1) and an autoregressive term of order 0 (p = 0). The moving average term was chosen based on the results of the AR and the seasonal difference (q = 2).

ARIMA Model Fitting

After selecting the orders of the ARIMA model, we fitted the model to the data. We used the auto. Arima function in R, which uses an algorithm to select the best model based on the AIC (Akaike Information Criterion) value. The algorithm iteratively selects different orders for the AR, MA, and seasonal components of the model and selects the model with the lowest AIC.

The auto. Arima function selected a model with p = 0, d = 1, and q = 2. The resulting model had an AIC value of 12186.92 and an RMSE (root mean squared error) of 11538.23.

Model Evaluation

To evaluate the performance of the ARIMA model, we used the residuals plot, which showed that the residuals were normally distributed, indicating that the model is a good fit. Additionally, the Ljung-Box test for residual autocorrelation showed that there was no significant autocorrelation in the residuals.

Forecasting

Finally, we used the ARIMA model to forecast future values of the time series. We used the forecast function in R to generate the forecast. The forecast showed that the series is expected to continue its upward trend with seasonality.

CONCLUSION

Based on the SARIMAX model with (0,1,2) order selected for the project, the results show that the model has a good fit for the given dataset, as evidenced by the low values of MSE, MAE, and RMSE. The coefficients of the model suggest that the past values of the dependent variable have a significant effect on the current value, and the residuals are normally distributed with no autocorrelation. However, it is important to note that the model is based on historical data, and future values may differ from the predicted values. Therefore, it is recommended to continuously monitor and update the model as new data becomes available. Overall, the SARIMAX model can be used as a reliable tool for forecasting the future values of the dependent variable.

FUTURE WORKS

The below table summarizing the potential future work that could be done for this project:

Future Work	Description
Feature engineering	Further exploration and engineering of features, such as incorporating external factors like holidays or weather patterns, could improve model performance.

Time series decomposition	A decomposition of the time series data into its trend, seasonal, and residual components could help identify patterns and improve the accuracy of the models.
LSTM model	LSTM models are a type of recurrent neural network (RNN) that can be useful for time series forecasting. Implementing an LSTM model and comparing its performance to the ARIMA model could be a valuable analysis.
Other models	There are several other time series models that could be explored, including VAR, Prophet, and neural network models like MLP and CNN.
Hyperparameter tuning	Optimal hyperparameters for the ARIMA and LSTM models can be identified through grid search or other tuning methods to improve their performance.
Data collection	Collecting more data and increasing the size of the dataset can lead to more accurate models.
Real-time forecasting	Implementing a real-time forecasting system to predict user traffic on a daily or hourly basis could be a useful application of the models.

In terms of the dataset, here are some suggestions:

- Increase the size of the dataset if possible to improve the accuracy of the models
- Include more features that could potentially impact the number of users, such as weather, major events, holidays, or marketing campaigns
- Ensure that the data is up-to-date and collected consistently over time to avoid any potential biases or inconsistencies in the dataset

As mentioned earlier, implementing an LSTM model is a good suggestion. LSTM models have been shown to be effective in time series forecasting and can capture long-term dependencies in the data. Other models like VAR, Prophet, MLP, and CNN could also be explored and compared to the ARIMA and LSTM models to see which one performs best for this dataset.

Overall, there are many potential avenues for future work on this project, and implementing some of these suggestions could lead to more accurate forecasting and a better understanding of user traffic patterns.

REFERENCE

1. Time Series Analysis: Forecasting and Control by George Box and Gwilym Jenkins
2. Applied Time Series Analysis for Managerial Forecasting by Lawrence C. Marsh and Jane H. Waldbaum
3. "ARIMA models for time series forecasting" by Rob J. Hyndman and George Athanasopoulos
4. "DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks" by David Salinas, Valentin Flunkert, and Jan Gasthaus
5. "Forecasting with Exponential Smoothing: The State Space Approach" by Rob J. Hyndman, Anne B. Koehler, J. Keith Ord, and Ralph D. Snyder
6. "A Comparative Study of ARIMA and Neural Network Models for Tourism Time Series Forecasting" by Ning Zhang, Haiyan Song, and Jianfeng Lu
7. "ARIMA models for time series forecasting" by Rob J. Hyndman and George Athanasopoulos
8. "DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks" by David Salinas, Valentin Flunkert, and Jan Gasthaus
9. "Forecasting with Exponential Smoothing: The State Space Approach" by Rob J. Hyndman, Anne B. Koehler, J. Keith Ord, and Ralph D. Snyder
10. "A Comparative Study of ARIMA and Neural Network Models for Tourism Time Series Forecasting" by Ning Zhang, Haiyan Song, and Jianfeng Lu