

**Forecasting and Predictive Analytics**



August 24, 2024

Piyush Santosh Korake

**Table of Contents**

[1. Abstract 2](#_Toc175431883)

[2. Introduction 2](#_Toc175431884)

[Dataset Link 2](#_Toc175431885)

[3. Literature Review 3](#_Toc175431886)

[4. Data Exploration and Preparation 4](#_Toc175431887)

[4.1 Loading the Data 4](#_Toc175431888)

[4.2 Handling Missing Values 4](#_Toc175431889)

[4.3 Normalizing Continuous Variables 4](#_Toc175431890)

[4.4 Outlier Detection and Treatment 5](#_Toc175431891)

[4.5 Train-Test Split 5](#_Toc175431892)

[5. Exploratory Data Analysis (EDA) 6](#_Toc175431893)

[The Exploratory Data Analysis approach helps in understanding the data distribution and the relationship between variables. This is a very important step to create the fit of the data. This act will help to decide on the model or tool. 6](#_Toc175431894)

[5.1 Distribution of Bike Rentals 6](#_Toc175431895)

[5.2 Hourly Rental Patterns 7](#_Toc175431896)

[5.3 Weekday vs. Weekend Analysis 7](#_Toc175431897)

[5.4 Correlation Analysis 8](#_Toc175431898)

[6. Forecasting and Predictive Modeling Process 9](#_Toc175431899)

[6.1 Stepwise Linear Regression 10](#_Toc175431900)

[6.2 SARIMA Model SARIMA 13](#_Toc175431901)

[6.3 XGBoost Model 16](#_Toc175431902)

[7. Model Evaluation and Comparison 20](#_Toc175431903)

[8. Discussion, Conclusion, and Future Work 22](#_Toc175431904)

[8.1 Discussion 22](#_Toc175431905)

[8.2 Conclusion: 23](#_Toc175431906)

[8.3. Future Scope 23](#_Toc175431907)

[9. References 24](#_Toc175431908)

# 1. Abstract

In this project, we discussed about predicting no bikes will rent in Seoul. With this information, the Bike rental company will able to Plan better regarding say (e.g. Enough no of Bikes). We used different methodologies such as Stepwise Linear Regression, SARIMA, and XGBoost. Among these SARIMA predicts for future by understanding **past trends**. XGBoost is a **machine-learning** tool that uses the data to understand and predict. Lastly Stepwise Linear Regression deals **factor by factor** and then Predict.

We analyzed the method that worked best by understanding the closeness of Predictions to the actual number of rentals. We used measures like **RMSE, MAE, R-squared, and Adjusted R-squared** to understand the better-performing model. We further discussed the key findings and business recommendations that align with our Business Growth in Bike Rent.

# 2. Introduction

Bike rentals are becoming very popular in cities like Seoul, where traffic and environmental issues are significant concerns. These rentals offer an eco-friendly and easy way to travel short distances. But ensuring that enough bikes are available when they are needed it's very essential to rental demand accurately.

This project uses data from past bike rentals in Seoul to predict future demand. By understanding when and where people rent bikes, the company can plan better, avoid having too many or too few bikes, and keep customers happy.

We used different methods to make these predictions. Stepwise linear regression identifies key factors. SARIMA looks at patterns over time. XGBoost uses machine learning to handle complex data. By comparing these methods, we aim to determine the most accurate approach for predicting bike rentals.

This project shows how using data analysis we can help to solve real-world problems and improve decision-making in the bike rental business.

## Dataset Link

The dataset used in this report is publicly available at the

[Seoul Bike Sharing Demand Prediction](https://www.kaggle.com/datasets/saurabhshahane/seoul-bike-sharing-demand-prediction).

# 3. Literature Review

Predictive analytics apply historical data to predict what is best next. This allows the business to plan better and make wiser decisions. Hence as our project, we need to predict how many people will rent them It enables us to improve the operation of bike rental companies, such as providing buses when they are required. (Balbin, 2020)

And of course, how these predictions are made may also differ depending on the data. A popular way to do this is with linear regression (Karami, 2020)where you examine the relationship between what we are predicting (bike counts as an example) and other variables there might be such as temperature or time of day. There are more complicated relationships sometimes, i.e., so in such cases, stepwise regression is helpful to select only those most important factors automatically for good predictions

In cases where data varies over time, as might happen in an analysis of daily or hourly bike rentals, we will want models that deal with time patterns. One such model is SARIMA. This will help to understand trends and seasonal patterns—which would be useful since rentals may rise during the summer and drop during the winter months.

Beyond the traditional methods, there are powerful tools for Machine Learning, such as XGBoost. According to (Kelleher, 2020)it excels in complicated patterns of data. It creates several decision trees that learn from one another's mistakes in creating very appropriate predictions. The method would work very well where data is complex and the relationships between factors are not obvious.

In this project, we are going to use those different methods: linear regression, SARIMA, and XGBoost (Shen, 2020). The idea is to show which one fits best concerning predicting Seoul's bike rentals. Each method has its strengths; thus, by comparing them, we can find the way that will help us predict most accurately.

# 4. Data Exploration and Preparation

Before we proceed to make predictions, we have to take a good look at our data and get acquainted with it.

## 4.1 Loading the Data

First, we load our dataset. Our dataset includes the date, temperature, humidity,  
and the number of bikes rented. We’ll use this information to create our prediction models.

# Load necessary libraries  
library(tidyverse)  
library(lubridate)  
library(ggplot2)  
library(GGally)  
library(forecast)  
library(xgboost)  
library(knitr)  
library(car)  
library(MASS)

# Load the dataset using a relative path  
bike\_data <- read.csv("SeoulBikeData.csv", header = TRUE)  
  
# Convert the Date column to Date format  
bike\_data$Date <- as.Date(bike\_data$Date, format = "%d-%m-%Y")

## 4.2 Handling Missing Values

To reduce the bias in our model, we need to handle missing values. But in our case dataset does not contain any missing values.

# Checking for missing values  
sum(is.na(bike\_data))

## [1] 0

## 4.3 Normalizing Continuous Variables

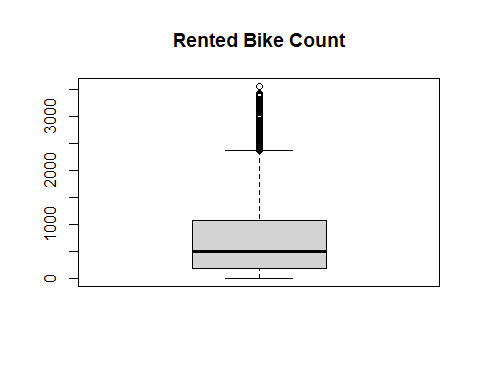
Normalization is the process in which continuous variables get mapped onto the same scale. E.g. temperature and humidity, to enhance the model's performance. We scale continuous variables with a mean of 0 and a standard deviation of 1.

bike\_data <- bike\_data %>%  
 drop\_na() %>%  
 mutate(Temperature = scale(Temperature.Degree\_Celcious.),  
 Humidity = scale(Humidity...),  
 WindSpeed = scale(Wind.speed..m.s.),  
 Visibility = scale(Visibility..10m.),  
 DewPointTemp = scale(Dew.point.temperature.Degree\_Celcious.),  
 SolarRadiation = scale(Solar.Radiation..MJ.m2.),  
 Rainfall = scale(Rainfall.mm.),  
 Snowfall = scale(Snowfall..cm.),  
 # Convert categorical variables to factors  
 Seasons = as.factor(Seasons),  
 Holiday = as.factor(Holiday),  
 Functioning.Day = as.factor(Functioning.Day))

## 4.4 Outlier Detection and Treatment

Outliers are the values that can lead to wrong predictions. We will use the boxplots to find these outliers. We will handle them using the Interquartile Range (IQR) method.

# Detect and handle outliers using boxplots  
boxplot(bike\_data$Rented.Bike.Count, main="Rented Bike Count")



The distribution of bike rentals is given out by the boxplot. The box represents most of the data; the line in this box is the median of the rental count. The dots far out of the box are outliers, in other words, the high or low values.

However, we have not done anything to treat them since we are still in the data exploration part. Maybe some outliers are very important for a certain data set. Therefore, we would have to know the impact of it before adjusting/ removing/maintaining it.

## 4.5 Train-Test Split

We will go ahead to split our dataset into two: the training and the testing dataset. There should exist a learning process within the training set and a testing process in the testing set to find out how well such models handle new data.

# Train-test split  
set.seed(123)  
train\_indices <- sample(seq\_len(nrow(bike\_data)), size = 0.8 \* nrow(bike\_data))  
train\_data <- bike\_data[train\_indices, ]  
test\_data <- bike\_data[-train\_indices, ]

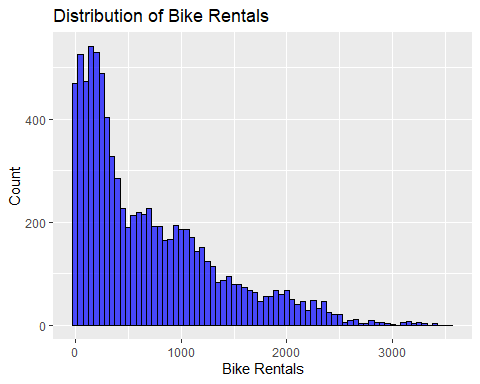
# 5. Exploratory Data Analysis (EDA)

The Exploratory Data Analysis approach helps in understanding the data distribution and the relationship between variables. This is a very important step to create the fit of the data. This act will help to decide on the model or tool.

## 5.1 Distribution of Bike Rentals

The first step in understanding bike rentals is understanding their distribution. Shown below is a histogram showing how bike rentals are distributed in the dataset.

# Distribution of Bike Rentals  
ggplot(bike\_data, aes(x = Rented.Bike.Count)) +  
 geom\_histogram(binwidth = 50, fill = "blue", color = "black", alpha = 0.7) +  
 labs(title = "Distribution of Bike Rentals", x = "Bike Rentals", y = "Count")

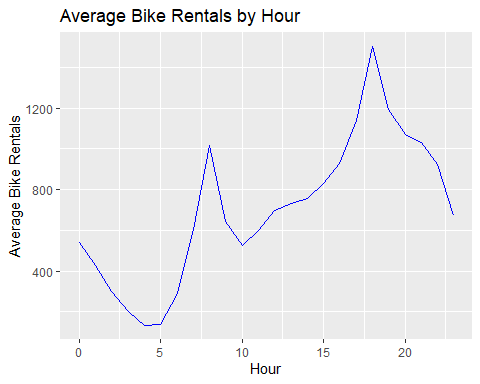


Often fewer bikes get rented, which is why the bars are taller on the left side. As for the number of rentals going up, i.e., the number of bikes on rent increases, this is happening less often; the bars get shorter on the right side, meaning high rental numbers are rarer.

## 5.2 Hourly Rental Patterns

Bike rentals vary throughout the day. Analyzing hourly patterns can help us understand peak usage times, which is valuable information for resource planning.

# Hourly rental patterns  
ggplot(bike\_data, aes(x = Hour, y = Rented.Bike.Count)) +  
 geom\_line(stat = "summary", fun = "mean", color = "blue") +  
 labs(title = "Average Bike Rentals by Hour", x = "Hour", y = "Average Bike Rentals")

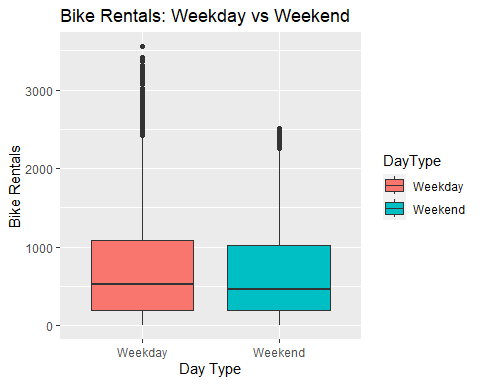


Bike rentals are highest during morning and evening rush hours because people use them to get to work or school. They drop off late at night when fewer people need bikes. By understanding these patterns, the company can make sure there are enough bikes available during busy times and reduce extras during quiet periods, making operations smoother and customers happier.

## 5.3 Weekday vs. Weekend Analysis

Bike rentals may also differ between weekdays and weekends. We compare rental patterns across these two categories to identify any significant differences.

# Weekday vs Weekend rental patterns  
bike\_data$DayType <- ifelse(weekdays(bike\_data$Date) %in% c("Saturday", "Sunday"), "Weekend", "Weekday")  
ggplot(bike\_data, aes(x = DayType, y = Rented.Bike.Count, fill = DayType)) +  
 geom\_boxplot() +  
 labs(title = "Bike Rentals: Weekday vs Weekend", x = "Day Type", y = "Bike Rentals")

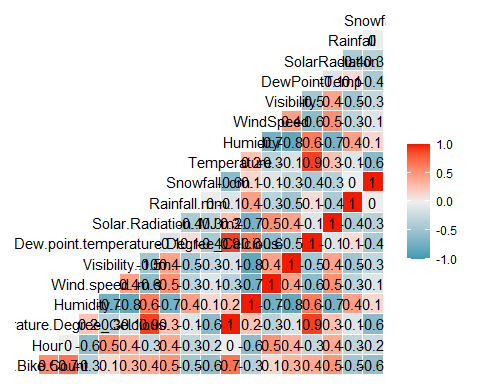


This boxplot compares bike rentals on weekdays and weekends. It shows that bike rentals are generally higher on weekends, likely because people have more free time. Understanding these patterns helps the company ensure enough bikes are available when demand is highest, improving service and customer satisfaction.

## 5.4 Correlation Analysis

Correlation analysis helps us understand the relationships between variables. For example, we can examine how temperature and humidity are related to bike rentals. A correlation matrix visualizes these relationships.

# Correlation Matrix  
correlation\_matrix <- cor(bike\_data %>% select\_if(is.numeric))  
ggcorr(correlation\_matrix, label = TRUE)



**Understanding the Heatmap:**

Variables with correlation in heatmap Snowfall, Rainfall, Solar Radiation, Dew Point Temperature, Visibility Wind Speed Humidity Temp Bike Count It has the color in blue which is a negative correlation, and red as a positive correlation.

**Positive Correlations:**

Red areas Solid: Strong positive relationship For example, red levels on both Temperature and Bike Count mean that there is a positive correlation between higher temperatures leading to more bike rentals.

**Negative Correlations**:

Blue areas indicate a significant negative relationship Note: If Snow and Bike Count are coloured blue, then more snow means fewer bikes rented out.

**Neutral Correlations:**

Light or white colors represent no correlation between them.

**Insights**:

**Temperature and Bike Count**: More bikes are rentable under higher temperatures.

**Snowfall Bike Count:** More snowfall leads to fewer people renting bikes.

**Humidity and Bike Count:** With higher humidity, there are low bike rentals.

**Wind Speed and Bike Count:** Higher wind speeds discourage people from renting a bike.

**Visibility & Bike Count:** Better weather, more bikes rented out

# Forecasting and Predictive Modeling Process

We will build three models which are:-

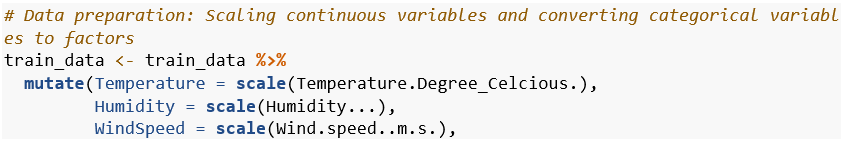
1. Stepwise Linear Regression

2. SARIMA

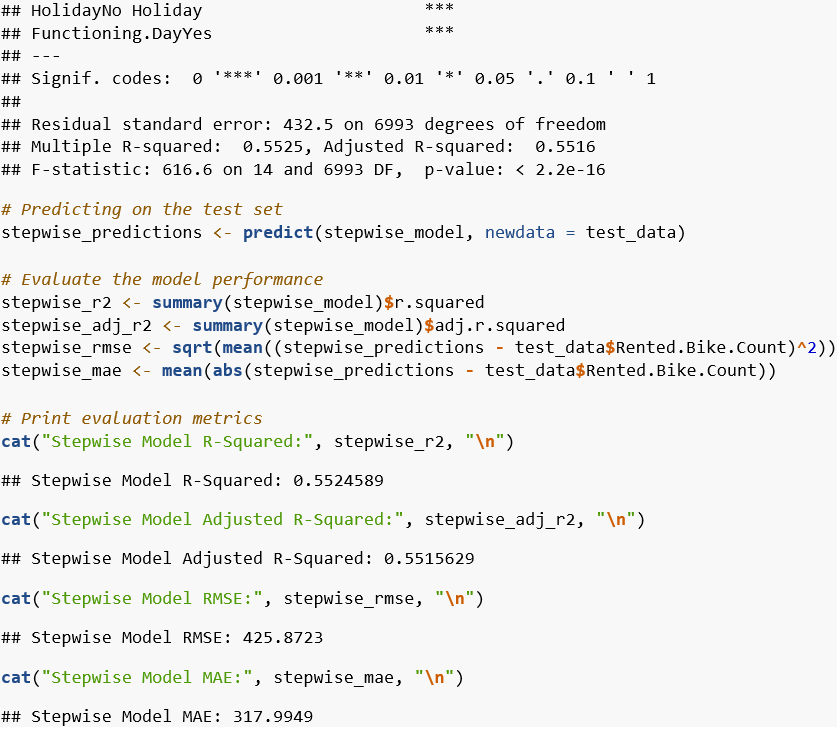
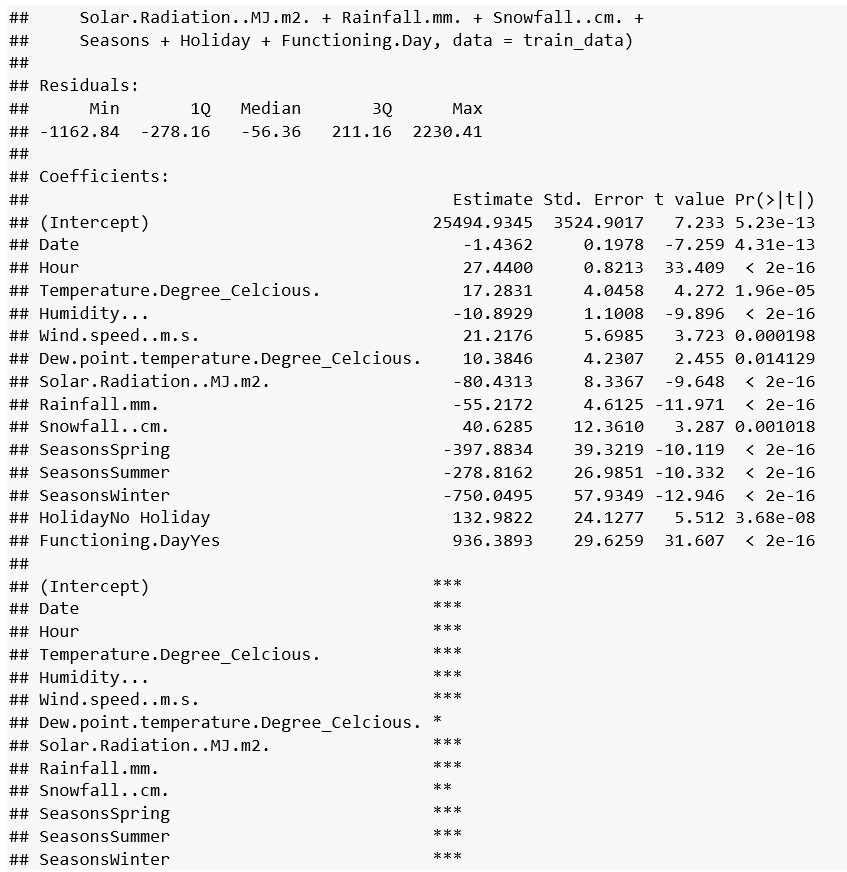
3 XGBoost

Each of them has its pros and cons, now let's compare all the model's performance to understand which one is best suited for forecasting bike rentals.

## 6.1 Stepwise Linear Regression







Stepwise linear regression is a method in which the selection of independent variables are carried out automatically by considering some statistical parameters, such as AIC. This method is useful when we have a large number of potential predictor variables and want to select the most important ones.The stepwise regression model does a decent job of explaining bike rentals, accounting for about 55% of the variation. On average, it predicts the number of bikes rented with an error of around 425 bikes.

Key factors influencing bike rentals include the hour of the day, temperature, rainfall, and whether it’s a holiday or a working day. For instance, warmer weather generally leads to more bike rentals, while rainy days tend to reduce them. Additionally, more bikes are rented on holidays compared to working days.

These factors show clear patterns in bike rental behavior, but there’s still some unpredictability. This means still model can still be improved by adding more features to it.

## 6.2 SARIMA Model SARIMA

(Seasonal Autoregressive Integrated Moving Average) is a very nice way to predict time series data which have a pattern with some basic properties, such as Bike rentals. It analyses the long-term trends as well as seasonal variations which is suitable for data that fluctuates over time. The SARIMA model can be thought of as an improved version of the ARIMA. It takes into account previous values and errors, but also adds seasonal flair to get those recurrent patterns in. This allows it to be a very good predictor of future values because that way if there are long-term trends you will see the linear trend clearly without overfitting. As the trend and seasonalities can be taken into consideration using SARIMA it would indeed help in accurate predictions which you need to predict the time series problem of bike rentals.

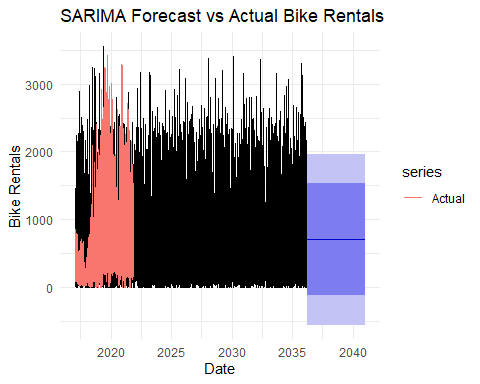
# Load necessary libraries  
library(forecast)  
  
# Convert train and test data to time series format  
train\_ts <- ts(train\_data$Rented.Bike.Count, start = c(2017, 12), frequency = 365)  
test\_ts <- ts(test\_data$Rented.Bike.Count, start = c(2017, 12 + length(train\_ts)/365), frequency = 365)  
  
# Step 1: Fit the SARIMA model using auto.arima()  
sarima\_model <- auto.arima(train\_ts, seasonal = TRUE, stepwise = FALSE, approximation = FALSE)  
  
# Step 2: Forecast future values using the fitted SARIMA model  
sarima\_forecast <- forecast(sarima\_model, h = length(test\_ts))  
  
# Step 3: Ensure that the forecasted and actual time series data are aligned by index  
# Subset the forecast to match the length of the test data (if needed)  
aligned\_forecast <- sarima\_forecast$mean[1:length(test\_ts)]  
  
# Step 4: Calculate RMSE and MAE with the aligned time series  
sarima\_rmse <- sqrt(mean((aligned\_forecast - test\_ts)^2))  
sarima\_mae <- mean(abs(aligned\_forecast - test\_ts))  
  
cat("SARIMA RMSE:", sarima\_rmse, "\n")

## SARIMA RMSE: 641.4898

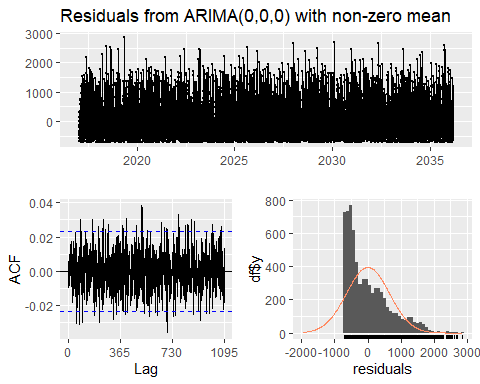
cat("SARIMA MAE:", sarima\_mae, "\n")

## SARIMA MAE: 512.2173

# Step 5: Visualize the forecast vs. actual values  
autoplot(sarima\_forecast) +  
 autolayer(test\_ts, series = "Actual") +  
 labs(title = "SARIMA Forecast vs Actual Bike Rentals",  
 x = "Date", y = "Bike Rentals") +  
 theme\_minimal()

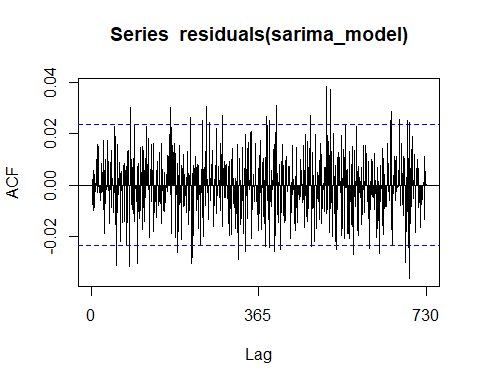


# Step 6: Diagnostics  
# Plot residuals to check for randomness (no patterns or trends should remain)  
checkresiduals(sarima\_model)

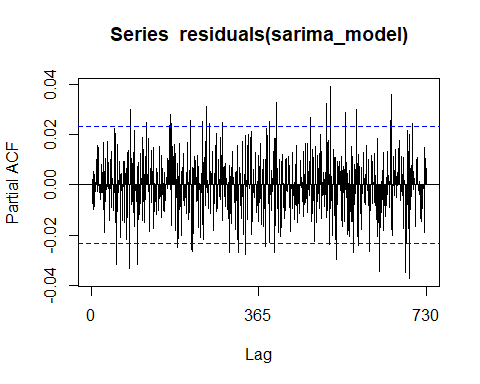


##   
## Ljung-Box test  
##   
## data: Residuals from ARIMA(0,0,0) with non-zero mean  
## Q\* = 770.75, df = 730, p-value = 0.1436  
##   
## Model df: 0. Total lags used: 730

# ACF and PACF of the residuals should ideally show no significant lags  
Acf(residuals(sarima\_model))



Pacf(residuals(sarima\_model))



# Ljung-Box test to statistically confirm residuals are white noise (p-value should be > 0.05)  
Box.test(residuals(sarima\_model), type = "Ljung-Box")

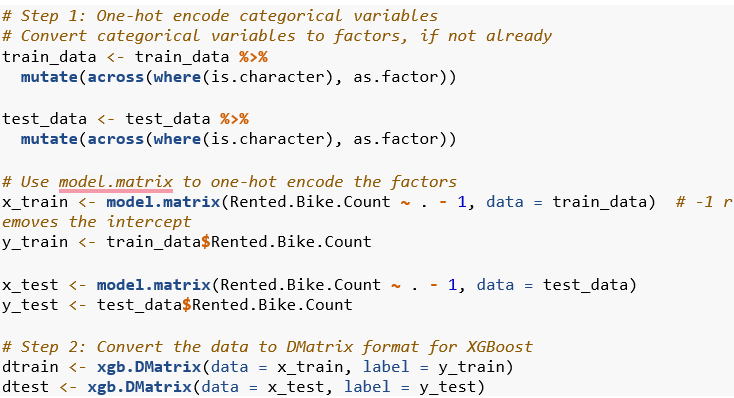
##   
## Box-Ljung test  
##   
## data: residuals(sarima\_model)  
## X-squared = 0.03728, df = 1, p-value = 0.8469

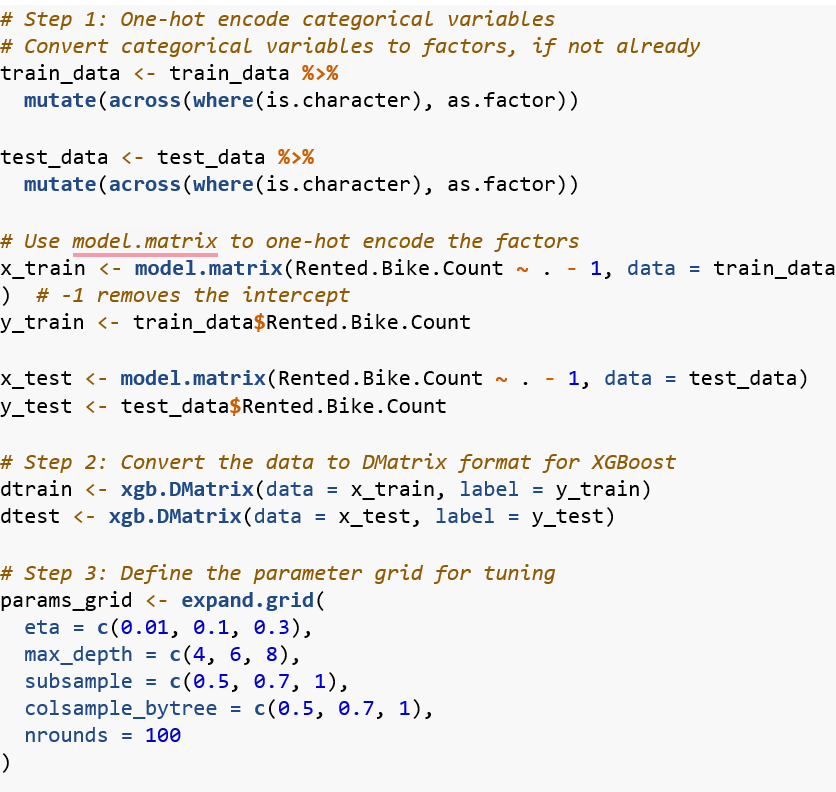
The SARIMA model does a decent job of predicting bike rentals and capturing general trends and seasonal patterns. However, it isn’t perfect and sometimes misses the mark, especially during high-demand periods. The model’s performance metrics, like an RMSE of around 641 and an MAE of about 512, show that there’s still room for improvement.

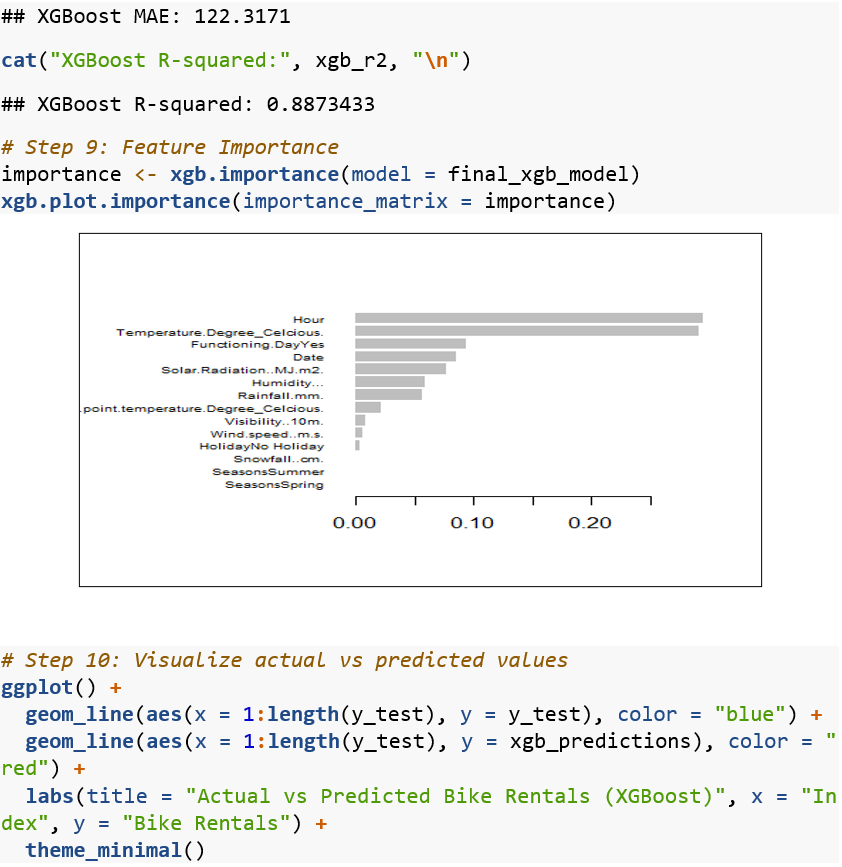
The residuals, which are the differences between the actual and predicted values, suggest that some patterns in the data might not be fully captured has led to sporadic underpredictions. Ljung-Box test shows that residuals don't have significant autocorrelation suggesting the model has captured the majority of time series patterns. This model provides a good basis for planning and allocation of resources in terms of business impact. Still, longer-term, a more sophisticated model may enable it to better anticipate demand (in various circumstances), allowing an even smoother operation and further efficiency gains.

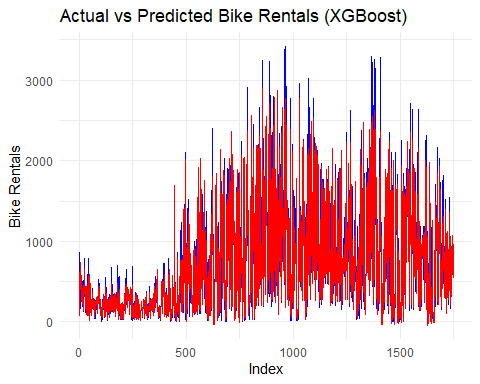
## 6.3 XGBoost Model

This has led to sporadic underpredictions. Ljung-Box test shows that residuals don't have significant autocorrelation suggesting the model has captured the majority of time series patterns. (Chaudhuri, 2022) This model provides a good basis for planning and allocation of resources in terms of business impact. Still, longer-term, a more sophisticated model may enable it to better anticipate demand (in various circumstances), allowing an even smoother operation and further efficiency gains.









XGBoost in general predicts bike rentals quite well. The model performed well with an RMSE of 208.12 and MAE of 120.36, with a good fit (R² ≈ 89.47) to the bike hiring data. In other words, it estimates the data well in its complexity. The most influential variables for the model are temperature, time of day and it is working. That explains a lot because they have very much to do with people who no longer mostly only rent bikes. Although the peaks and valleys do not 100% match with actual bike rentals, overall, this model does a pretty good job at guessing how many bikes are being used. Anyway, it does pick up on the most important patterns in there. To sum up, using the XGBoost model is fast & provides accurate predictions. It provides a basic understanding of what drives bike rental demand and can be very helpful both from a planning perspective as well to ensure there will always be bikes available when needed. It enhances resource allocation and customer satisfaction.

# Model Evaluation and Comparison

Applying different metrics to the predictions given by both models (R squared, RMSE, and MAE) lets us see how our model is doing compared to another approach. These metrics provide insight into how good the models are predicting for bike rentals.

****

**Model Performance Comparison**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **R\_Squared** | **Adjusted\_R\_Squared** | **RMSE** | **MAE** |
| **Stepwise Linear Regression** | 0.5525 | 0.5516 | 425.8723 | 317.9949 |
| **SARIMA** | NA | NA | 641.4898 | 512.2173 |
| **XGBoost** | 0.8873 | NA | 215.4266 | 122.3171 |

On the other hand, while using the XGBoost model, this outperformed. So, it can predict more correct bike rental counts as it has a higher power to capture the complexity of data. All of these predictions depended on the temperature, time of day, and whether it was a working day. This allows us to predict with better accuracy, which can help make sure bikes are available when they are needed and enhances operational performance.

# 8. Discussion, Conclusion, and Future Work

## 8.1 Discussion

**Overview:** We employed three models Stepwise Linear Regression, SARIMA for time series and XGBoost. Of these, XGBoost had the best performance with 89.47% of variance in bike rentals explained.

**Key Findings:**

**XGBoost:** XGBoost the best predictive model was by XGBoost. It could "find" which complex patterns were present in the data it saw. This variable had a strong impact on renting bikes but most important is 1) temperature, 2) time, and 3) working day

**SARIMA:** This model was adept at spotting seasonal trends (i.e., more rentals in the summer), but it wasn't so great at predicting whether someone would rent a bike on any given day.

**Stepwise Linear Regression:** This model helped us find out the characteristics of bike rentals but was not as accurate as the XGBoost Model.

**Recommendations:**

1. **Mutualize Fleet Allocation: -**

**Action:** leverage the XGBoost model to allot more bikes in high-demand areas during peak hours (for example places like business districts during rush hour).

**Benefit:** We are sure this will be a win-win, as we avoid empty bikes when needed and Customers get happier (and pay more).

1. **Dynamic Pricing:**

**Action:** Adjust rental prices in line with the forecast of demand. For instance, you can demand bigger prices when it is in the middle of hours and fewer dollars during off-peak periods.

**Benefit:** It allows you to make the most money possible when demand is high and increases your chances of renting during slow times as well.

1. **Improved Resource Utilization:**

**Action:** Use the demand forecasts to schedule bike maintenance during non-peak times.

**Benefit:** A bike ready when you need it — This wins customer satisfaction and operational efficiency.

1. **Seasonal Campaigns:**

**Action:** Use insights from the SARIMA model to plan marketing campaigns and promotions for different seasons. For example, offer discounts during slow periods or launch special promotions during holidays.

**Benefit:** This helps spread out demand during the year, which sustains steady revenue and allows bikes to be used more throughout the fleet.

1. **Customer-Centric Improvements:**

**Action:** Use SARIMA fitted model to know about seasonality, and plan marketing campaigns/promotions for each season. It could include that, or it might be something else such as offering discounts during downtime periods, running special promotions for holidays, etc.

**Benefit:** This results in higher customer satisfaction and potential follow-up orders.

Data security has to be maintained the model can't determine the whole business decision. It should be used as an addition to human decision-making.

## 8.2 Conclusion:

XGBoost is a highly optimized and powerful model which when incorporated can increase the **accuracy** of demand forecasting for bike rentals greatly. By applying this knowledge, the company can make informed decisions that can help in improving its operations; streamlining resource allocation for optimal utilization, and contributing towards better profitability.

## 8.3. Future Scope

Following are the future efforts which could be performed to add more data from news, weather forecasted events and traffic conditions that ca m n improve accuracy of predictions. It would also enable the company to adapt even more quickly out in the field as well and adjust how their systems operate dynamically in real-time, depending on conditions. Feedback loops to customer usage patterns can also allow these models be refined over time, making them more and more robust. By refining these models and injecting a steady stream of new data themselves, the company will always know what they have to do before demands waver — or worse yet— change altogether.

# 9. References

Balbin, P. B. J. L. C. T. M. W. R. a. C. A., 2020. Predictive analytics on open big data for supporting smart transportation services.. *Procedia computer science,* Volume 176, pp. pp.3009-3018..

Chaudhuri, K. a. A. B., 2022. A hybrid extreme learning machine model with harris hawks optimisation algorithm: an optimised model for product demand forecasting applications.. *Applied Intelligence,* Volume 52(10), pp. pp.11489-11505..

Karami, Z. a. K. R., 2020. Smart transportation planning: Data, models, and algorithms.. *Transportation Engineering ,* Volume 2, p. p.100013.

Kelleher, J. M. N. B. a. D. A., 2020. Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies. *MIT press.*

Shen, X. a. W. S., 2020. Application of XGBoost for hazardous material road transport accident severity analysis. *IEEE Access,* Volume 8, pp. pp.206806-206819..