Seoul Bike Sharing Demand Prediction

Rohit Verma
Data science trainee,
AlmaBetter, Bangalore

Abstract:

"Seoul Bike Sharing Demand Prediction" is the process by which bicycles are procured on several basis- hourly, weekly, membership-wise, etc. This phenomenon has seen its stock rise to considerable levels due to a global effort towards reducing the carbon footprint, leading to climate change, unprecedented natural disasters, ozone layer depletion, and other environmental anomalies.

In our project, we chose to analyse a dataset pertaining to Bike Rental Demand from South Korean city of Seoul, comprising of climatic variables like Temperature, Humidity, Rainfall, Snowfall, Dew Point Temperature, and others. For the available raw data, firstly, a through pre-processing was done after which a Here, hourly rental bike count is the regress and. To an extent, our linear model was able to explain the factors orchestrating the hourly demand of rental bikes.

Introduction:

Bike Sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city. Using these systems, people are able to rent a bike from one location and return it to a different place on an as-needed basis.

The first bike-share programs began in 1960s Europe, but the concept did not take off worldwide until the mid-2000s. In North America, they tend to be affiliated with municipal governments, though some programs, particularly in small college towns, centre on university campuses.

The typical bike-share has several defining characteristics and features, including station-based bikes and payment systems, membership, and pass fees, and per-hour usage fees. Programs are generally intuitive enough for novice users to understand. And, despite some variation, the differences are usually small enough to prevent confusion when a regular user of one city's bike-share uses another city's program for the first time.

With the onset of Industry 4.0, integration of Internet of Things (IoT) systems with bike-sharing ecosystem has eased the rental process to a significant extent. Real-time tracking of bikes, traffic density, and climate variables aids in gaining useful knowledge about trends, and patterns of renting process, thereby allowing an incisive prediction to meet future demand.

Considering the current ecosystem, bike-sharing can play a vital role in reducing the impact of carbon emissions and other greenhouse gases- major contributors in climate change. Sustainable and clean transport system, if successful, can provide a greener alternative to the traditional car-pool system, and help in reducing traffic congestion, too.

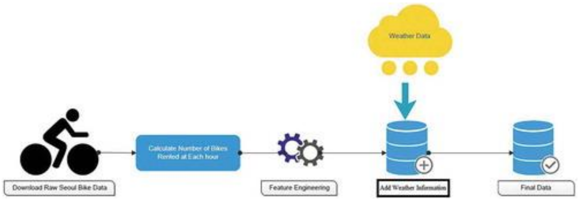
In addition to the environmental benefits, the sharing systems will impart healthier habits among commuting public, who in the hustle of tasking daily routine, often are unable to integrate optimum level of physical activity, which results in a barrage of ailments.
On a positive note, the global Bike-Sharing market size, which was sized at USD 2570.9 million in 2019, is expected to breach the USD 13780 million mark by 2026, with Compound Annual Growth Rate (CAGR) of 26.8% during 2021-2026, as per Market Analysis via MarketWatch.

For our project, we retrieved data from UCI Machine Learning Repository. The dataset contained per day Bike Rental Count with 8760 entries, possessing 14 attributes, out of which 13 variables-12 independent and one dependentform the part of our Regression Analysis. The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information. Date does not provide relevant information to generate a model to predict the Rental Bike Count. The primary objective was to build a superior statistical model to predict the number of bicycles that can be rented with the availability of data and understand the trends and factors affecting the rented bike count on a particular day.

Methodology:

Although the data used must be kept in private, it will be important to compare the results with other conventional machine learning algorithms to signify the importance of positives and negatives of each method considered in this study. This section briefly explains the algorithms used in this study. Also, there is no single machine-learning algorithm, which must be optimally applied for every scenario (Wolpert & Macready, 1997). Therefore, twelve prediction algorithms were considered in this study to compare their performance with each other.

Step 1- EDA

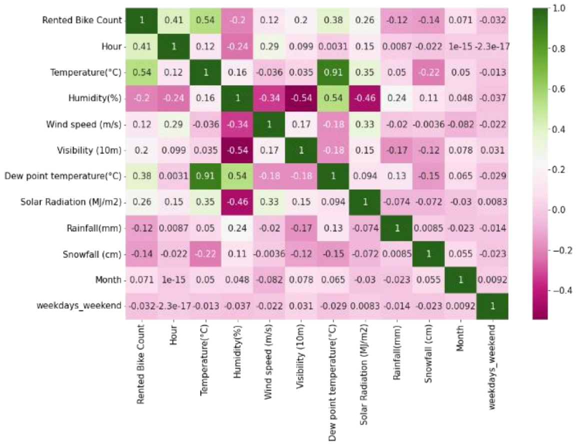

Weather Data
Download Raw Seoul Bike Data
Calculate Number of Bikes Rented at Each hour
Feature Engineering
Add Weather Information
Final Data

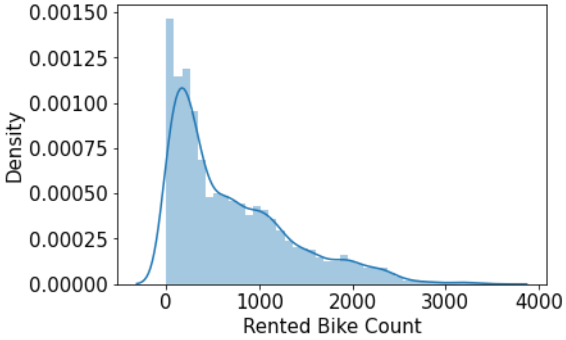
I. Preprocessing

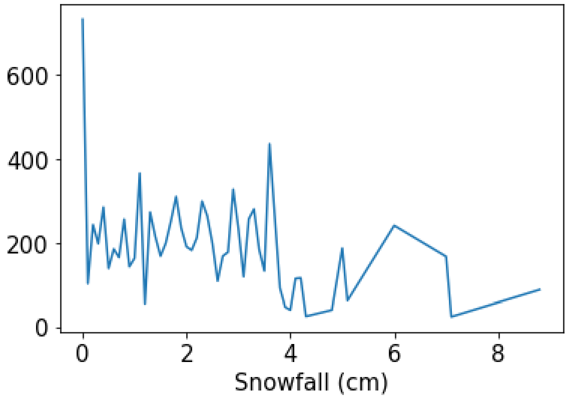
Data preprocessing is a data mining technique which consists in transforming the data in order to make it understandable. It could be changing the type, the format, splitting the data, verify that there are no missing values but also creating new columns thanks to columns we initially have. In machine learning, the data processing step is critical because it involves cleaning, integration, transformation, scaling, standardize data and many other tasks, in order to have a good preparation for the application of models. To begin we first did some data exploration by checking types, missing values and data description. We also changed the date type to DateTime which was initially a str object. Then, we created a column which takes the hour of the day and returns if it the day of the week night or day moment of the day (because we remind you that data is collected by hour), in order to do data visualization with the target. From the date, we also created two columns with the day of the week and the month of the year corresponding. And finally also did and encoding on the day to convert ’WeekDays’ in order to to visualize it and make it a feature for ML models.

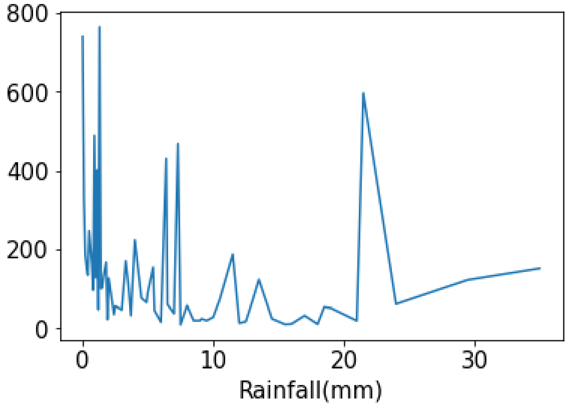
Step 2. Visualization

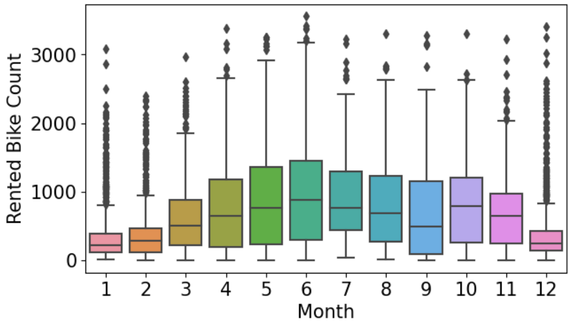
The first plot is maybe one of the most important because it shows all the correlations of the features. To complete this visualisation, we created a ranking of the features which are the most correlated to the target. So it gives an idea of which features we have to focus on. The following plot that we shows us the sum of rented bikes month by month in 2018. This distribution clearly shows us that there is a high raise of rents from April to November. That’s why we decided to verify that the rents raised proportionally to the temperature and to the solar radiation. Moreover on the notebook, you can see that as expected the raise of rainfall and snowfall comes with a decrease of the rents which is totally logical. We also created a features : ”Night/Day” in order to see the distribution following the moment of the day. From 8pm to 5am, we decided to qualify this moment as ’night’ and the rest is ’day’. Finally we wondered what were the hours during which the rents were the highest, and we found that it was around 8am and around 6pm which confirms that people take bikes to go to school or work and go home at night. This analysis is very interesting because it shows that Koreans take the global warming very seriously and respect the earth.

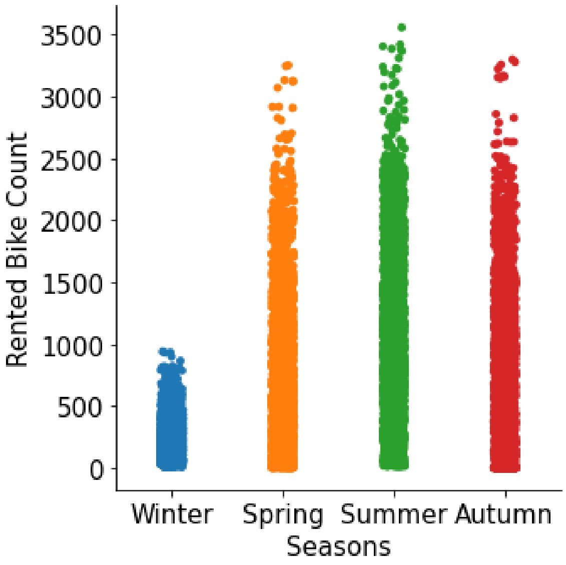


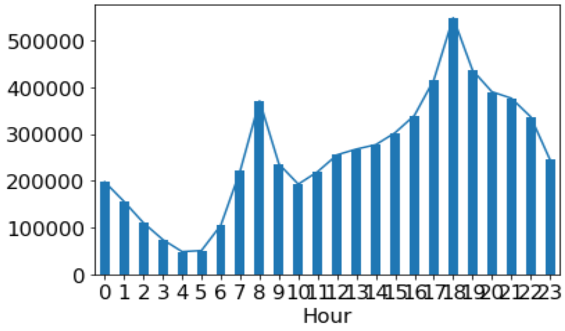

Density
0.00150
0.00125
0.00100
0.00075
0.00050
0.00025
0.00000
0
1000
2000
3000
4000
Rented Bike Count

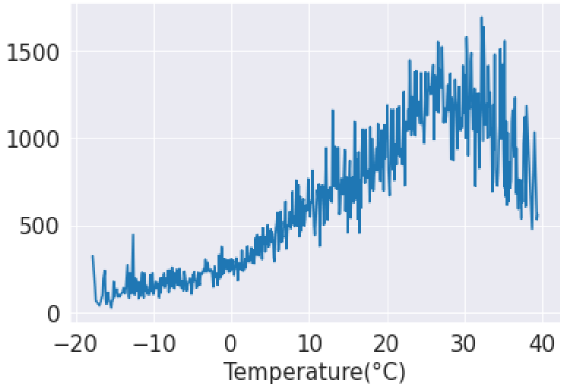

600
400
200
0
0
2
4
6
8
Snowfall (cm)

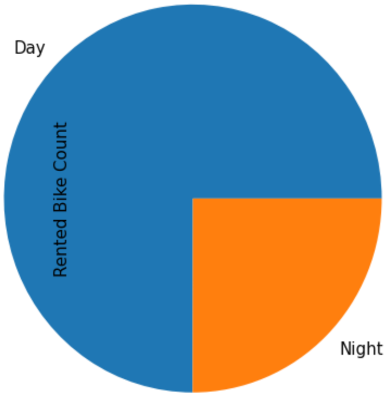

800
600
400
200
0
0
10
20
30
Rainfall(mm)


Rented Bike Count
3000
2000
1000
0
1
2
3
4
5
6
7
8
9
10
11
12
Month


Rented Bike Count
3500
3000
2500
2000
1500
1000
500
0
Winter
Spring
Summer
Autumn
Seasons


500000
400000
300000
200000
100000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour





. Let’s continue with the modeling part which is going to show us how to set machine learning algorithm to predict then number of rented bikes from weather conditions.

Step 3. Modeling

We have a regression problem because our target is the number of rented bikes per hour. So the goal of this part is to apply many algorithms in order to find the algorithm with the best indicator. The indicator we decided to choose is the R2. This choice is because we wanted to be able to compare these algorithms between them and to choose which one is the most efficient. Let’s apply regression techniques to our problem.

- **Linear multiple regression** - We ran a multiple linear regression, we assume that all the features have a linear relationship with the target. We also have to assume that these features have a Gaussian distribution and that features are not highly correlated between them, it is called multi-co linearity.

$$J = \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

The goal of this model is to minimize the RMSE and to get the R2 close to 1 or -1. We first fit the model on the training data and training target. Then we first predicted the training data then the test data in order to get indicators. For the training set, we got a R2 score = 0.48 and a RMSE = 468 which is not that huge knowing that we have 6394 values predicted. For the testing set, we got a R2 score = 0.44 and a RMSE = 467 which is not a very good score. The test and train score are very close.

$$\mathbf{y} = \boldsymbol{\theta}_1 + \boldsymbol{\theta}_2.\mathbf{x}$$

$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

- **Ridge and Lasso -** We first did the Ridge regression, and performed a grid search with different values of alpha. Then we calculated R2 train and test as we have an indicator to compare it to the linear regression.

$$\hat{y} = w[0] \times x[0] + w[1] \times x[1] + ... + w[n] \times x[n] + b \qquad (1.1)$$

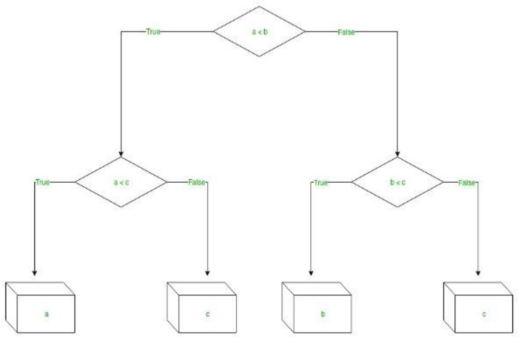
$$\sum_{i=1}^{M}(y_i-\hat{y}_i)^2=\sum_{i=1}^{M}\left(y_i-\sum_{j=0}^{p}w_j\times x_{ij}\right)^2+\lambda\sum_{j=0}^{p}w_j^2 \tag{1.3}$$

For the training set, we got a highest R2 score = 0.48. For the testing set, we got a R2 score = 0.47. These scores are very close to the previous model. That's why we are going to try other models. Then we tried the Lasso Regression which is a little bit different from Ridge regression.

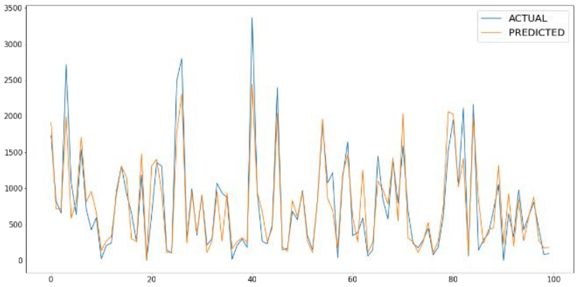
$$\text{For some } c > 0,\ \textstyle\sum_{j=0}^{p} w_j^2 < c$$

Unlike ridge, Lasso can exclude useless feature from the model because it reduces variance, so it helps do features selection. The Lasso regression can make features disappear because of the shape when we reduce dimension.The shape of the Lasso is a diamond whereas the Ridge is an ellipse. Then we performed a grid search on the Lasso for which we had the same of score as the Ridge grid search. We got 0.48 R2 score.

• **Decision Tree -** Then we applied a decision tree regressor on our data. We first scaled our X training and X testing sets. Then we applied a grid search on the model by tuning the feature 'max depth'. We fit our grid search on the X and y training sets. Then we kept the best model (with the best estimators) and got the score with the X and y testing sets. The result of the testing sets is 0.74 and is much more higher.

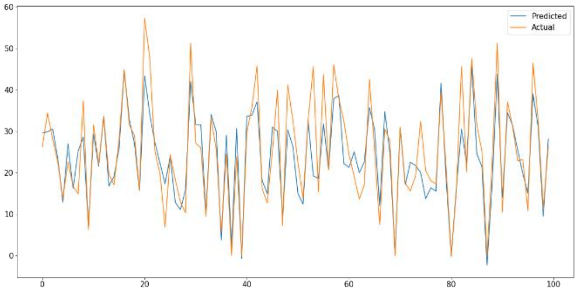


• **Random Forest regressor -** Then we applied a random forest regressor on our data. We also take the scaled X training and X testing sets. Then we made a grid search on the model by tuning the feature 'max depth', 'n estimators', 'min samples split', 'min samples leaf' and 'bootstrap'. We fit our grid search on the X and y training sets. Then we kept the best model (with the best estimators) and get the score with the X and y testing sets corresponding to the model with the best estimators. The result of the testing sets is 0.85 and is much higher and is a pretty good score.

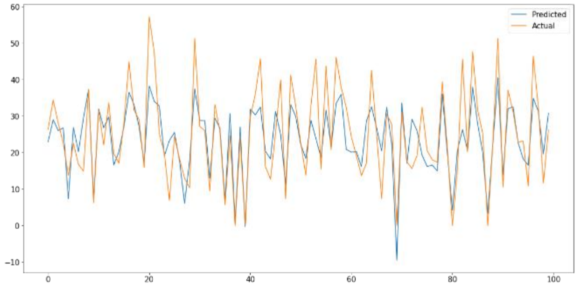


• **Extra Trees Regressor-** Finally we applied an extra trees regressor on our data. For this model, we also take the scaled X training and X testing sets to fit the model. Then we made a grid search on the model by tuning the feature 'max depth', 'n estimators', 'min samples split', 'min samples leaf' and 'bootstrap'. We fit our grid search on the X and y training sets. Then we kept the best model (with the best estimators) and got the score with the X and y testing sets. The score is pretty similar but a little bit higher than the random forest regressor. The best result of the testing sets that we got is 0.86 which is a pretty good score with a training score which is pretty similar to the testing score.

• **Polynomial Regression-** It is one of the most popular choices when we want to create a model for non-linearly separable data. It's like linear regression but uses the relationship between the variables X and y to find the best way to draw a curve that fits into the data points.

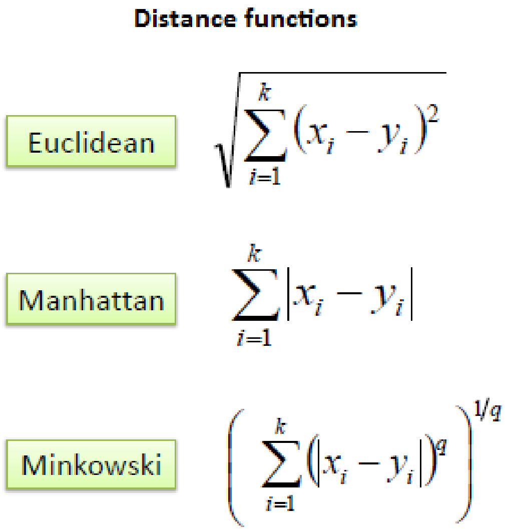


ElasticNet Regression- This Regression model trained with both L1 and L2 regularization. It's a hybrid of Lasso's and Ridge Regression techniques, therefore it's also well-suited for models showing heavy multicollinearity(heavy correlation of features with each other)



KNN Regression

A simple implementation of KNN regression is to calculate the average of the numerical target of the K nearest neighbors. Another approach uses an inverse distance weighted average of the K nearest neighbors. KNN regression uses the same distance functions as KNN classification.



Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Gradient Boosting

Gradient boosting refers to a class of esemble machine learning algorithms that can be used for classification or regression problems. Here I am using three gradient boosting algorithms i.e Xtreme Boost and CatBoost.

$$\hat{x}_k^i = \frac{\sum_{x_j \in D_k} \mathbb{1}_{x_k^i = x_k^j} \cdot y_j + ap}{\sum_{x_j \in D_k} \mathbb{1}_{x_k^i = x_k^j} + a}; \; if \; D_k = \{x_j : \sigma(j) < \sigma(i)\}$$

$$h^t = \underset{h \epsilon H}{arg\ min}\ \mathbb{E}\mathcal{L}(y, F^{t-1} + h)$$

Dataset:

The data set consists of two spreadsheets - 1. train.csv, containing data to train the prediction algorithm and 2. test.csv, containing data to test the prediction algorithm. The data fields in the train.csv are enumerated below

- datetime - hourly date + timestamp
- season - 1 = spring, 2 = summer, 3 = fall, 4 = winter
- holiday - whether the day is considered a holiday
- FunctioningDay - whether the day is neither a weekend nor holiday
- weather -
    - 1: Clear, Few clouds, Partly cloudy, Partly cloudy
    - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp - temperature in Celsius
- atemp - "feels like" temperature in Celsius
- humidity - relative humidity
- windspeed - wind speed
- count - number of total rentals
- Snowfall
- Seasons
- Visibility

Technologies Used:

Python: Python is a high-level interpreted language that supports different platforms like Windows, Linux, Mac, Rasberry Pi, etc.

Google Colaboratory Notebook:
It is a powerful tool to script python for data analysis and can contain live code, descriptive texts, visualization which can be created and shared just like a document.

Mathplotlib:
It is a 2D based plotting package that provides required modules and functions. A developer can customize font properties, styles, axes properties, etc.

Pandas:
It is used for data analysis and manipulation. Pandas can convert data structures and dataset formats to data frames on which operations like loading data, rename attributes, mapping, crosstab, sub-data frames, plotting, etc. can be performed

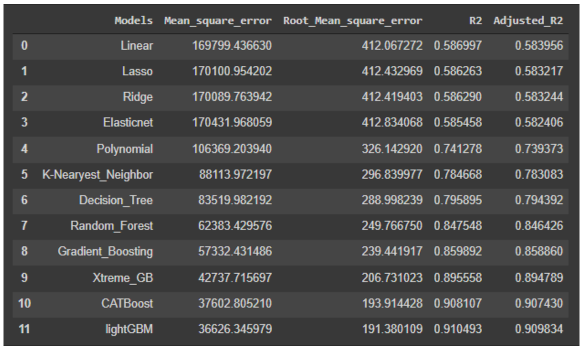
NumPy:
It provides structures for multiple dimensional array objects and tools for related operations. NumPy is usually used for high performance scientific computational tasks.

Scikit-learn:

Scikit-learn is a Python module integrating a wide range of state-of-the-art machine learning algorithms for medium-scale supervised and unsupervised problems. This package focuses on bringing machine learning to non-specialists using a general-purpose high-level language.

RESULTS AND DISCUSSIONS

The best score we got is 0.908 for the CatBoost after having applied a grid search on it. The representation below shows the order of the models's scores. But after applied the grid search we got a 0.908 score.



	Models	Mean_square_error	Root_Mean_square_error	R2	Adjusted_R2
0	Linear	169799.436630	412.067272	0.586997	0.583956
1	Lasso	170100.954202	412.432969	0.586263	0.583217
2	Ridge	170089.763942	412.419403	0.586290	0.583244
3	Elasticnet	170431.968059	412.834068	0.585458	0.582406
4	Polynomial	106369.203940	326.142920	0.741278	0.739373
5	K-Nearyest_Neighbor	88113.972197	296.839977	0.784668	0.783083
6	Decision_Tree	83519.982192	288.998239	0.795895	0.794392
7	Random_Forest	62383.429576	249.766750	0.847548	0.846426
8	Gradient_Boosting	57332.431486	239.441917	0.859892	0.858860
9	Xtreme_GB	42737.715697	206.731023	0.895558	0.894789
10	CATBoost	37602.805210	193.914428	0.908107	0.907430
11	lightGBM	36626.345979	191.380109	0.910493	0.909834

Regression Parameters

Before the outlier treatment we obtained a Regression Model on the dataset containing 8760 observations. The parameters saw a slight improvement after Outlier Treatment and Correlation analysis.

- To refine our model, we cleaned a few outliers to obtain an efficient Regression line. Rental Orders above 2500 were removed from our dataset, owing to the scattered distribution leading to noisy data.
- Similarly, Rainfall, Snowfall, Solar Radiation, and Wind Speed entries exceeding 10mm, 4cm, 3.5 MJ/m2, and 5m/s respectively were removed from our dataset, too. Our final dataset comprises of 8567 observations.
- Python Value is the coefficient between the Predicted and Observed values of the dependent variable. 0.753 suggests a high positive correlation between the Original and Forecasted Rental Bike count.
- Python-Square Value is the goodness-of-fit and a statistical measure of how close the data are fitted to the regression line. The table value of 0.567 suggests that our linear regression model able to determine 56.7% of changes in the Rental Bike Count.
- Adjusted Python-squared compares the explanatory power of regression models that contain different numbers of predictors. It calculates R-Square of only Independent Variables those are statistically significant.
- A minute difference between R-Square and Adjusted R-Square suggests all our Independent Variables being significant,

despite both values being on a relatively lower side.

- R-square change, which is just the improvement in Rsquare when the second predictor is added. The R-square change is tested with an F-test, which is referred to as the Fchange. A significant F-change means that the variables added in that step significantly improved the prediction.

Hypothesis Formation

1. H10: There is no relationship between Rental Bike Count and Independent Variables.
2. H11: There exists a relationship between Rental Bike Count and Independent Variable.
3. H20: There is no statistical significance between Rental Bike Count and Explanatory Variables
4. H21: There exists some statistical significance between Dependent Variables and Explanatory Variables and not all coefficients are Zero.

8. Conclusion:

We calculated a regressionmodels, which clearly shows that,

- Functioning Hour is a significant negative predictor (estimate = -932.492).
- It also shows that Seasons compared to humidity is a significant negative predictor (estimate = -15.050) and ¬ Snowfall compared to Wind speed is a significant positive predictor (estimate = -12.966) of bike rentals.

- The Visibility and Temperature was not included because its p-value (0.580) & (0.528) respectively exceeded alpha value of 0.05 making it insignificant.

- Regression models with low R-squared values can be perfectly good models for several reasons.

- Some fields of study have an inherently greater amount of unexplainable variation. In these areas, your R2 values are bound to be lower. For example, studies that try to explain human behaviour generally have R2 values less than 50%. People are just harder to predict than things like physical processes.

- Fortunately, if you have a low R-squared value but the independent variables are statistically significant, you can still draw important conclusions about the relationships between the variables. Statistically significant coefficients continue to represent the mean change in the dependent variable given a one-unit shift in the independent variable. Clearly, being able to draw conclusions like this is vital.

- As observed in our case, 0.56 is a relatively low value but statistical significance aids us to understand the factors affecting the Rental Bike Count better.

- To extract better results and patterns from the datasets, advanced algorithms like Random Forest, Decision Tree and CatBoost could be implemented.

References-

A. Sathishkumar V E, Jangwoo Park, Yongyun Cho (2020), ‘Using data mining techniques for bike sharing demand prediction in metropolitan city’, The International Journal for the Computer and Telecommunications Industry

B. Sathishkumar V E and Yongyun Cho (2020), ‘A rulebased model for Seoul Bike sharing demand predictiusing weather data’, European Journal of Remote Sensing.

C. ‘**Bike Sharing:** It’s about the community’, Cycling Industries Europe - https://cyclingindustries.com/fileadmin/content/docume nts/170707_Benefits_of_Bike_Sharing_UK_AI.pdf

D. Seoul Bike Sharing Demand Data Set, UCI Machine Learning Repository - https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Shar ing+Demand#

E. Hadi Fanaee-T, and Joao Gama (2013), 'Event labeling combining ensemble detectors and background knowledge', Progress in Artificial Intelligence.

F. Bike-Sharing Service Market Market Size 2021-2026, MarketWatch - https://www.marketwatch.com/pressrelease/bike-sharing-service-market-market-size-2021- 2026-comprehensive-study-development-statusopportunities-future-plans-competitive-landscape-andgrowth-2021-01-11

G. Data Source :http://data.seoul.go.kr/