IE 7300 Statistical Learning for Engineering

Bike Rental Customer Prediction using ML

Group Number : 8

- Saloni Bhutada (NEU ID: 002191872)
- Aman Maheshwari (NEU ID: 001008819)

1. Abstract:

The rental bike-sharing dataset is a valuable source of information for understanding the patterns and characteristics of bike-sharing systems.

The data is generated by bike-sharing programs around the world and includes information about the date, season, year, month, hour, holiday, weather, temperature, humidity, wind speed, and rental counts of bikes.

In this project, we aim to use regression analysis to predict the number of bike rentals based on these attributes. This data science project aims to analyze the Rental Bike Sharing Dataset from 2011 and 2012 to develop a regression model that can accurately predict the number of bike rentals based on the given attributes. Specifically, we will explore the relationships between rental counts and different variables, such as temperature, weather, and time of day, to build a regression model that can accurately predict rental counts.

The project will involve data cleaning, exploratory data analysis, feature engineering, and model development using regression techniques.

This project will provide insights into the factors that influence bike sharing usage and help bike sharing companies to optimize their operations and services the ultimate goal is to build a robust and accurate regression model that can be used to forecast bike rentals in the future, thereby enabling rental bike companies to optimize their inventory and pricing strategies.

2. Introduction:

Advancements in machine learning have made it easier for us to predict any business outcome with ease. Bike sharing systems have become an increasingly popular mode of transportation in urban areas around the world. With the growth of these systems comes a wealth of data that can be analyzed to gain insights into travel patterns and behaviors. In this project, we will analyze the Rental Bike Sharing Dataset, which includes data from bike sharing systems in 2011 and 2012. This dataset contains a range of variables related to weather, holidays, and bike usage, providing a rich source of information for analysis.

With respect to machine learning, predicting the number of bike rentals is a regression task and this problem was solved using different models such as Linear Regression (Closed Form Solution), Linear Regression (Gradient Descent), Linear Regression (With L1 Regularization), Linear Regression (With L2 Regularization) or KNN Regressor.

Following is the roadmap we have followed in the implementation of this project:

The information included in the data created by bike sharing systems is extensive and may be examined to get insights into urban mobility patterns, travel behavior, and transit demand. This information may be utilized to improve bike sharing systems and to influence transportation planning and policy decisions.

- Exploratory Data Analysis (EDA):
  Plotting different graphs to visualize the total users count and other features, distribution of other features and multivariate analysis.
- Correlation Matrix:
  Plotting heat map to visualize the correlation between each of the features with themselves as well as with the value to be predicted.
- Data wrangling:
  Conversion of data into numeric data types to be able to perform operations on them.
  Renaming columns to make sense.
  Dropping unnecessary columns.
  One-hot/binary encoding on the nominal features of the dataset - education and marital status
- Scaling of data - Using standardization to limit the values between [1, -1] and make their mean approximately zero.
- Train-Test split:
  Data has been split into two using the sklearn_model selection train_test_split function with a training size of 0.7 i.e., the training set is 70% and a test size of 0.3 i.e., the testing set is 30% of the entire data for Linear Regression and KNN Regression
- Computation of results using inbuilt packages to see which model performs best:
- Linear Regression (Closed Form Solution, Gradient Descent, Ridge Regression and Lasso Regression) and KNN Regression.
- Model building by writing own code:
  Custom Linear Regression model
  KNN Regression model

3. Data Description:

The information included in the data created by bike sharing systems is extensive and may be examined to get insights on urban mobility patterns, travel behavior, and transit demand. This information may be utilized to improve bike sharing systems and to influence transportation planning and policy decisions.

a. The data shows bike rentals from the Capital Bikeshare system, corresponding to years 2011 and 2012 in Washington D.C.

b. Analyzing bike sharing data may assist in identifying popular routes and sites where bikes are regularly leased and returned, informing choices on where to position additional bike stations or upgrade bike infrastructure. It can also assist in understanding how various conditions such as weather, time of day, and day of the week influence travel behavior and transportation demand.

c. Data from bike sharing services may be combined with data from other sources, such as transit and traffic, to get a more comprehensive picture of urban mobility patterns and to influence integrated transportation planning and policy choices.

d. Researchers and politicians interested in improving urban mobility and sustainability can benefit from bike sharing programs as a useful source of data.

There are a total of 17389 instances in the dataset. There are 16 attributes in total, including the target variable as count, which indicates depending upon the various factors, what would be the bike rent count in every month, season or year.

instant:	record index
dteday:	date
season:	season (1: winter, 2:spring, 3:summer, 4:fall)
yr:	year (0: 2011, 1:2012)
mnth:	month (1 to 12)
hr:	hour (0 to 23)
holiday:	weather day is holiday or not
weekday:	day of the week
working day:	if day is neither weekend nor holiday is 1, otherwise is 0.
1:00	Clear, Few clouds, Partly cloudy, Partly cloudy
2:00	Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
3:00	Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
4:00	Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
temp:	Normalized temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min = -8, t_max=+39 (only in hourly scale)
atemp:	Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min= -16, t_max=+50 (only in hourly scale)
hum:	Normalized humidity. The values are divided to 100 (max)
windspeed:	Normalized wind speed. The values are divided to 67 (max)
casual:	count of casual users
registered:	count of registered users
cnt:	count of total rental bikes including both casual and registered

4. Dataset Information and Null Values Check:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17379 entries, 0 to 17378
Data columns (total 17 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   instant     17379 non-null  int64
 1   dteday      17379 non-null  object
 2   season      17379 non-null  int64
 3   yr          17379 non-null  int64
 4   mnth        17379 non-null  int64
 5   hr          17379 non-null  int64
 6   holiday     17379 non-null  int64
 7   weekday     17379 non-null  int64
 8   workingday  17379 non-null  int64
 9   weathersit  17379 non-null  int64
 10  temp        17379 non-null  float64
 11  atemp       17379 non-null  float64
 12  hum         17379 non-null  float64
 13  windspeed   17379 non-null  float64
 14  casual      17379 non-null  int64
 15  registered  17379 non-null  int64
 16  cnt         17379 non-null  int64
dtypes: float64(4), int64(12), object(1)
memory usage: 2.3+ MB
```

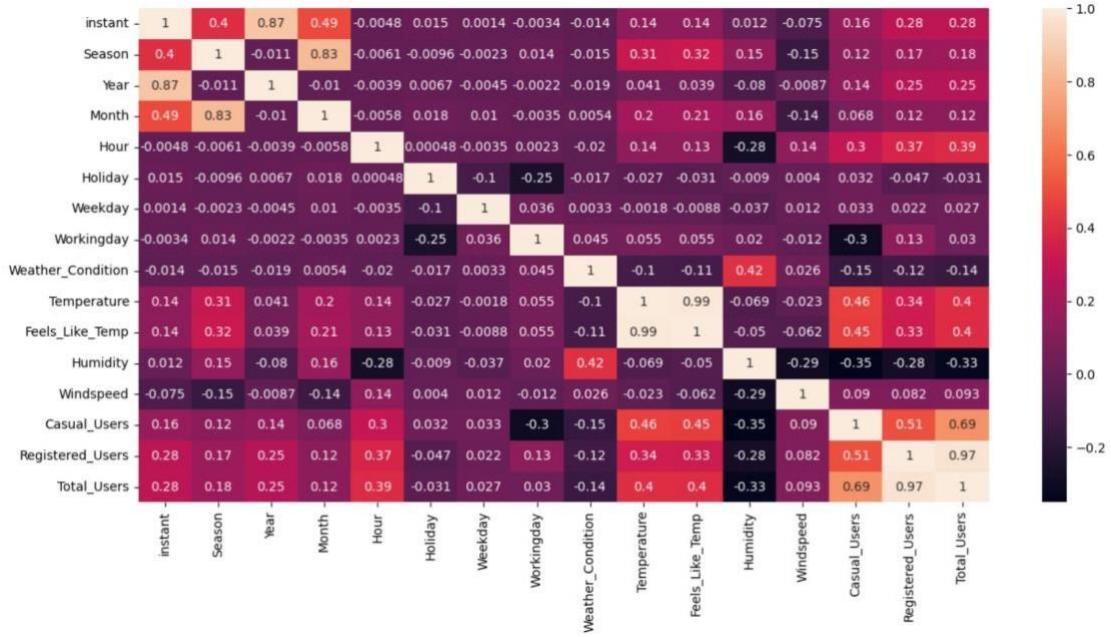
Data Frame Information

```
df.isnull().sum()
```

```
instant       0
dteday        0
season        0
yr            0
mnth          0
hr            0
holiday       0
weekday       0
workingday    0
weathersit    0
temp          0
atemp         0
hum           0
windspeed     0
casual        0
registered    0
cnt           0
dtype: int64
```

Null Values Check

Data frame heatmap to visualize the correlation between all the features:



a. From the above heatmap we can say that the features like Season, Hour, Temperature have a positive correlation with the target variable 'Total_Users'.

b. After analyzing the data we found out that the two features Casual_Users and Registered_Users have multi collinearity and we should not have that for multivariate linear regression. The two columns are simply the addition of our target variable therefore decided to drop the columns.

c. We can also observe a negative correlation between Humidity and the target variable.

Problem Solving Methods:

We have used two regressors comprising of Linear Regression (Closed Form Solution, Gradient Descent, Ridge Regression and Lasso Regression) and KNN Regressor.

a. Linear Regression:

Linear regression is a statistical approach used to establish a relationship between a dependent variable and one or more independent variables. The objective of linear regression is to find the line of best fit that describes the relationship between the variables. There are different types of linear regression, each with its own approach to finding the line of best fit. In this response, we will discuss four types of linear regression: closed form solution, gradient descent, ridge regression, and lasso regression.

Closed form solution:

The closed form solution is also known as the ordinary least squares method, which involves finding the line of best fit by minimizing the sum of squared errors between the predicted and actual values of the dependent variable. The formula for calculating the coefficients of the line of best fit is given by: **β = (X^T X)^(-1) X^T Y** where β is a vector of coefficients, X is a matrix of independent variables, Y is a vector of dependent variable values.

Gradient Descent:

Gradient descent is an iterative optimization algorithm that seeks to minimize the cost function by adjusting the coefficients of the line of best fit in small steps. The cost function is defined as the difference between the predicted and actual values of the dependent variable. The formula for gradient descent is given by: **βj = βj - α * ∂J(β) / ∂ βj**
where βj is the jth coefficient of the line of best fit, α is the learning rate, and J(β) is the cost function.

Ridge Regression:

Ridge regression is a regularized version of linear regression that adds a penalty term to the cost function to prevent overfitting. The penalty term is proportional to the square of the coefficients of the line of best fit.
The formula for ridge regression is given by: **β = (X^T X + λI)^(-1) X^T Y**
where λ is the regularization parameter and I is the identity matrix.

Lasso Regression:

Lasso regression is also a regularized version of linear regression that adds a penalty term to the cost function to prevent overfitting. The penalty term is proportional to the absolute value of the coefficients of the line of best fit.
The formula for lasso regression is given by: **β = argmin(|| Y - Xβ ||^2 + λ || β ||)**
where λ is the regularization parameter.

b. KNN Regression:

K-Nearest Neighbors (KNN) regression is a non-parametric algorithm used for predicting continuous output variables. It is based on the idea that data points that are close to each other in the feature space will have similar output values. KNN regression is a simple algorithm that makes predictions by finding the K nearest neighbors to a given test point, and then taking the average of their output values as the predicted output for the test point.

The main steps of KNN regression are as follows:
Determine the value of K:
The first step in KNN regression is to determine the value of K, which represents the number of neighbors to consider when making a prediction. This value can be determined using cross-validation or other techniques.

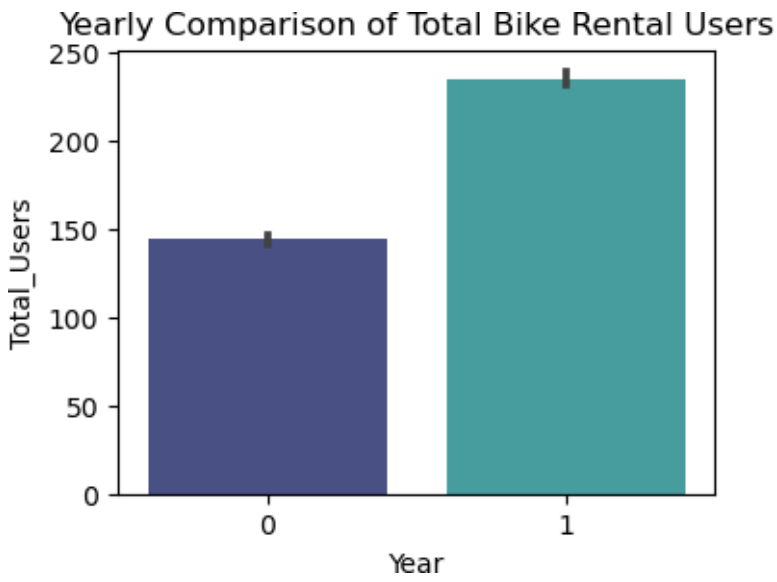
Calculate the distance between the test point and each training point:
The distance between the test point and each training point is calculated using a distance metric, such as Euclidean distance or Manhattan distance.

Identify the K nearest neighbors:
The K nearest neighbors to the test point are identified based on their distance from the test point.

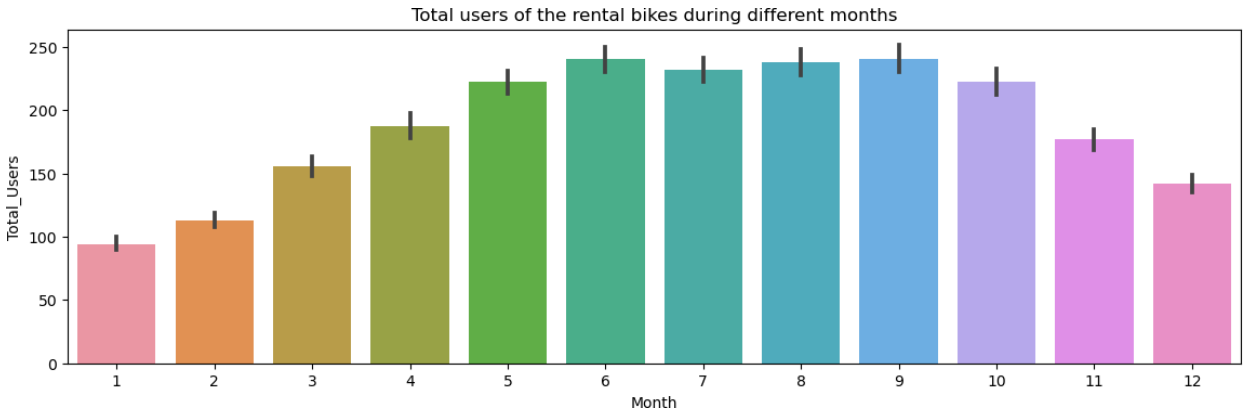
Calculate the predicted output value:
The predicted output value for the test point is calculated by taking the average of the output values of the K nearest neighbors.

Overall, KNN regression is a simple and effective algorithm for predicting continuous output variables. However, it can be sensitive to the value of K and the choice of distance metric, so careful tuning is necessary to achieve optimal performance.

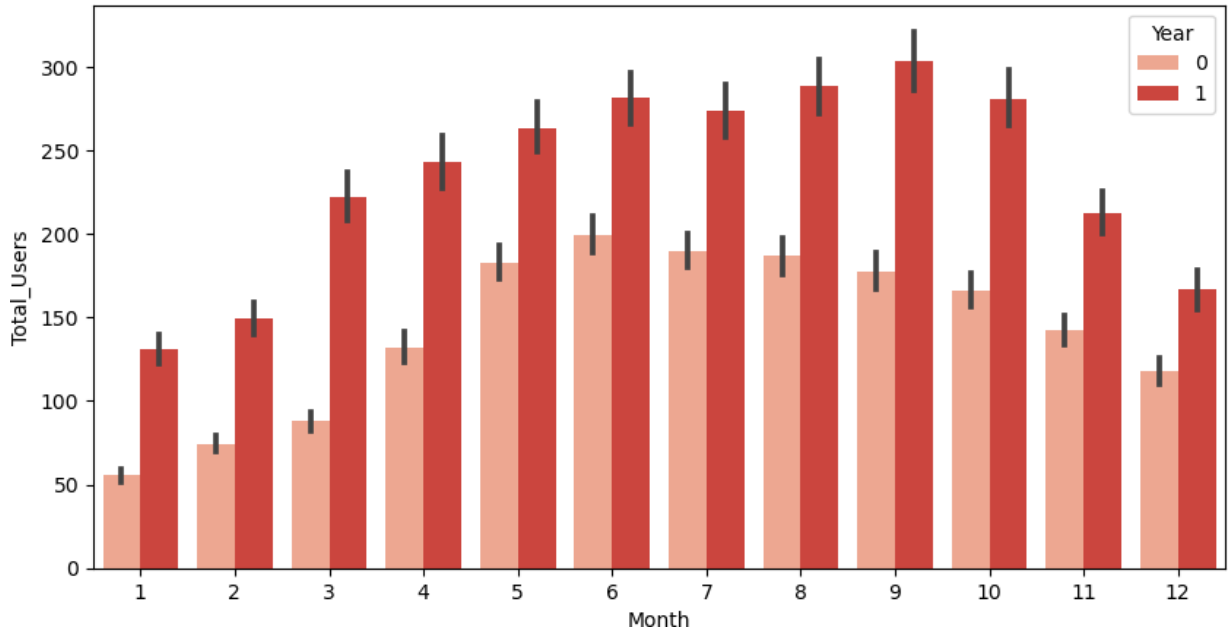
5. Exploratory Data Analysis:



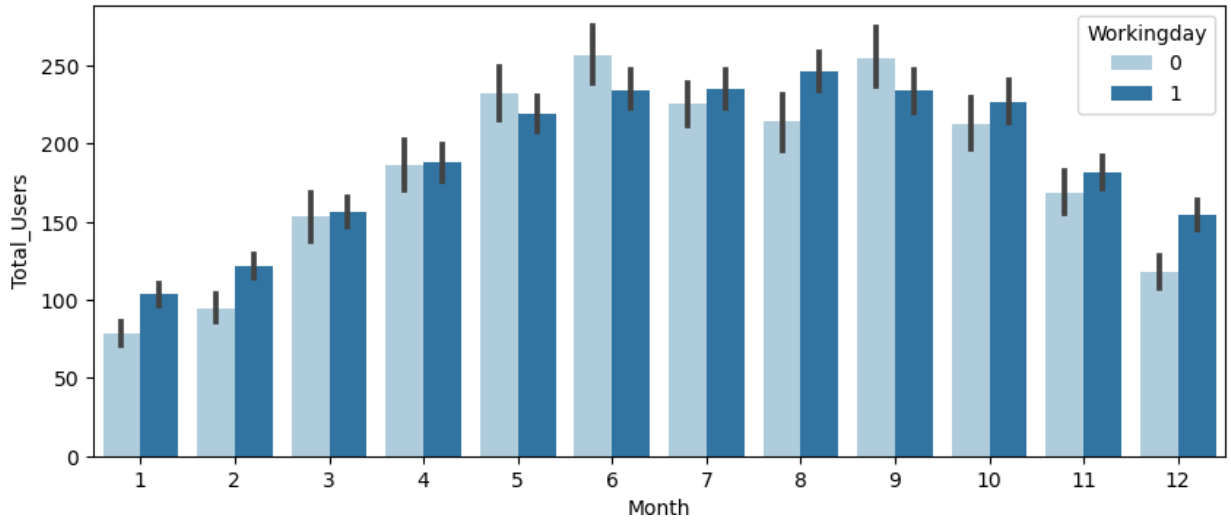
The above figure shows us that the total users count distribution in the year 2011 and 2012. Year 2012 has more users than the year 2011 by around 40%



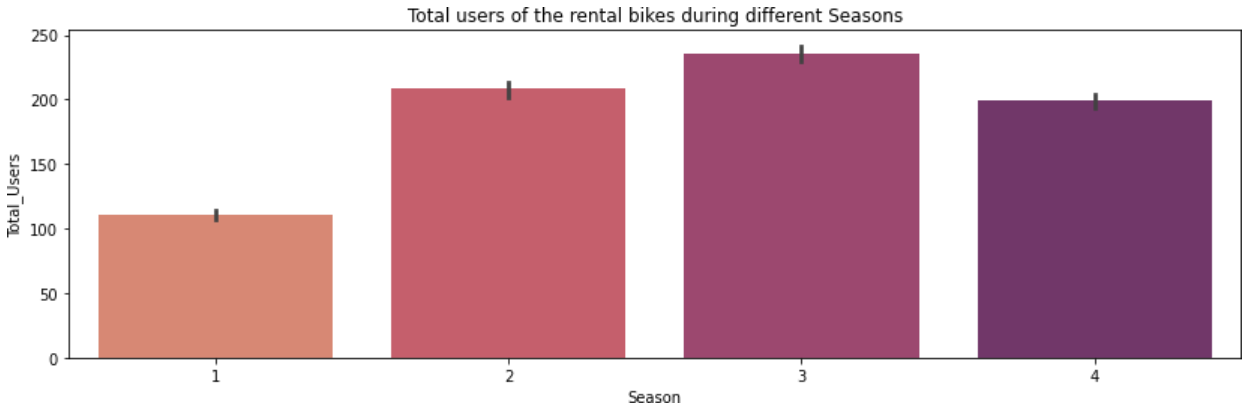
The above figure shows us the total users count distribution across all the months.



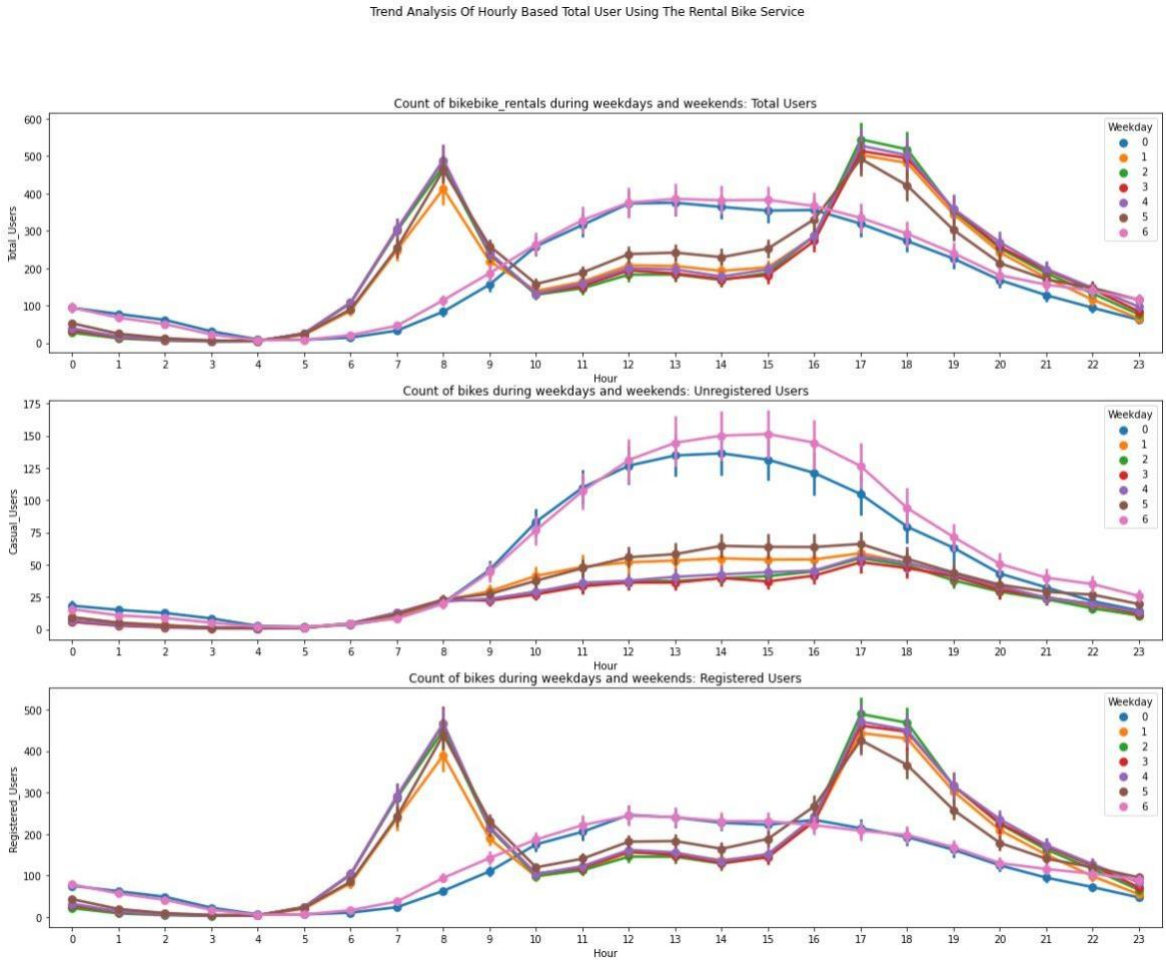
The above figure shows us the total users count for all the months compared to the year 2011 and 2012 and we can see that the total users count in the year 2012 has increased for every month as compared to the total users count in the year 2011.



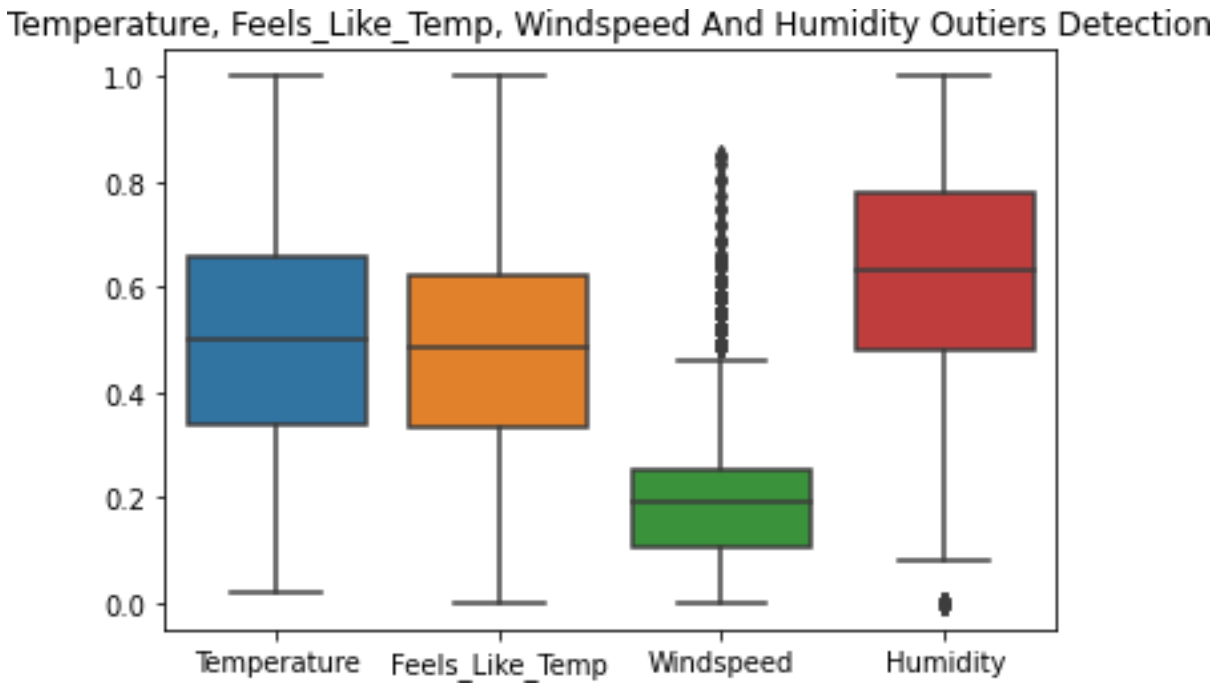
The above figure shows us the total users count for all the months compared to the working day and weekdays.



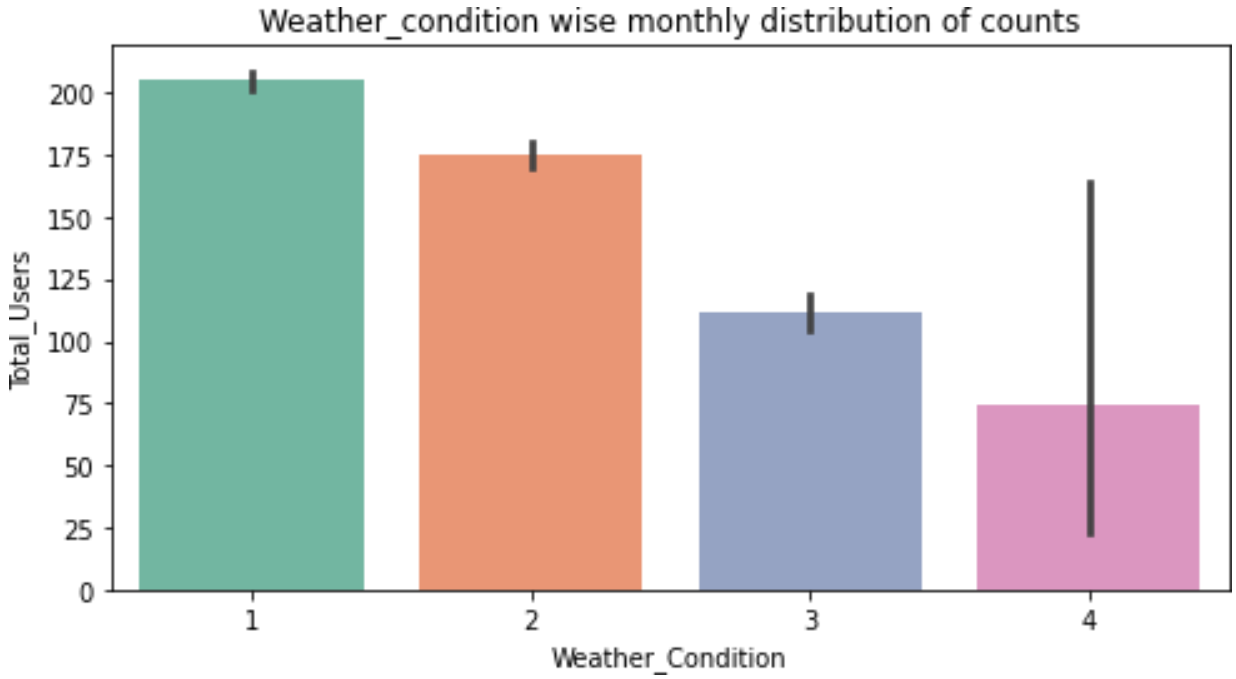
The above figure shows the total users count compared to the different seasons of the year and we can see that for the summer season total users count rises of all. (1: Winter 2: Spring 3: Summer 4: Fall)



From the above figure we can see the hourly trend analysis compared to all the days of the week. It can be clearly seen that the usage of the rental bikes by the registered users is more on the weekdays during the office commute times and the usage of casual users is more on the weekends during afternoon.



From the above box plot, we can observed that no outliers are present in normalized temp but few outliers are present in normalized windspeed and humidity variable.



Weather Conditions:

c. Clear, Few clouds, Partly cloudy, Partly cloudy

d. Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist

e. Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds

f. Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog

6. Results and Discussions:

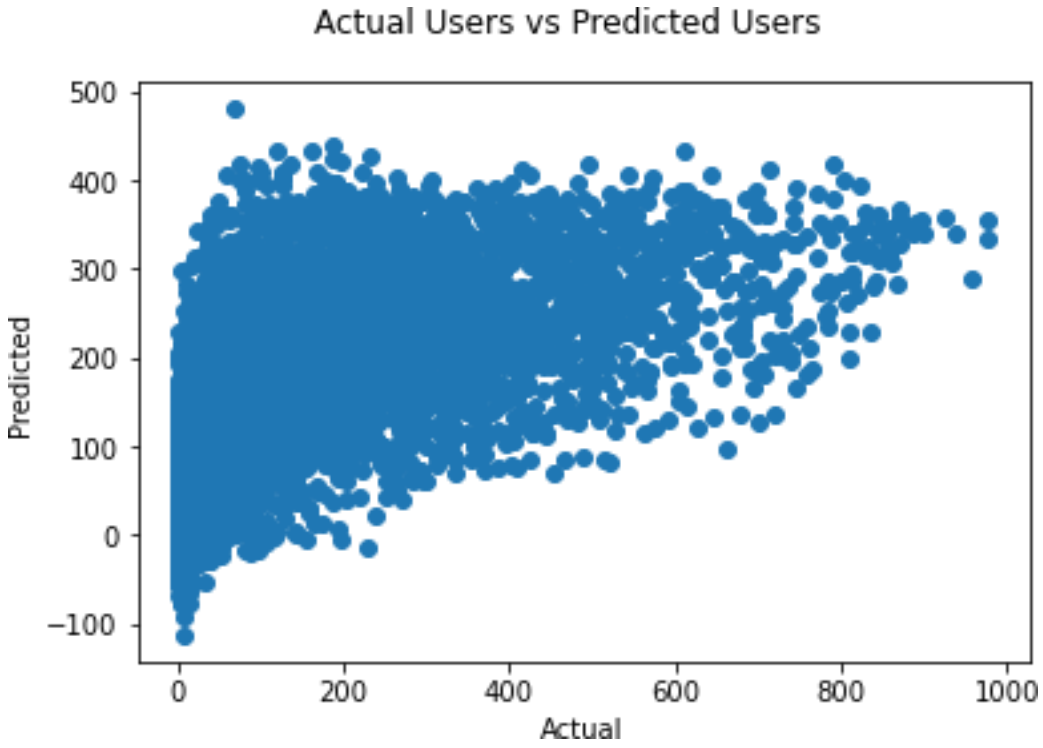
Linear Regression:

a. Linear Regression Closed Form Solution:
The RMSE computed from the Linear Regression with using Normal Equation is 140.76 and R2 Score is 0.40

b. Linear Regression with L2 Regularization:
The RMSE computed from the Linear Regression with using L2 Regularization is 140.76 and R2 Score is 0.40

c. Gradient Descent without L2 Regularization:
The RMSE computed from the Linear Regression with using Normal Equation is 165.88 and R2 Score is 0.16

d. Gradient Descent with L2 Regularization:
The RMSE computed from the Linear Regression with using L2 Regularization is 165.88 and R2 Score is 0.16

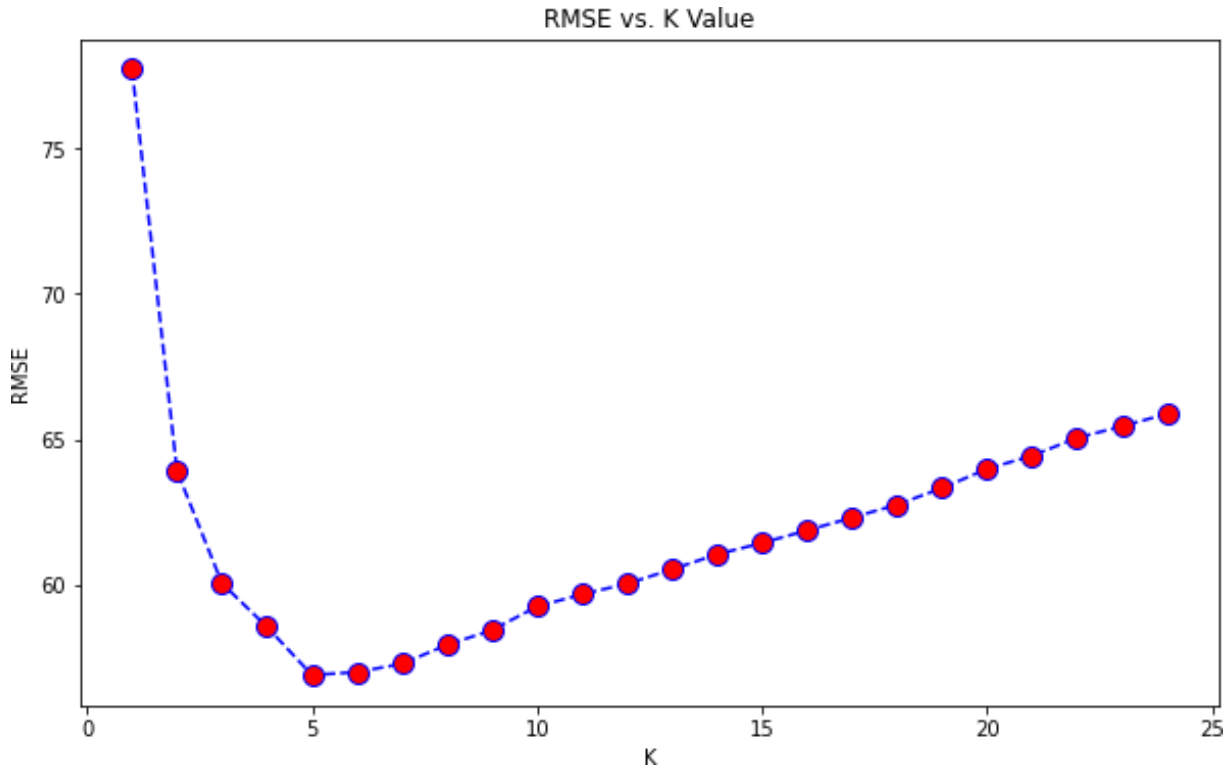


The above plot gives us an idea about how the predicted user counts look like with actual user counts.

Predictions after implementing Linear Regression with L2 Regularization Regressor:

	Season	Year	Month	Hour	Holiday	Weekday	Workingday	Weather_Condition	0	Total_Actual_Users	Total_Predicted_Users
0	1	0	1	0	0	6	0	1	-1.931027	7.0	58.150545
1	1	0	1	1	0	6	0	1	-1.904782	5.0	78.170780
2	1	0	1	2	0	6	0	1	-1.904782	743.0	370.588015
3	1	0	1	3	0	6	0	1	-1.703064	208.0	194.130849
4	1	0	1	4	0	6	0	1	-1.703064	333.0	315.657940

KNN Regression:



The above plot gives us the K values over the range of 0 to 25 and we can see approximately which K value gives us the lowest RMSE value.

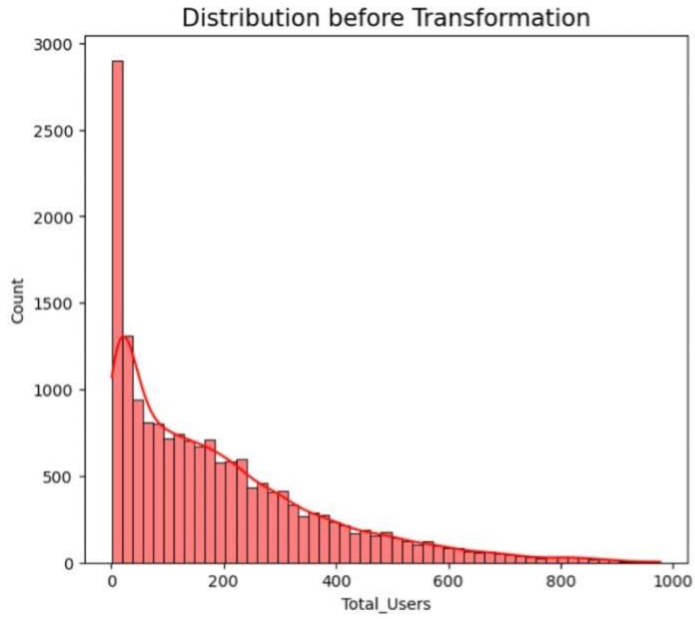
The RMSE computed from the KNN Regression is 56.90 and R2 Score is 0.88

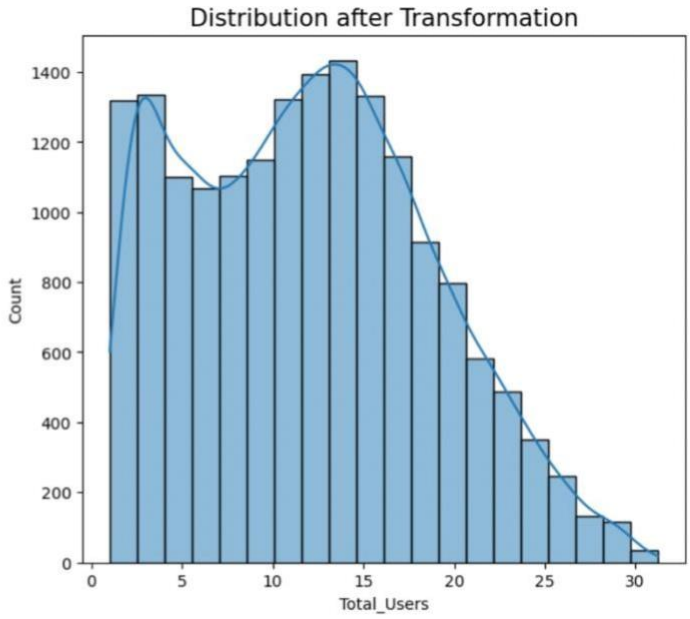
Predictions after implementing KNN Regressor

	Season	Year	Month	Hour	Holiday	Weekday	Workingday	Weather_Condition	0	Total_Actual_Users	Total_Predicted_Users
0	1	0	1	0	0	6	0	1	-1.931027	7.0	12.666667
1	1	0	1	1	0	6	0	1	-1.904782	5.0	9.166667
2	1	0	1	2	0	6	0	1	-1.904782	743.0	589.833333
3	1	0	1	3	0	6	0	1	-1.703064	208.0	160.500000
4	1	0	1	4	0	6	0	1	-1.703064	333.0	283.166667

7. Model Improvement:

a. Improving the skewness through the data






```
Skewness was 1.28 before & is 0.29 after Skrt transformation.
```

The RMSE computed from the Linear Regression with using L2 Regularization and after reducing skewness is 4.82 and R2 Score is 0.48

The RMSE computed from the KNN Regression is 1.85 and R2 Score is 0.91

Holiday	Weekday	Workingday	Weather_Condition	Temperature	Humidity	Windspeed	Total_Users_Sqrt_Transformed	Total_Actual_Users	Total_Predicted_Users
0	6	0	1	0.24	0.81	0.0	4.000000	7.0	10.774258
0	6	0	1	0.22	0.80	0.0	6.324555	5.0	7.634611
0	6	0	1	0.22	0.80	0.0	5.656854	743.0	566.510033
0	6	0	1	0.24	0.75	0.0	3.605551	208.0	154.565136
0	6	0	1	0.24	0.75	0.0	1.000000	333.0	306.481407

b. Introduced PCA:

We have implemented PCA to do the dimensionality Reduction of our data. After performing the PCA we reduced three features to one and we again found out the results.
The RMSE computed from the Linear Regression with using L2 Regularization is 151.15 and R2 Score is 0.30
The RMSE computed from the KNN Regression is 64.83 and R2 Score is 0.84

8. Conclusion:

Based on the number of features and the above model implementations after preprocessing, it seems reasonable to consider the KNN Regressor as a baseline model for the use case at hand

- Although improvements were attempted through preprocessing and feature engineering, the results did not show a significant impact on the model's performance
- Additionally, attempts to treat the outliers did not result in better evaluation metrics or predictions, indicating that the outliers may contain valuable information for the model
- Therefore, it may be worthwhile to explore other regression models and consider additional approaches to improve the model's performance, such as hyperparameter tuning, ensemble methods, etc