Capstone Project - 2

Supervised ML - Regression

Bike Sharing Demand Prediction

by:
Anas Malik

Project Flowchart

1. Business Problem Statement
2. Importing Dependencies and Loading dataset
3. Data Inspection
4. Data Wrangling
5. Exploratory Data Analysis
6. Data Pre-processing
7. Model Implementation
8. Cross Validation and Hyperparameter Tuning
9. Conclusion

Business Problem Statement

Nowadays in most of the urban cities, rental bikes have been implemented and the use of rental bikes is increasing day to day.

The most challenging part is to make rental bikes available to public at the right time. In cities, providing a stable supply of rental bikes becomes a major concern.

So the problem is to predict how many number of bikes are required at each hour for a stable supply of rental bikes.

Our main aim here is to solve this problem and predict the bike demands across different hours according to weather condition.

Dependencies and Loading Dataset

I have imported some basics dependencies which are as follows:

1. Numpy
2. Pandas
3. Matplotlib
4. Seaborn
5. Scikit-learn

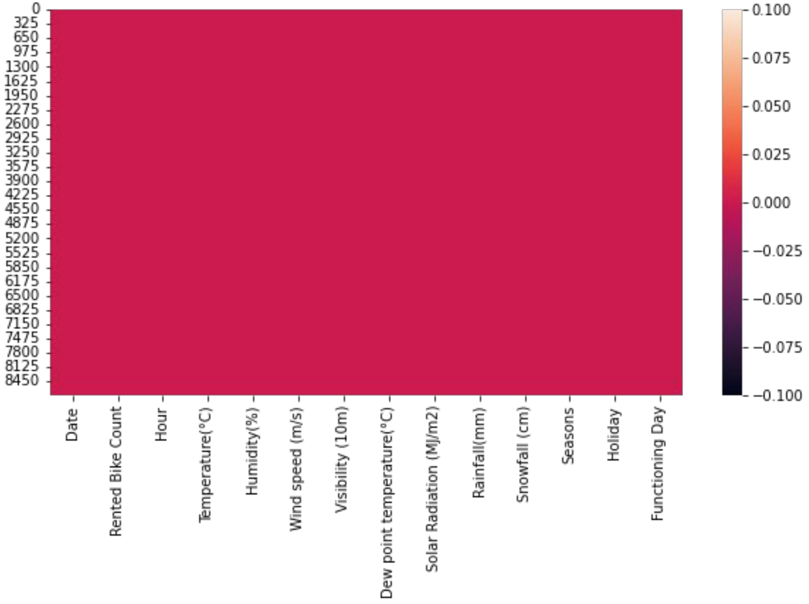
Then mounted google drive for dataset.

To load the dataset, I used pandas with .read_csv() function.

Note : The dataset for this project was Bike_Sharing_Demand_Prediction

Data Inspection

- The given dataset is of Seoul(Capital of South Korea).
- The dataset had 8760 rows and 14 columns.
- There was no missing values.
- There were no duplicated entries as well.
- The data present was a one year data of Seoul.
- Rented Bike Count was the target variable.



Column Description

- **Date -** Date of the day.
- **Rented Bike Count -** Number of bikes rented.
- **Hour -** Hour of day when bike rented.
- **Temperature -** Temperature of the day.
- **Humidity -** Humidity of the day in percentage.
- **Wind Speed -** Speed of the wind at that day.
- **Visibility -** Visibility of objects in meter.
- **Dew Point Temperature -** Temperature at the beginning of the day in celsius.
- **Solar Radiation -** Electromagnetic Rays from sum that day.
- **Rainfall -** Is that a raining day ?
- **Snowfall -** Is that a snowy day ?
- **Seasons -** what was the season ?
- **Holiday -** Is that a holiday ?
- **Functioning Day -** Is that a functioning day ?

Data Wrangling

- First of all, created a copy of main dataset.
- Rename some typical columns names such as “Rented Bike Count” to “Rented_Bike_Count”, “wind speed (m/s)” to “Wind_speed” etc.
- Convert the datatype of Date column from object to datetime.
- Created two new columns i.e., Month and day with the help of Date Column.
- Then created a new column Weekend with the help of day column. This column was about whether the day was a weekend day or not(Saturday or Sunday).
- Then dropped Date and day columns from the dataset.

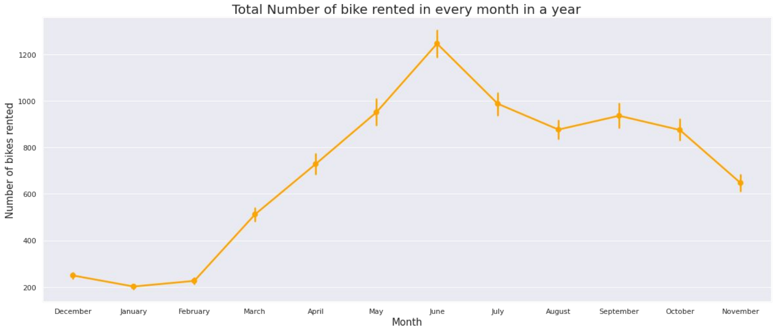
Exploratory Data Analysis

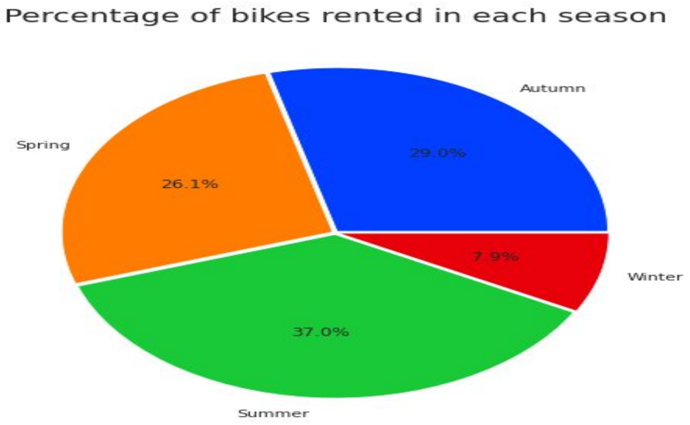
I Performed the EDA part in four steps which are as follows :

1. Univariate Analysis - Check distribution of each column.
2. Bivariate Analysis - Visualize each variables with target variable.
3. Multivariate Analysis - Visualize two or more variables with target variable.
4. Relationship of Columns.

EDA (contd.)

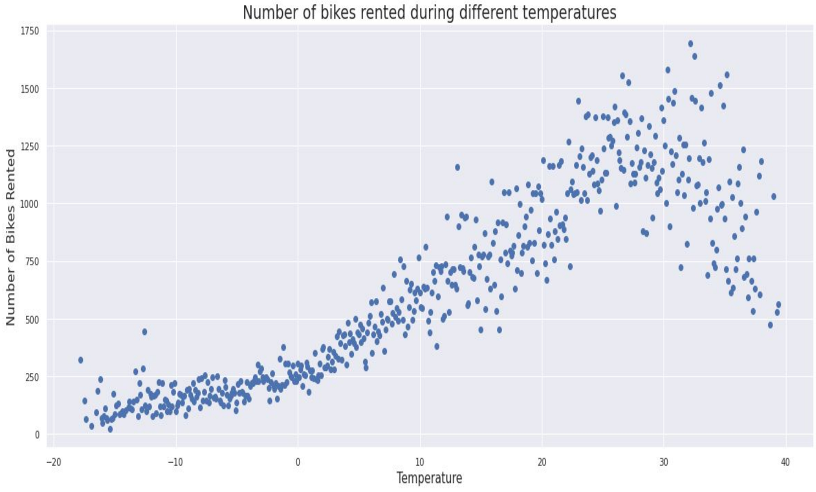
- The renting of bikes increases as summer season starts.
- The renting of bikes decreases as winter season starts.
- In summers, 37 % people prefer to rent a bike.
- In winters, only 7.9 % people prefer to rent a bike.

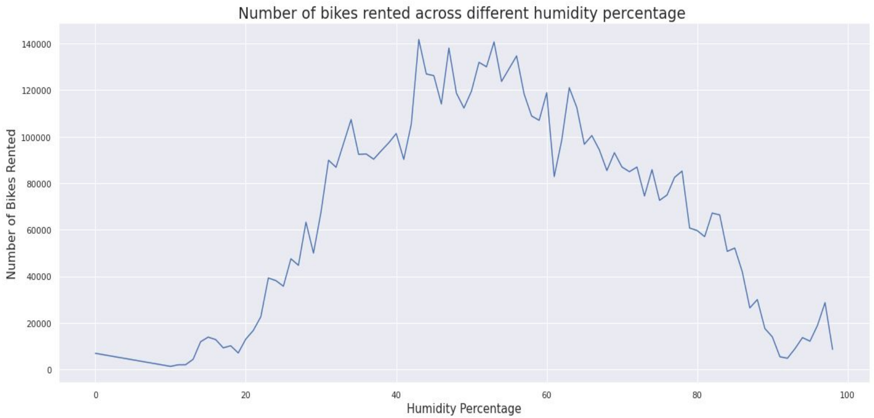




EDA (contd.)

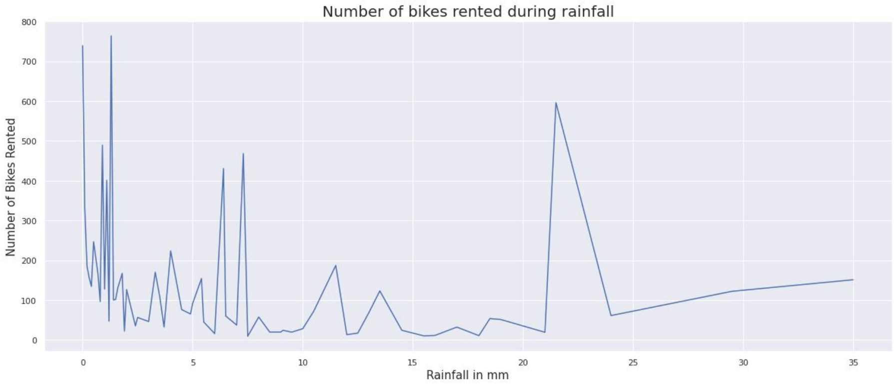
- Renting of a bike increases when the temperature of the day is in between 15 degree and 30 degree celsius.
- When humidity present in air is in between 30 to 70 %, chances of renting a bike increases.

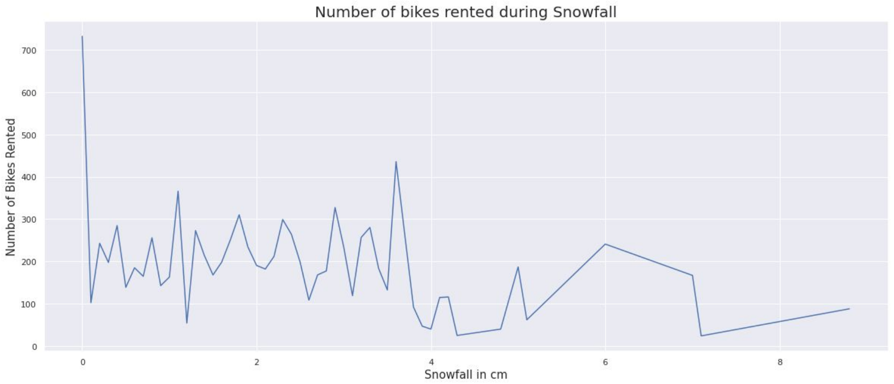




EDA (contd.)

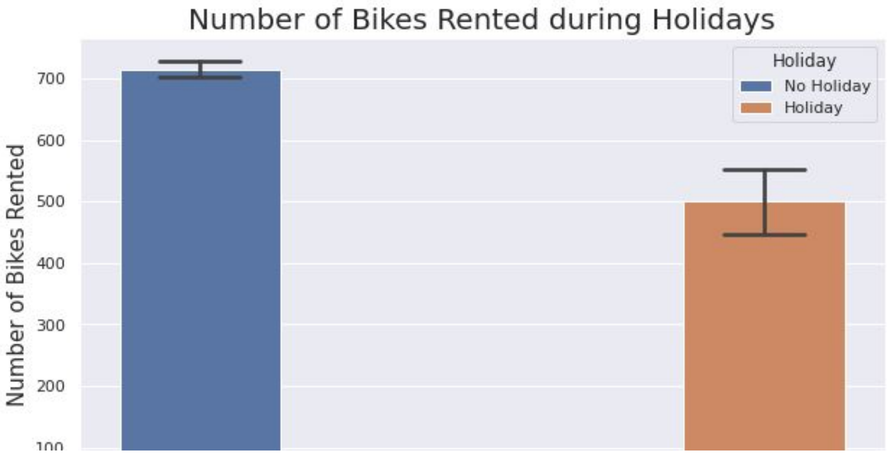
- Shiny sky is a sign of renting more number of bikes.
- When it rains, some people still consider to rent a bike.
- But when there is heavy snowfall, chances of renting a bike becomes less.

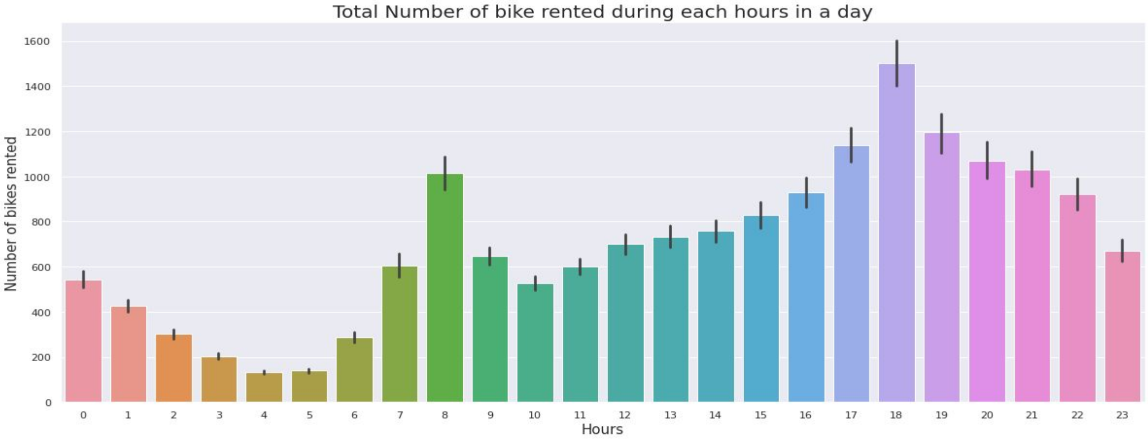




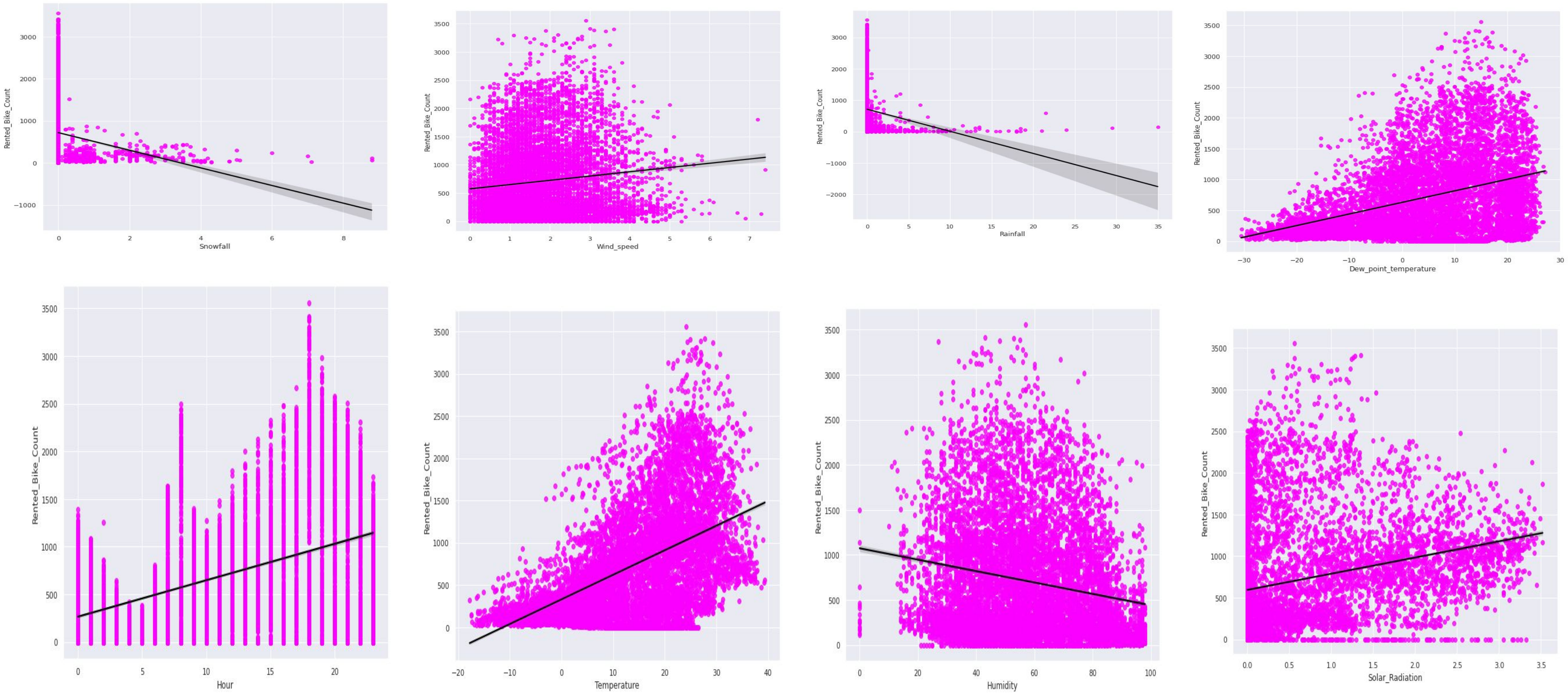
EDA (contd.)

- On a functioning day, people rent more number of bikes as compare on holidays.
- The most number of bikes are rented in morning and evening around 8AM and 6PM.





EDA (contd.)


Rented_Bike_Count
Snowfall
Wind_speed
Rainfall
Dew_point_temperature
Hour
Temperature
Humidity
Solar_Radiation

EDA (contd.)

- **Positively Correlated :**

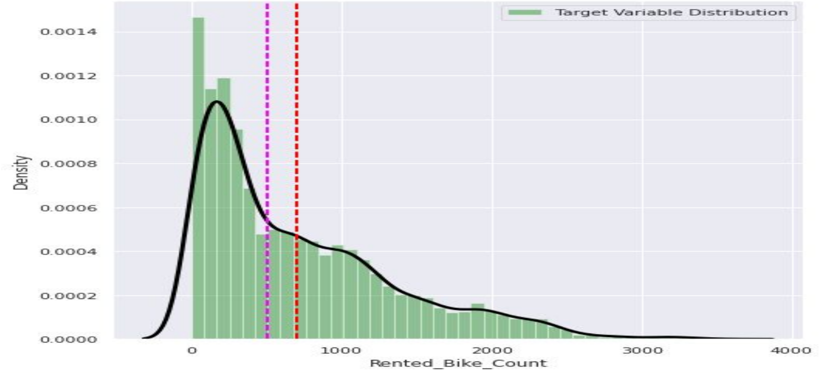
1. Solar_Radiation
2. Hour
3. Visibility
4. Wind_speed
5. Temperature
6. Dew_point_temperature

- **Negatively Correlated:**

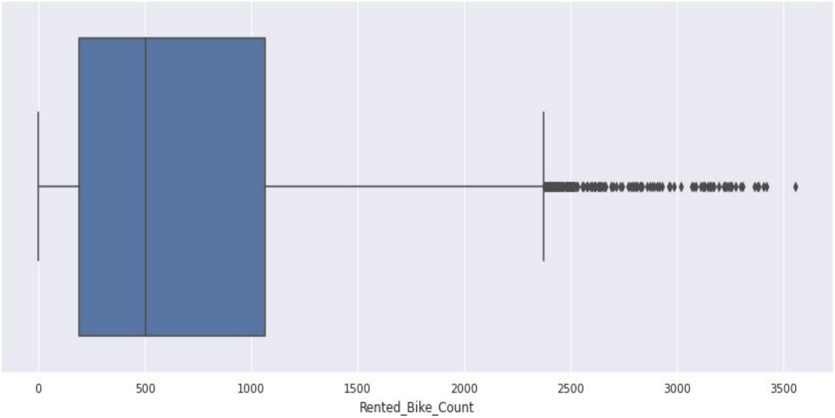
1. Rainfall
2. Humidity
3. Weekend
4. Snowfall

Data Pre-processing

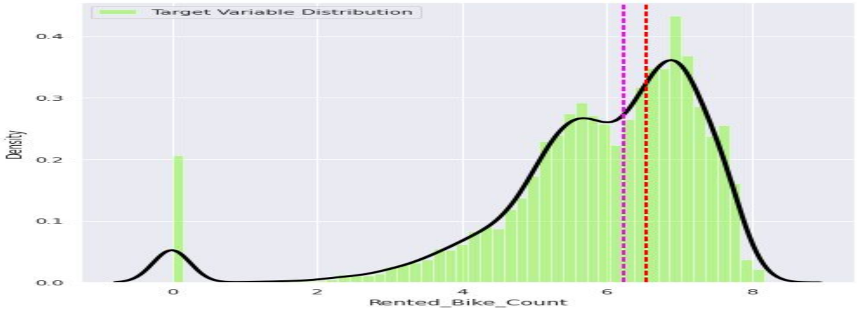
- Visualize the target variable distribution.



The reason behind the right skewness was that there were some outliers present in the data distribution.

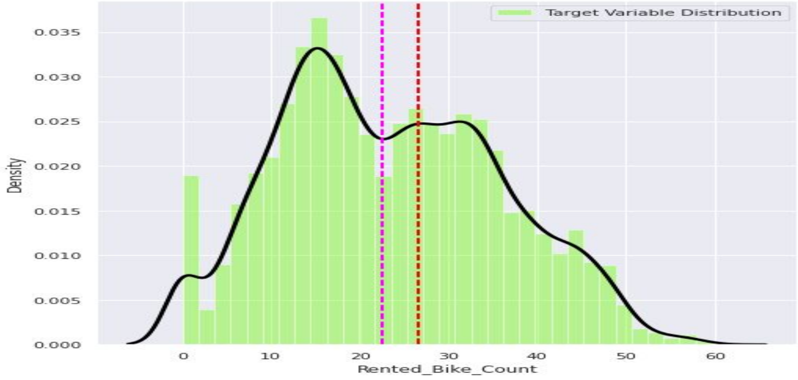


- Applied log transformation.

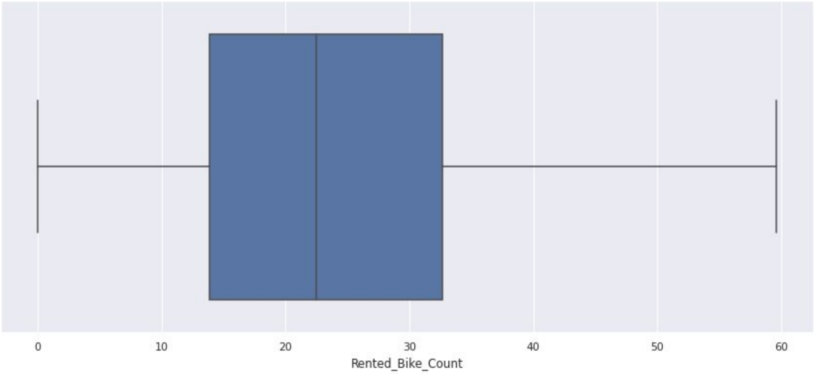


Data Pre-processing(contd.)

- Applied Square Root Transformation.



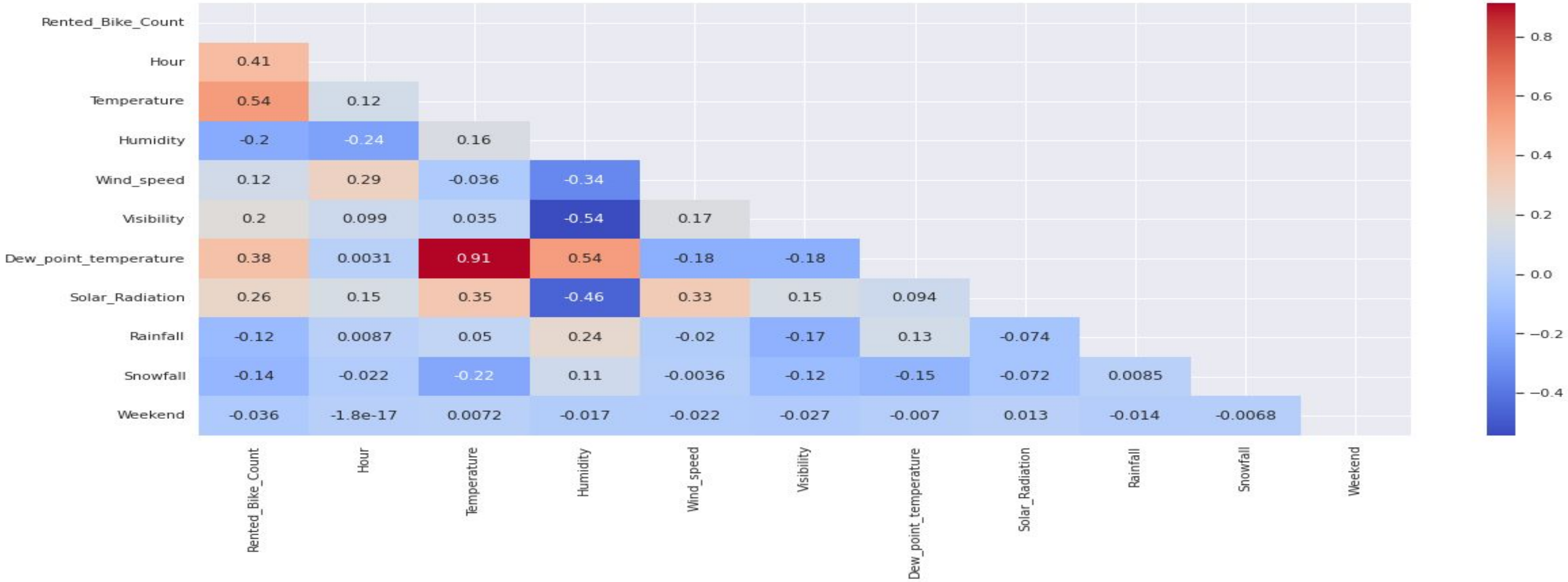
After applying square root transformation, our distribution becomes normally distributed.



Note: After applying Square Root Transformation, there was no outlier present in the distribution.

Correlation Between Variable

Dew_point_temperature and Temperature columns are highly correlated.



Removing Multicollinearity

- Similar to heatmap, variance inflation factor is also showing Dew_point_temperature and Temperature have very high “VIF”.
- As Dew_point_temperature doesn’t play much impact on renting a bike, I decided to drop this columns.

	Columns	VIF_value
0	Visibility	9.106191
1	Rainfall	1.081868
2	Wind_speed	4.809775
3	Hour	4.418398
4	Weekend	1.409388
5	Solar_Radiation	2.882383
6	Temperature	33.984042
7	Snowfall	1.120882
8	Dew_point_temperature	17.505235
9	Humidity	5.617480

Removing Multicollinearity(contd.)

- After removing the column, all the columns have less than 10 VIF value.

	Columns	VIF_value
0	Visibility	4.738121
1	Rainfall	1.079752
2	Wind_speed	4.608625
3	Hour	3.930173
4	Weekend	1.378871
5	Temperature	3.230140
6	Snowfall	1.120665
7	Solar_Radiation	2.254781
8	Humidity	5.016930

Green Sign to move for Feature Encoding.

Feature Encoding

- Firstly separated all the columns who are categorical in nature.
- All of the columns are nominal type, so I chose one hot encoding rather than label encoding.

Train_Test_Split

```
[ ] # Assigning the values in two part X and y
    X = data.drop(columns=['Rented_Bike_Count'], axis=1)
    y = np.sqrt(data['Rented_Bike_Count'])
```

```
[ ] X.shape, y.shape
```

```
((8760, 47), (8760,))
```

```
[ ] # Splitting the data into training set and test set
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=0)
```

```
[ ] print("Shape of train data : ", X_train.shape, y_train.shape)
    print("Shape of test data : ",X_test.shape, y_test.shape)
```

```
Shape of train data :  (6570, 47) (6570,)
Shape of test data :  (2190, 47) (2190,)
```

Feature Scaling

The difference between the data is high and low and for finding residuals which calculate distances like MSE, RMSE etc, it take more time to calculate. To reduce the time we use feature scaling.

There are two most popular type of feature scaling

1. Standardization - it standardize data with mean 0 and std. 1.
2. Normalization - it normalize the data in the range of 0 and 1.

I choose standardization for feature scaling with the help of StandardScalar() function of scikit-learn library.

Model Implementation

❖ Linear Regression :

Training Metrics :

Mean Square Error	35.07755090622306
Root Mean Square Error	5.922630404323999
Mean Absolute Error	4.474055591986692
R2 Score	0.7722099078993463

Testing Metrics :

Mean Square Error	33.27390585673638
Root Mean Square Error	5.7683538255499185
Mean Absolute Error	4.410100719860122
R2 Score	0.7722099078993463

- After Cross validation, getting the same result, therefore no overfitting is seen.

Model Implementation(contd.)

❖ Ridge Regression :

Evaluation Metrics

Mean Square Error	33.27663692231535
Root Mean Square Error	5.768590549026283
Mean Absolute Error	4.41028503807013
R2 Score	0.78934358054839

CV Evaluation Metrics

Mean Square Error	33.289046682617254
Root Mean Square Error	5.769666080685888
Mean Absolute Error	4.411305418312488
R2 Score	0.7892650210570101

- No Overfitting is seen.

Model Implementation(contd.)

❖ **Lasso Regression :**

Evaluation Metrics

Mean Square Error	33.87405315907444
Root Mean Square Error	5.820142022242623
Mean Absolute Error	4.457706017227814
R2 Score	0.785561660949612

CV Evaluation Metrics

Mean Square Error	33.87405315907444
Root Mean Square Error	5.820142022242623
Mean Absolute Error	4.457706017227814
R2 Score	0.785561660949612

- No Overfitting is seen.

Model Implementation(contd.)

❖ Elastic Net Regression :

Evaluation Metrics

Mean Square Error	34.14204279014603
Root Mean Square Error	5.843119268861969
Mean Absolute Error	4.4821264887616445
R2 Score	0.78386516330583

CV Evaluation Metrics

Mean Square Error	33.30551825703942
Root Mean Square Error	5.771093332899704
Mean Absolute Error	4.412070056234412
R2 Score	0.7891607484137558

- No Overfitting is seen.

Model Implementation(contd.)

- **Random Forest Regression :**

Evaluation Metrics

Mean Square Error	12.692243668652257
Root Mean Square Error	3.5626175305036965
Mean Absolute Error	2.2110714544850123
R2 Score	0.919652258962123

CV Evaluation Metrics

Mean Square Error	13.964937998034555
Root Mean Square Error	3.736969092464474
Mean Absolute Error	2.360394257697525
R2 Score	0.9115955183993694

- No Overfitting is seen.

Conclusion

EDA Conclusion :

- The number of renting bikes increases as summer season starts.
- The number of renting bikes decreases as winter season starts.
- In summer, 37% people prefer to rent a bike and in winters, only 7.9% people prefer to rent a bike.
- Most number of people prefer to rent a bike when the temperature of the day is in between 15 degree to 30 degree celsius.
- When humidity percentage is in between 30 to 70, people prefer to rent a bike.
- Most of the bikes are rented when sky is shiny or there is no rainfall or snowfall.
- Majority of the bikes are rented on a functioning day. On holiday majority of people prefer to stay home.
- When there is a functioning day, most of the bikes are rented during daytime. The peak hour of renting bike is 6 PM evening. The second highest peak hour of renting a bike is 8 AM morning.
- The lowest number of bikes are rented around 4 AM and 5 AM.

Conclusion

ML Model Conclusion :

- Random Forest algorithm gave the best performance among all of the models applied.
- The Linear and Ridge regression both shows the 77% accuracy on training dataset. On test dataset, Linear Regression shows 77% accuracy and Ridge Regression shows 78% accuracy.
- The Lasso and Elastic Net Regression both shows 76% accuracy on training dataset and 78% accuracy on test dataset.
- After doing cross validation on Lasso, Ridge and Elastic Net regression, the accuracy for all of them remain same on test dataset. This shows no overfitting.

Thank You!