

Capstone Project - 2

Bike Sharing Demand Prediction

Submitted by

Dileep Rawat

Data science trainee, Almabetter

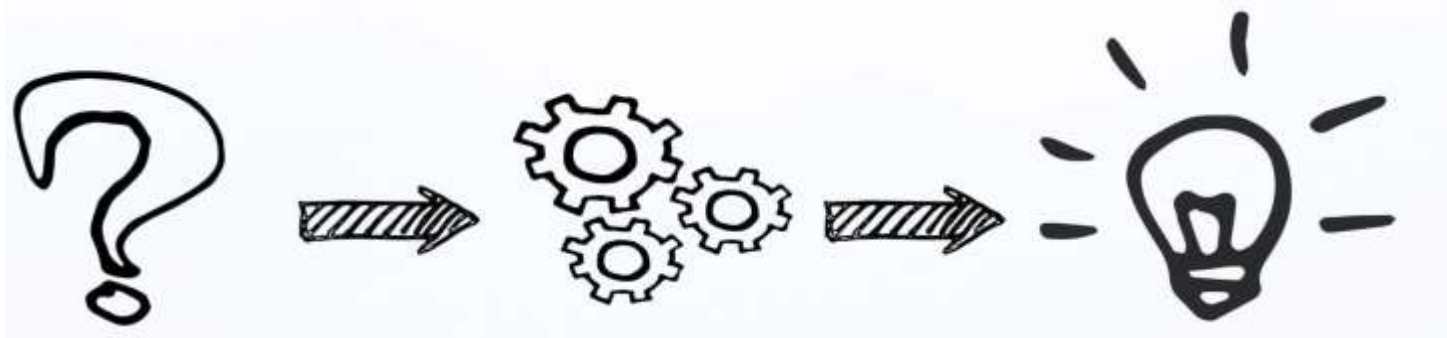
Agenda

- Problem Statement
- Data Description
- Methodology
- Pre-processing and Data Cleaning
- Exploratory Data Analysis (EDA)
- Modelling Approach
- Fitting various model
- Model performance comparison
- Model Validation
- Challenges faced and Conclusions



Problem Statement

- Bike Rentals have become a popular service in recent year and it seems people are using it more often with relatively cheaper rates and ease of pick up and drop at own convenience is what making this business thrive.
- It is important to make the rental bike available and accessible to the public at the right time as it lessens the **waiting time**, eventually, providing the city with a **stable supply** of rental bikes
- The goal of this project is to build a ML model that is able to predict the demand of rental bikes in the city of Seoul.



Data description

The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information.

Attribute Information:

- Date : year-month-day
- Rented Bike count - Count of bikes rented at each hour
- Hour - Hour of the day
- Temperature-Temperature in Celsius
- Humidity - %
- Windspeed - m/s
- Visibility - 10m
- Dew point temperature - Celsius
- Solar radiation - MJ/m²
- Rainfall - mm
- Snowfall - cm
- Seasons - Winter, Spring, Summer, Autumn
- Holiday - Holiday/No holiday
- Functional Day – NoFunc (Non Functional Hours), Fun(Functional hours)



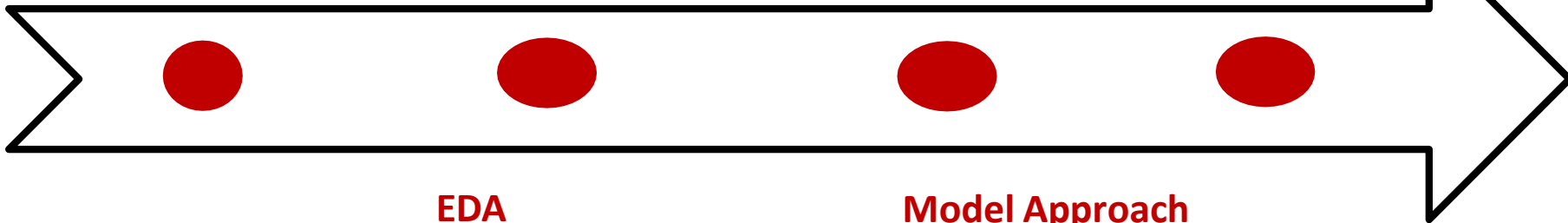
Methodology

Pre-processing & Data Cleaning

- Removing duplicate rows
- Handling missing values
- Convert column to appropriate datatype
- Adding new features and renaming the features

Model validation

- Model selection
- Feature importance
- Conclusion



EDA

- Data distribution of features
- Deal with multicollinearity
- Separate dependent and independent features

Model Approach

- Data transformation
- Fitting
- Prediction
- Evaluation matrices

Pre-processing and Data Cleaning

1. Removing Duplicate values

```
# checking duplicates
len(bike_df[bike_df.duplicated()])

0
```

There were no duplicate values

4. Adding new features and renaming the features

- In a city, it is highly likely that the rental bike demand may follow different pattern over the weekends when people do not generally go to work.
- To capture this trend, we can define a new feature 'weekend' which indicates whether a said day is a weekend (1) or not (0).

```
[ ] # engineering new feature 'weekend' from day_of_week
bike_df['weekend'] = bike_df['day_of_week'].apply(lambda x: 1 if x>4 else 0)
```

```
[ ] bike_df['month']= bike_df['date'].dt.month
bike_df['day_of_week']= bike_df['date'].dt.dayofweek

# {0: 'Monday', 1: 'Tuesday', 2: 'Wednesday', 3: 'Thursday', 4: 'Friday', 5: 'Saturday', 6: 'Sunday'}
```

2. Handling Missing values

```
[ ] # checking null values
bike_df.isnull().sum()

Date      0
Rented Bike Count  0
Hour      0
Temperature(°C)  0
Humidity(%)  0
Wind speed (m/s)  0
Visibility (10m)  0
Dew point Temperature(°C)  0
solar Radiation (MJ/m2)  0
Rainfall(mm)  0
Snowfall (cm)  0
Seasons    0
Holiday    0
Functioning Day  0
dtype: int64
```

Above result says that there are no null values in the data

3. Convert columns to appropriate datatypes

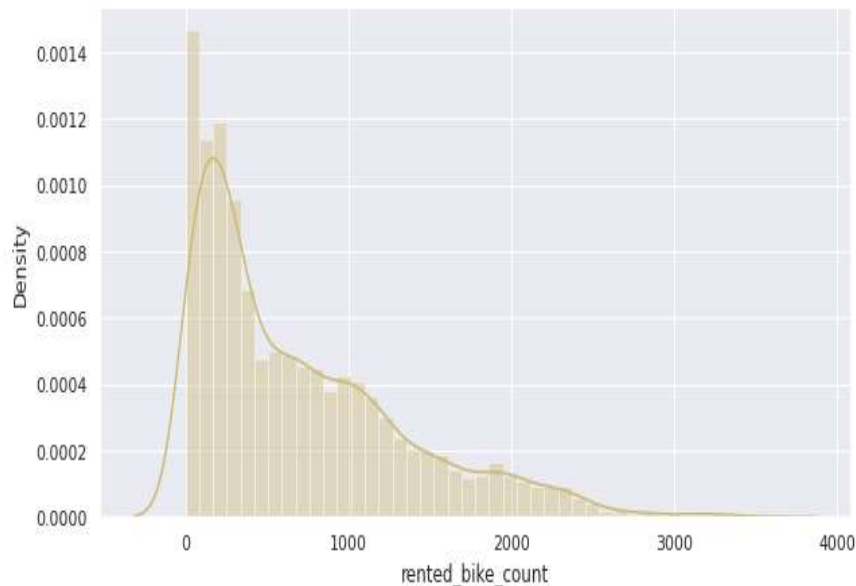
- Converted the Date column to datetime object

```
[ ] # converting date column dtype object to date
bike_df['Date']=pd.to_datetime(bike_df['Date'])
```

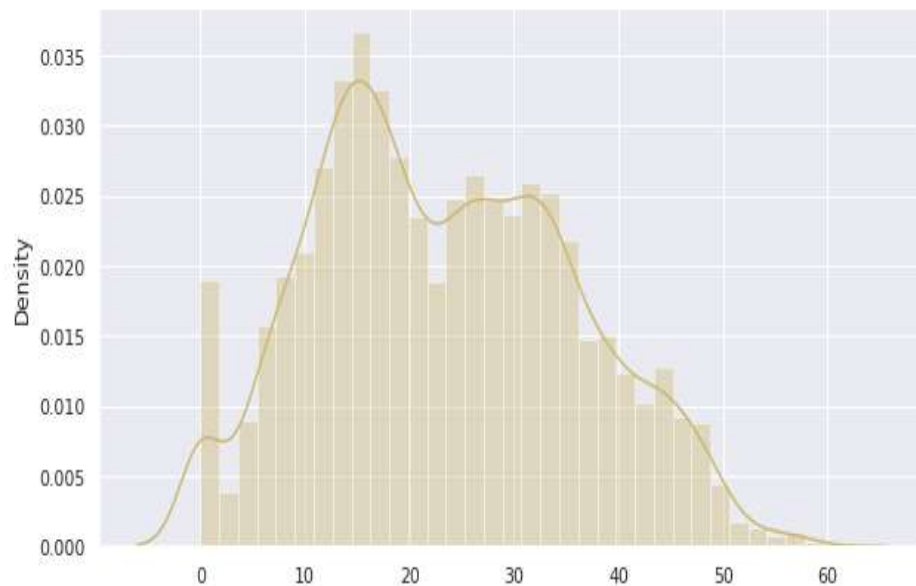
Exploratory Data Analysis (EDA)

- The dependent variable - rented bike counts is **positively skewed**

Before transformation



After using sqrt transformation

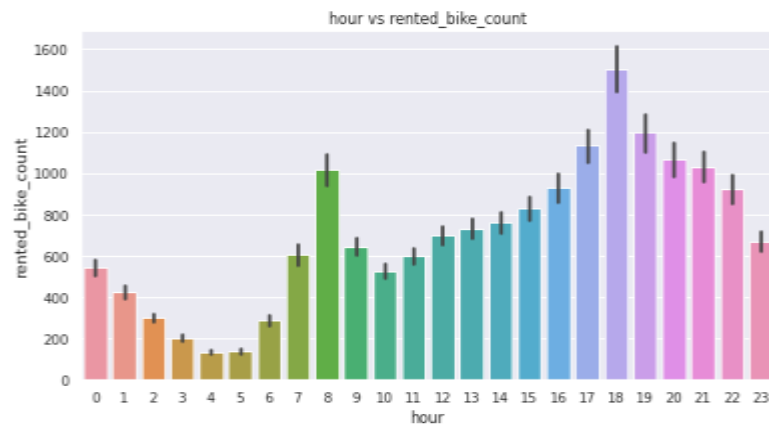
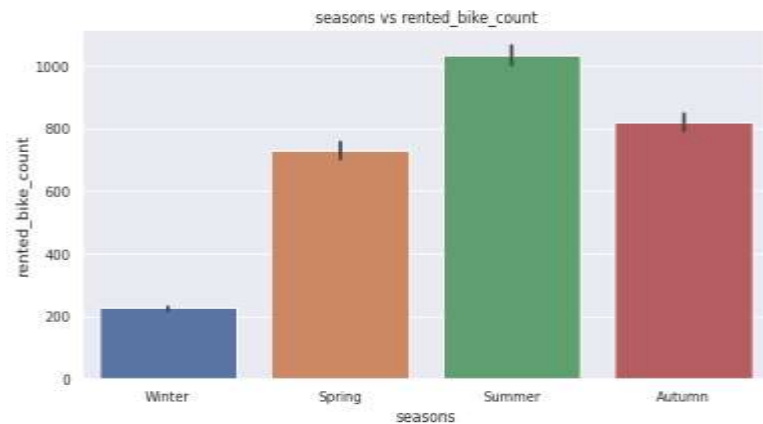
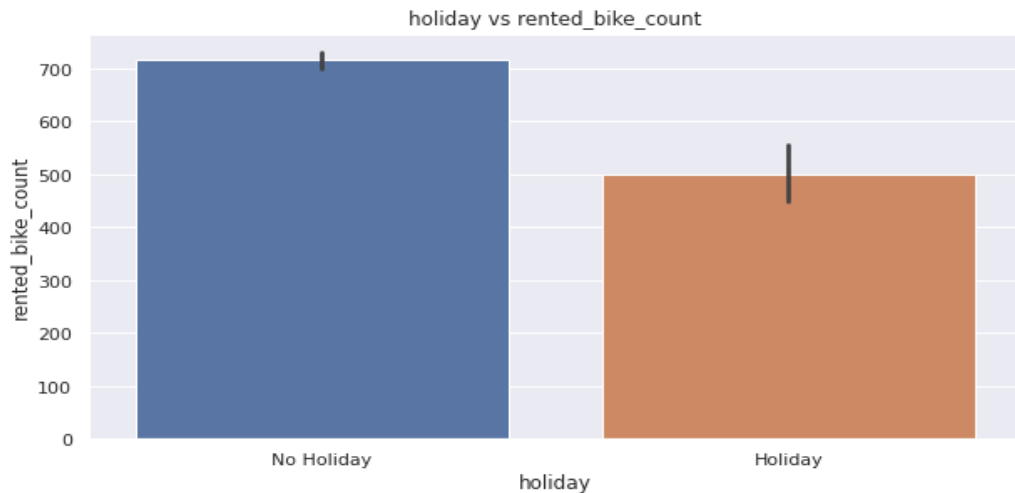


Exploratory Data Analysis (EDA)

- **Normally distributed attributes:** temperature, humidity.
- **Positively skewed attributes:** wind, solar radiation, snowfall, rainfall.
- **Negatively skewed attributes:** visibility.

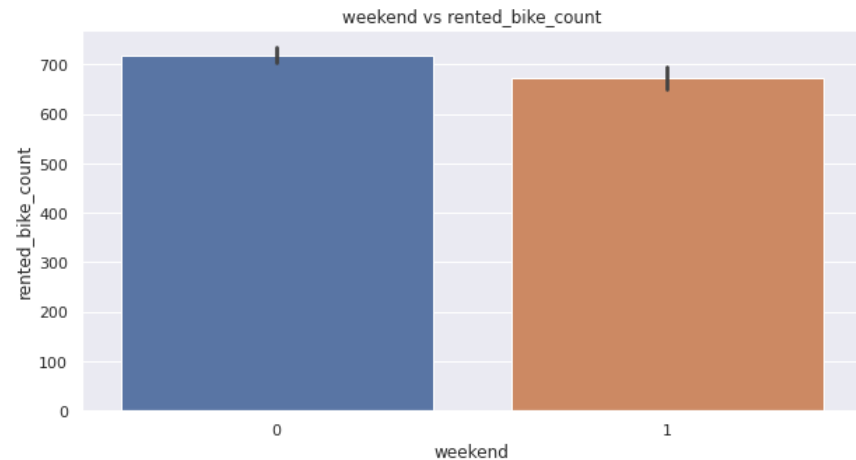
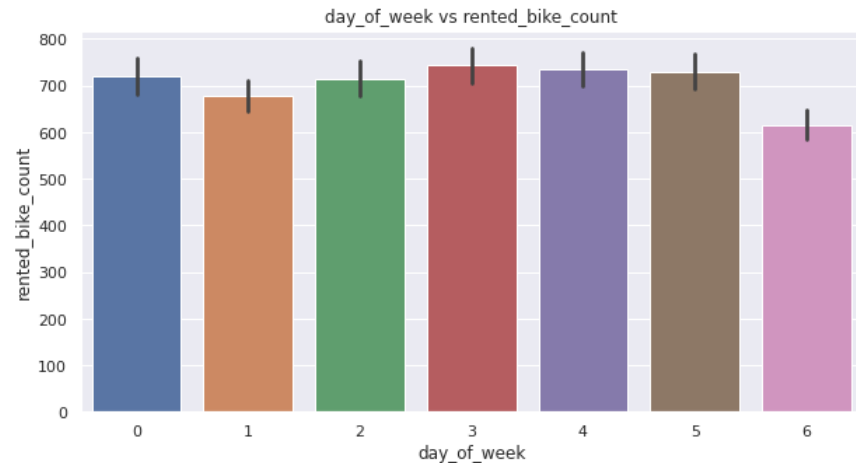
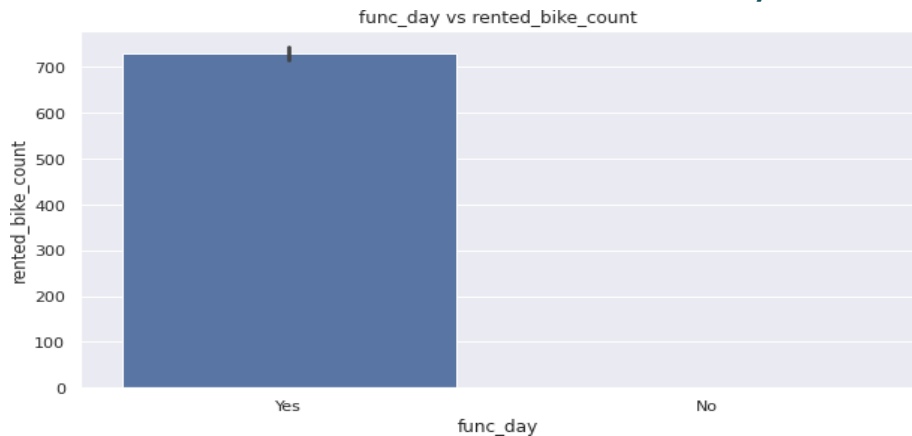
EDA (Contd.)

- The number of bikes rented is higher during the rush hours.
- The rented bike counts is higher during the summer and lowest during the winter.
- The rented bike count is higher on working days than on non working days.



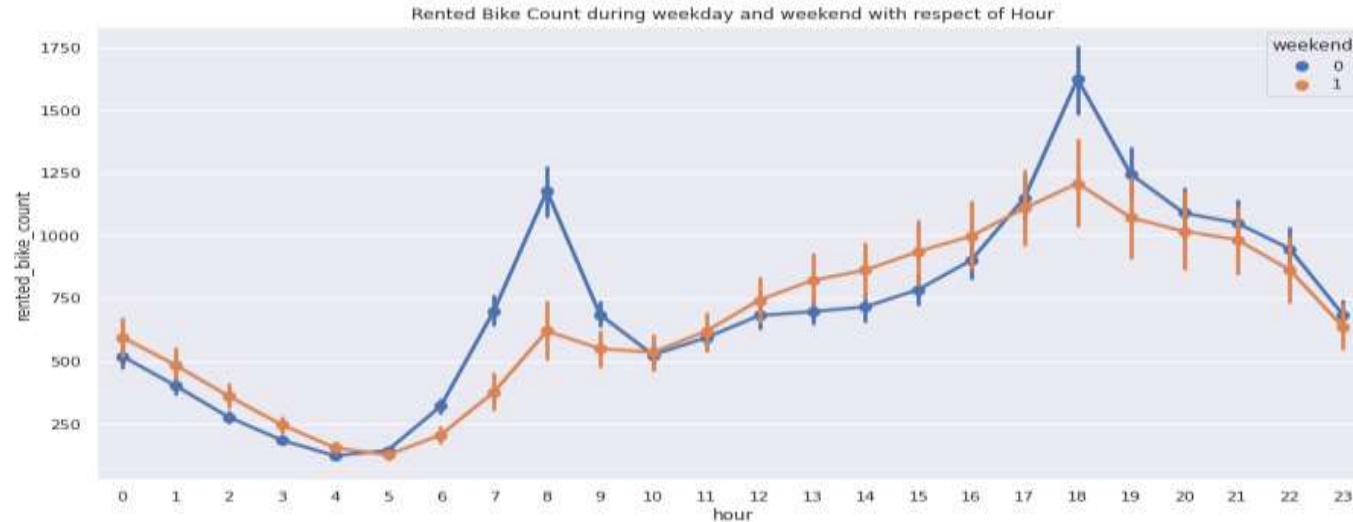
EDA (Contd.)

- On a non functioning day, no bikes are rented in all the instances of the data.
- The number of bikes rented on average remains constant throughout Monday - Saturday, it dips on Sunday.
- The rented bike counts is slightly lower on weekends than on weekdays.



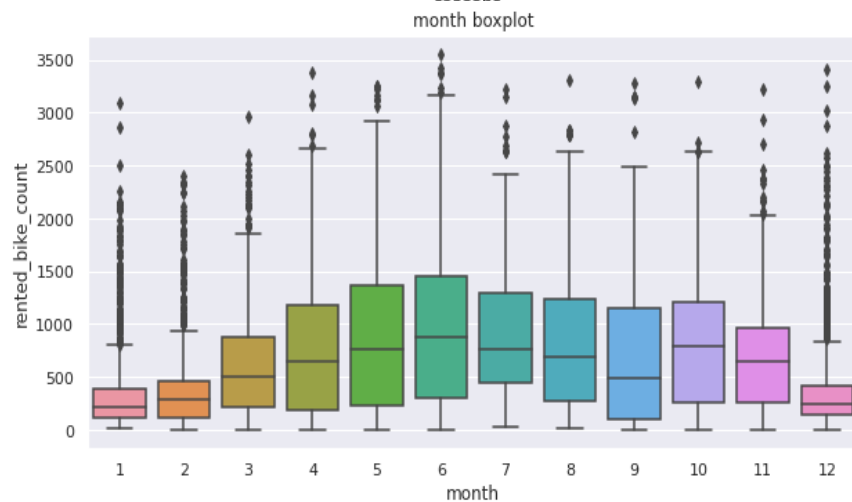
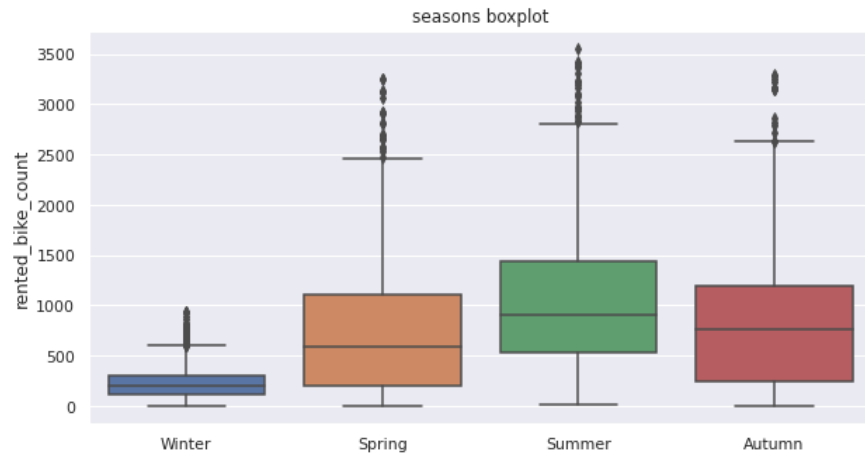
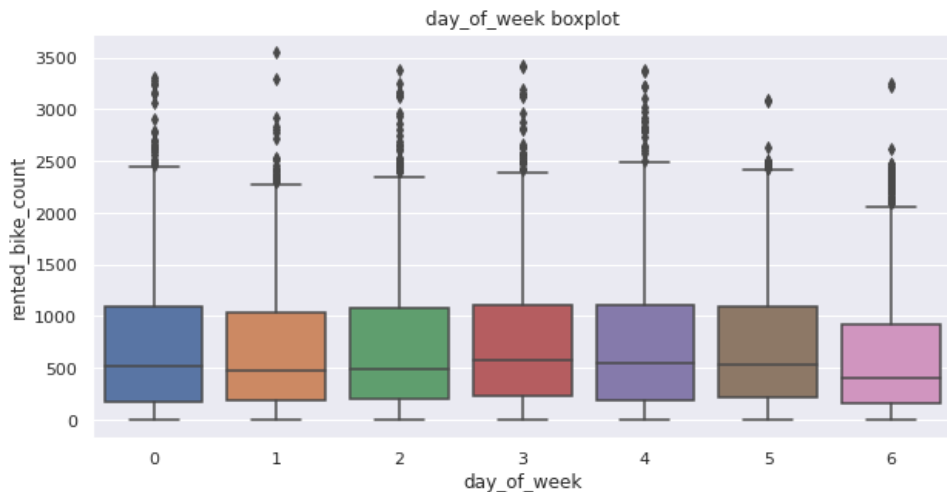
EDA (Contd.)

- On a regular day, there is a huge demand for rental bikes on morning 8 AM and Evening 6 PM.
- On holidays and weekends, the demand for rental bikes increases gradually throughout the day.



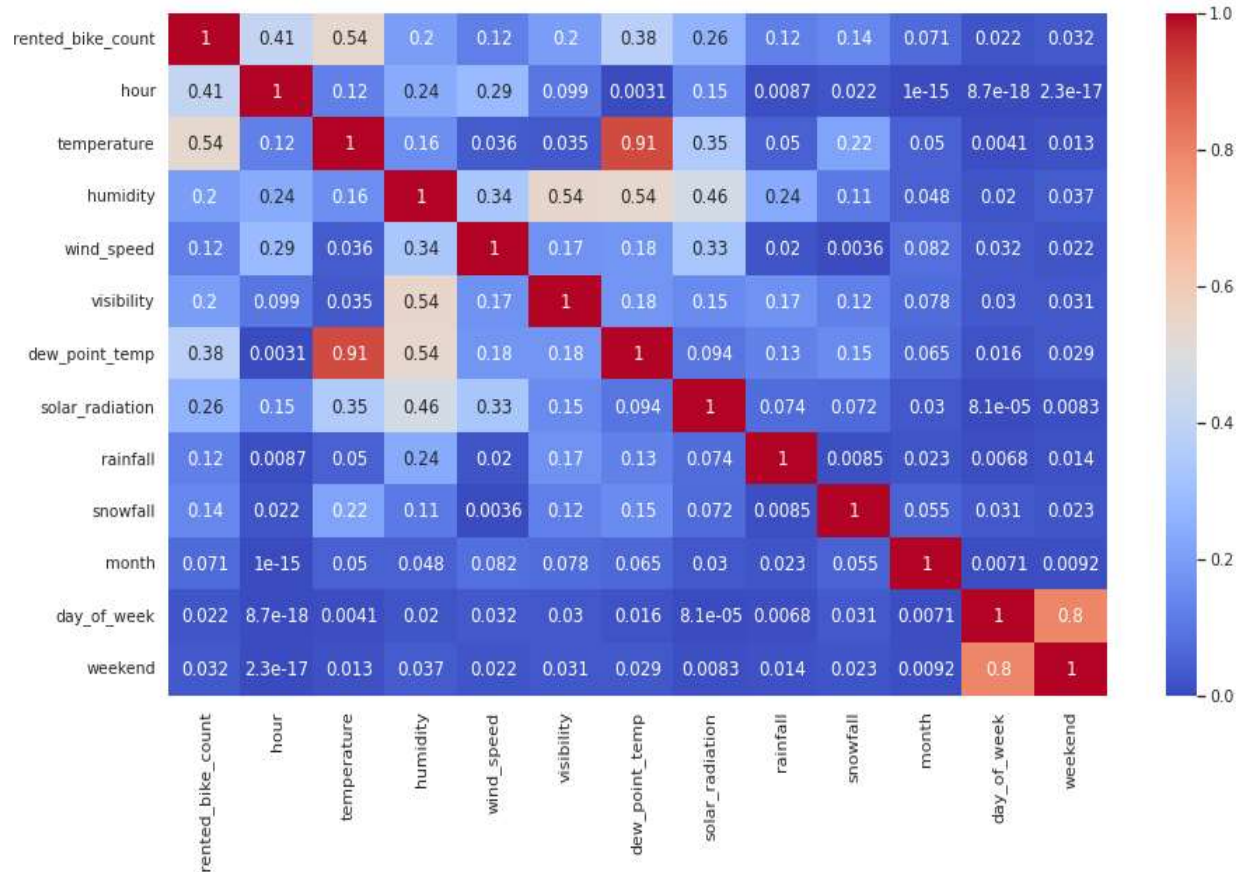
EDA (Contd.)

- There are outliers in the data.
- We cannot handle them since we may eliminate patterns we had discovered earlier.




EDA (Contd.)

- There is high correlation between temperature and dew_point_temp and between day_of_week and weekend.



EDA (Contd.)

- A variance inflation factor (VIF) is a measure of the amount of multicollinearity in regression analysis.
- Multicollinearity exists when there is a correlation between multiple independent variables in a multiple regression model.



	variables	VIF
0	temperature	3.166007
1	humidity	4.758651
2	wind_speed	4.079926
3	visibility	4.409448
4	solar_radiation	2.246238
5	rainfall	1.078501
6	snowfall	1.118901

There is no multicollinearity in the data.

EDA Summary

- The dependent variable - rented bike counts is positively skewed.
- Demand for rental bikes is lowest in the winters; highest in summers
- On regular days, there is a surge in demand for rental bikes during rush hours, this was absent during holidays and weekends.
- On a regular day, there is a huge demand for rental bikes on morning 8 AM and Evening 6 PM.
- On holidays and weekends, the demand for rental bikes increases gradually throughout the day.
- The data contains **outliers**, but we didn't handle them since by doing so, we may eliminate the patterns in the data we discovered.
- There is high correlation between temperature and dew_point_temp and between day_of_week and weekend.

Modelling Approach

- Since the data contains outliers, and many categorical attributes, It won't be wise to fit linear models, as they will give high errors.
- We will use tree models instead, since they can handle outliers and categorical attributes better than linear models.
- We will use decision tree as a baseline model.
- Subsequently, to get better predictions, we will use ensemble models: Random forests, GBM, XG Boost.
- Final choice of model will depend on whether interpretability or accuracy is important to the stakeholders.

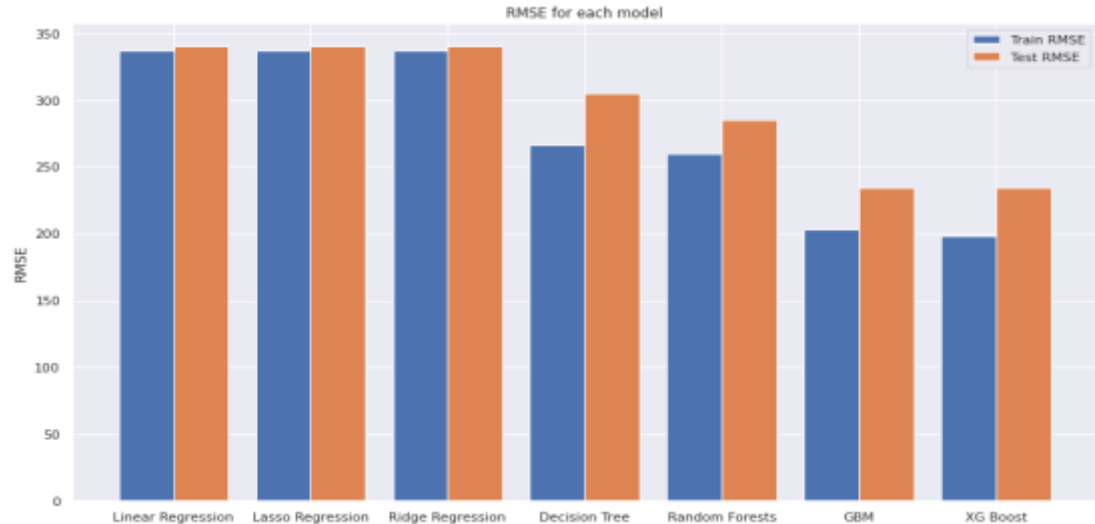
Fitting various model

1. Linear Regression
2. Lasso Regression
3. Ridge Regression
4. Decision trees
5. Random Forest
6. Gradient Boosting
7. Extreme Gradient Boosting



Model performance comparison

- The test RMSE is slightly higher than train RMSE for all models.
- The XG Boost model has the lowest train and test RMSE compared to others.



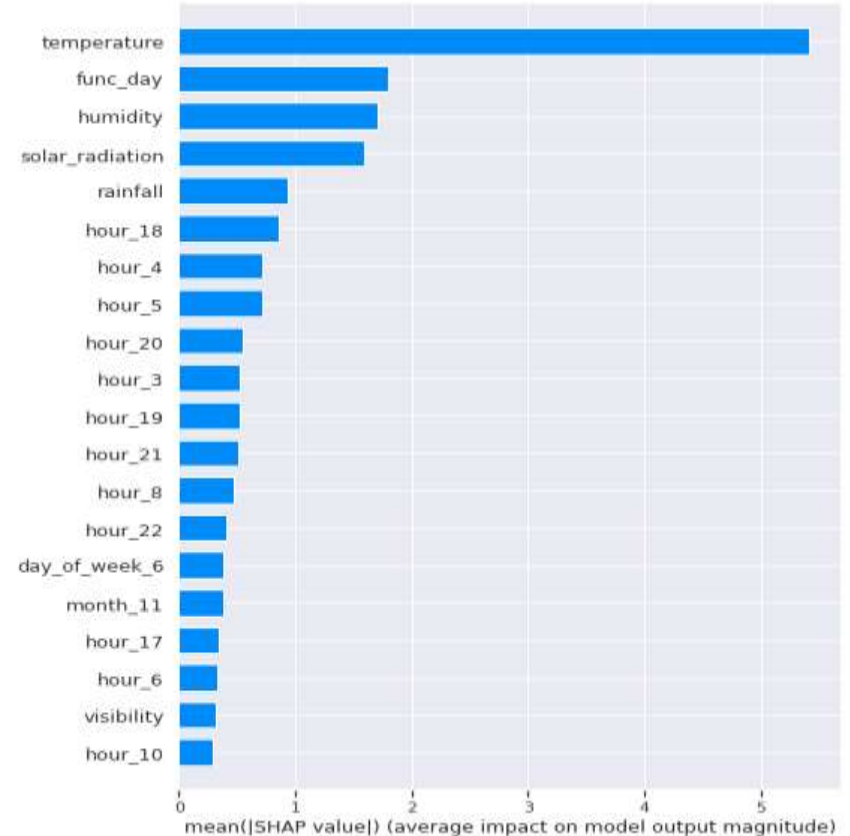
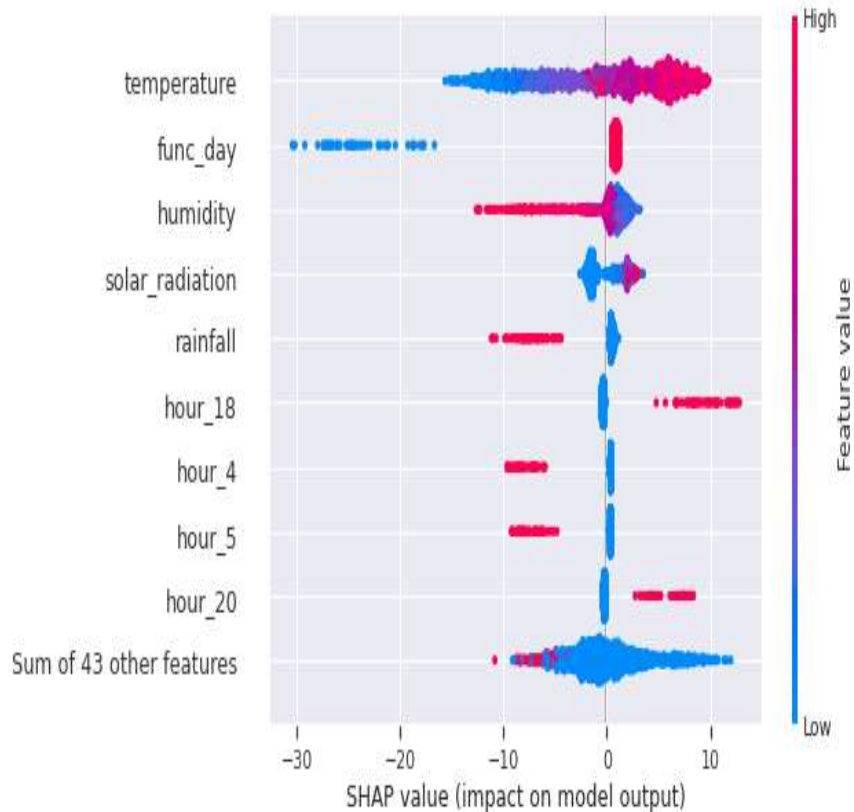
Sl. No.	Regression Model	Train RMSE	Test RMSE	Train R2 Score (%)	Test R2 Score (%)
1	Linear Regression	336.6434731173121	339.60454792093066	72.70783075000146	72.44324388929424
2	Lasso Regression	336.67583327272996	339.63459425861004	72.70258352938185	72.43836753394474
3	Ridge Regression	336.96492040087674	339.88744944073846	72.65568548144334	72.39731343678014
4	Decision Tree	266.46990157342736	304.3139647514398	82.90008382688056	77.87288148896646
5	Random Forest	259.74801093895513	284.77425770169367	83.75191735133997	80.62317779572011
6	Gradient Boosting	203.24240118623138	234.15091762746536	90.05221723508889	86.89995385236813
7	XG Boost	198.03930327528042	233.77731132104725	90.5550331511655	86.94172482055693

Model validation

- By observing Evaluation matrices for all the models-
 - Linear Regression, Lasso and Ridge are not at its best.
 - Decision Trees, Random Forest are quite good with linear models, but they are not giving optimum prediction.
 - Gradient boosting type models are giving better results.



Top features which helping to make our prediction



Challenges Faced

- Comprehending the problem statement, and understanding the business implications
- Feature engineering – deciding on which features to be dropped / kept / transformed
- Choosing the best visualization to show the trends among different features clearly in the EDA phase
- Deciding on how to handle outliers
- Choosing the ML models to make predictions
- Deciding the evaluation metric to evaluate the models
- Choosing the best hyperparameters, which prevents overfitting

Conclusion

- We trained 7 unique Machine Learning models using the training dataset, and the its respective performance was improved through hyperparameter tuning.
- Thus, we have successfully built predictive models that can predict the demand for rental bikes based on different weather conditions and other.
- The XG Boost prediction model had the lowest RMSE.
- The final choice of model for deployment depends on the business need; if high accuracy in results is necessary, we can deploy XG Boost model.
- If the model interpretability is important to the stakeholders, we can choose deploy the decision tree model.

