Bike sharing dataset:

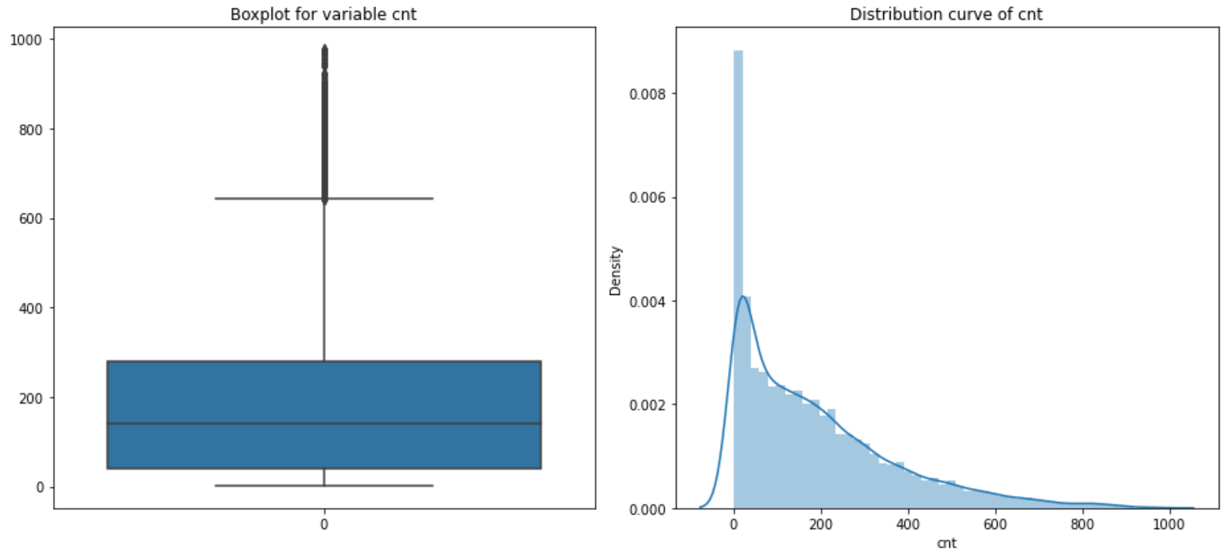
The prediction of the bike sharing count value is done to:

- estimate the expected revenue of the bike sharing company
- provide the required number of bicycles at any particular bike storage location
- Predict the usage of customers during various weather conditions.

EDA (Explorative Data Analysis):

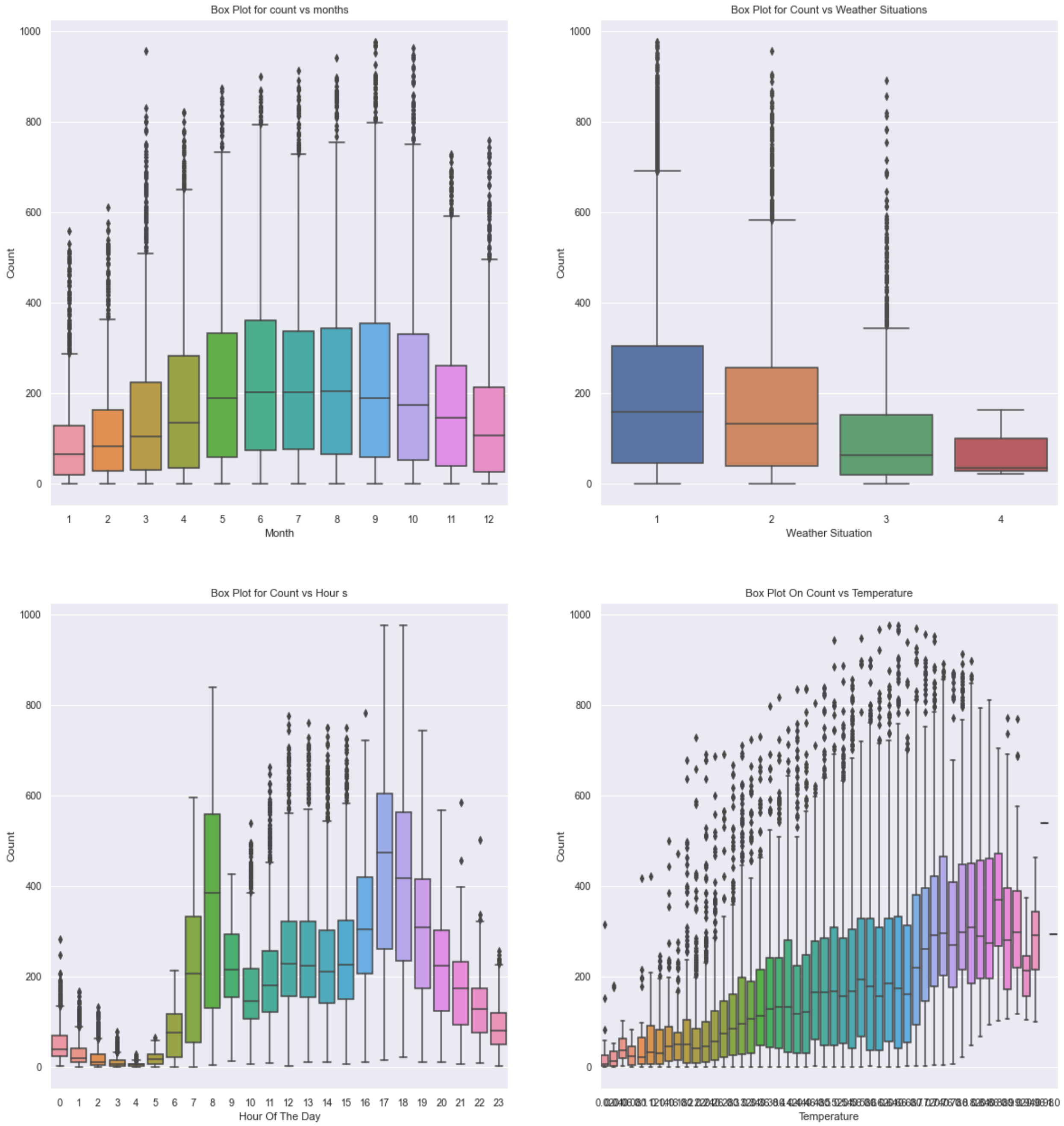
EDA Analysis is performed to analyze and visualize data set in order to summarize their main characteristics. The primary goal of EDA is to understand the data's underlying structure, patterns, and relationships between variables.

Let us first look at how the target variable 'cnt' is distributed in the dataset.

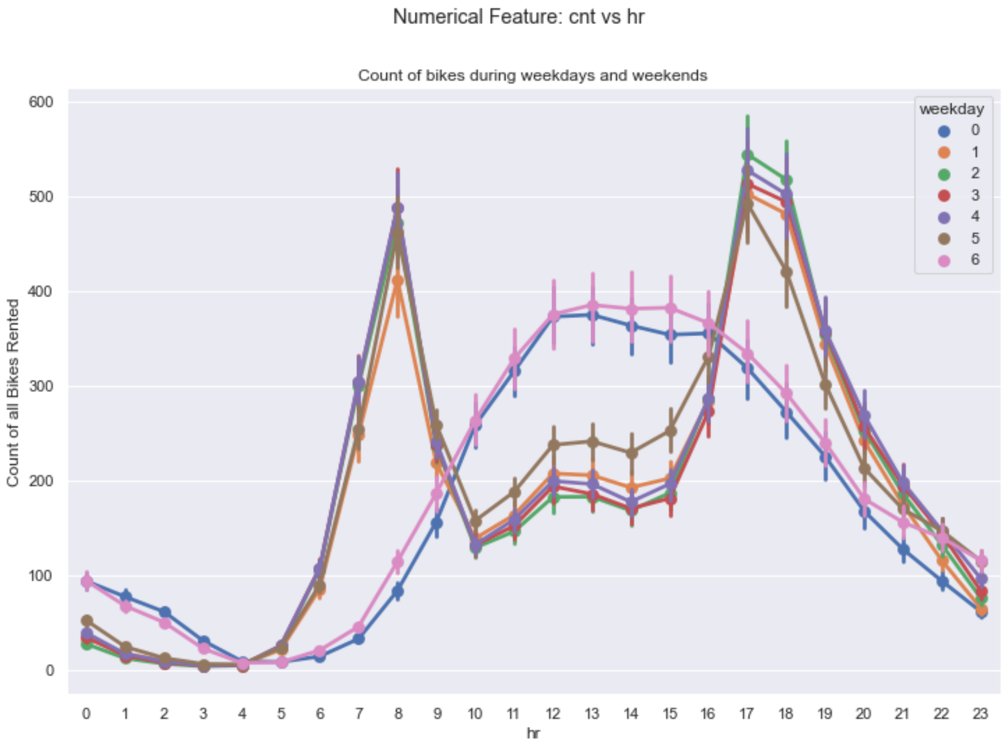


Observation:

- Both the plots show a right skewed data. This means that the mean is greater than the median, indicating that the majority of data points are concentrated in the lower range of values, while a few extreme values pull the mean higher. Hence, we conclude that the data contains outliers. In the following step, the outliers of the count values were removed using median and interquartile range (IQR) as the count values do not fit a normal distribution. This reduces the data set from 15641 to 15179 samples.

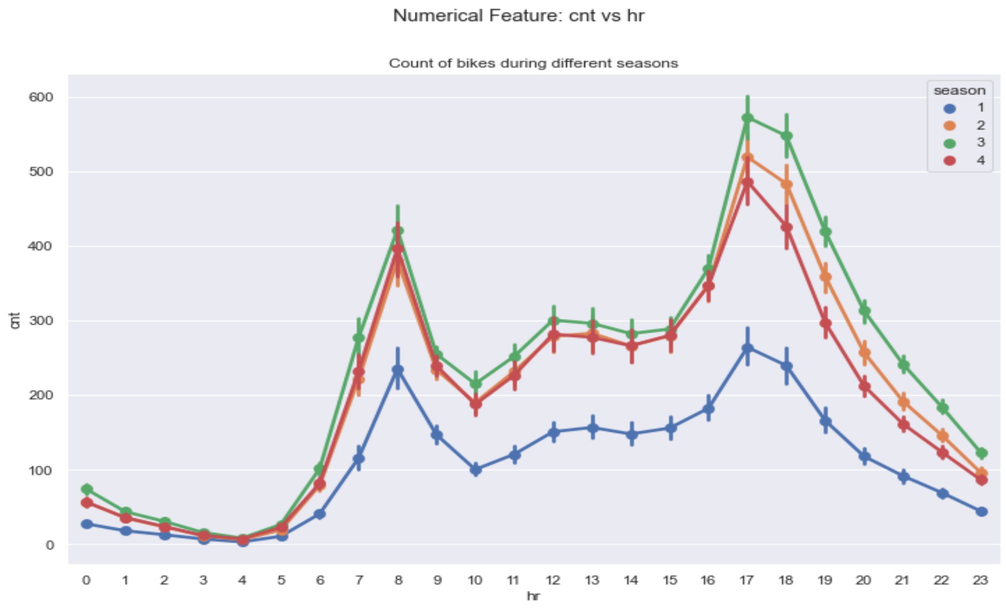


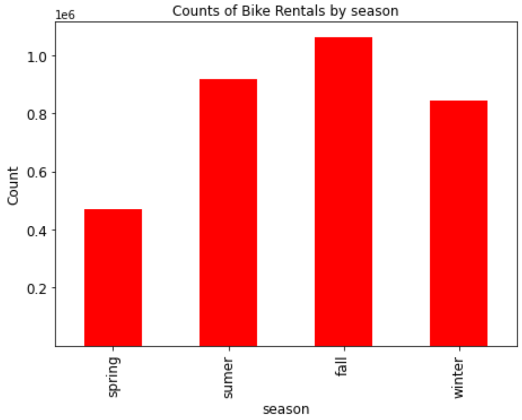
The hourly box plots show a local maximum at 8 am and one at 5 pm which indicates that most users of the bicycle rental service use the bikes to get to work or school. Another important factor seems to be the temperature: higher temperatures lead to an increasing number of bike rents and lower temperatures not only decrease the average number of rents but also shows more outliers in the data.

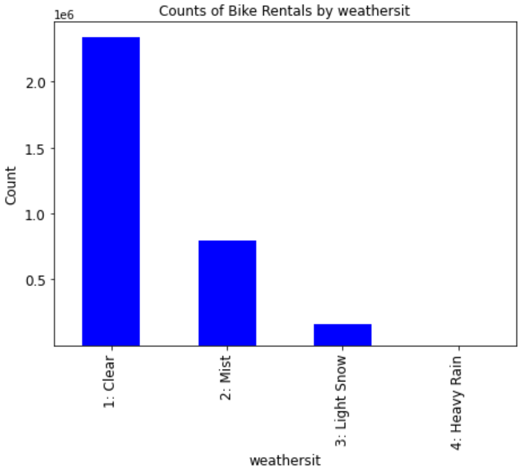


- Analysis of customer count vs Days. High demand: 7-9 am and 17-19 hours
- Average demand : 10-16 hours
- Low demand : 0-6 and 20-24 hours

Analysis of customers based on seasons, we see max no. of customers using the rental bikes during Fall season which is represented by 4.





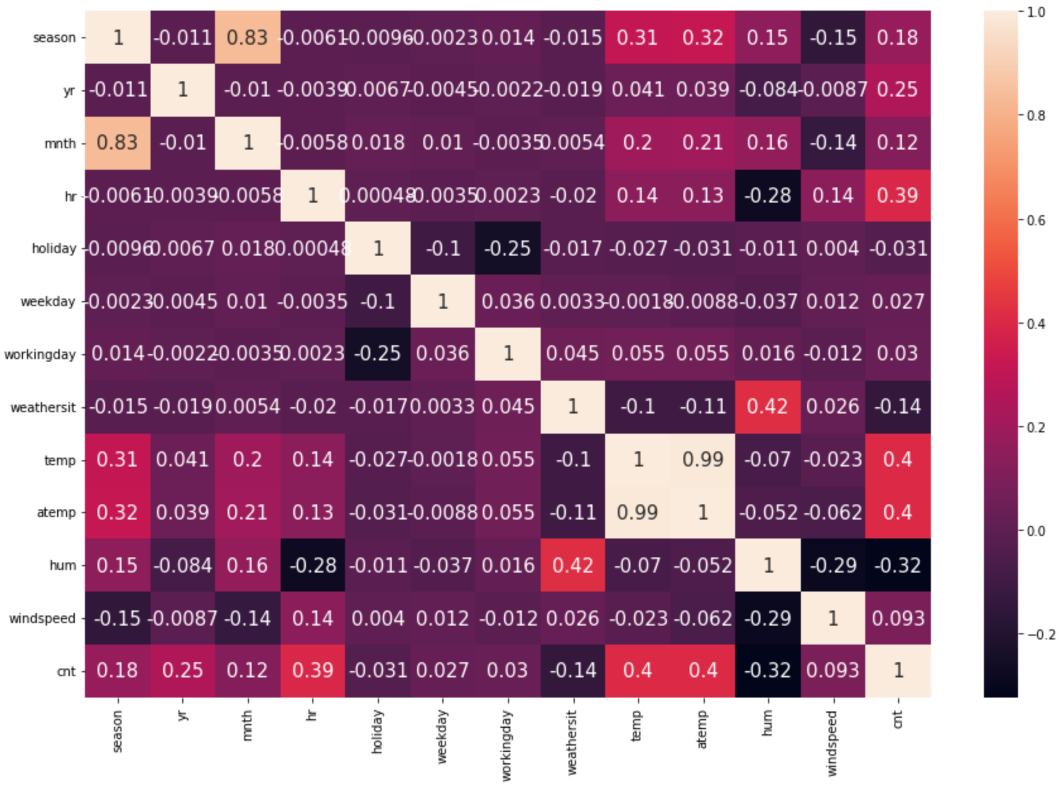


The above graphs confirm the point plot showing maximum users in Fall season and with clear sky conditions.

Correlation matrix:

This is a great way to analyze relationships between variables in a dataset. Correlation matrices provide insights into how pairs of variables are related to each other.

- -1 indicates a perfect negative correlation, 1 indicates a perfect positive correlation, and 0 indicates no linear correlation.
- A positive correlation (coefficient > 0) implies that as one variable increases, the other tends to increase as well. A negative correlation (coefficient < 0) implies that as one variable increases, the other tends to decrease.
- A coefficient close to -1 or 1 indicates a strong relationship, while a coefficient close to 0 suggests a weak or no linear relationship.



Conclusion:

- Casual and registered contain direct information about the bike sharing count which is to predict. Therefore, they are not considered in the feature set.
- The variables "hr" and "temp" seem to be promising features for the bike sharing count prediction.

Model selection

Predicting the count values necessitates the use of a regression algorithm that takes into account both categorical and numerical attributes.

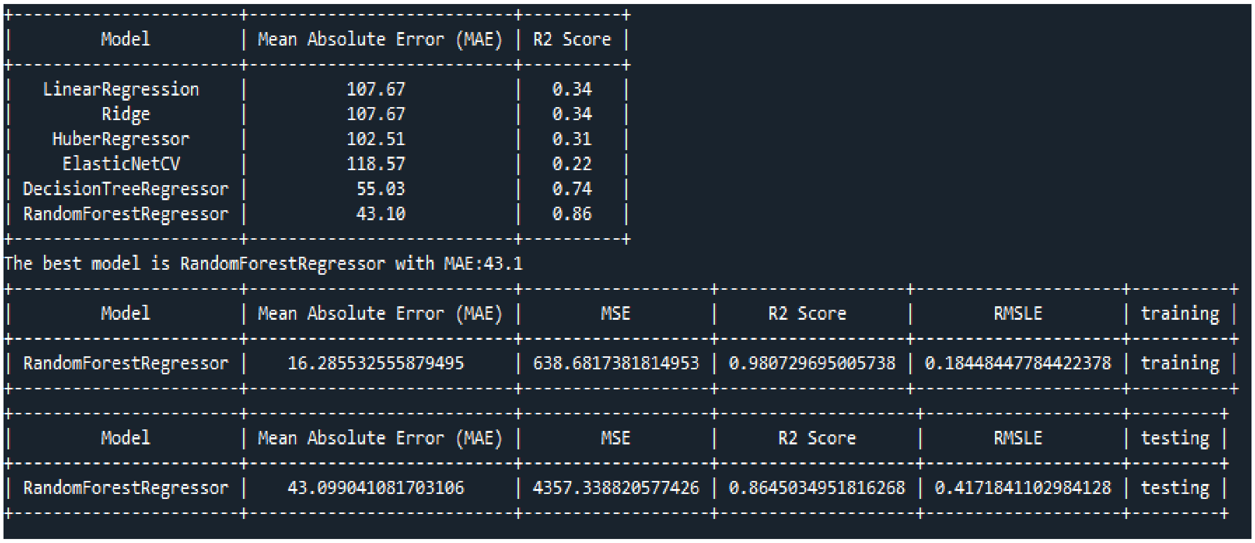
The dataset is small to medium sized and the analysis process highlighted the potential significance of certain features.

Given these aspects of the task and the data, I assessed a range of potential regression algorithms, including Linear Regression, Ridge, Elastic Net CV, Decision Tree and Random Forest.

Random forest model:

Random forest model shows the best result and is therefore chosen.

The results of the code are displayed here:



```
+-----------------------+---------------------------+----------+
|         Model         | Mean Absolute Error (MAE) | R2 Score |
+-----------------------+---------------------------+----------+
|    LinearRegression   |           107.67          |   0.34   |
|         Ridge         |           107.67          |   0.34   |
|     HuberRegressor    |           102.51          |   0.31   |
|      ElasticNetCV     |           118.57          |   0.22   |
| DecisionTreeRegressor |           55.03           |   0.74   |
| RandomForestRegressor |           43.10           |   0.86   |
+-----------------------+---------------------------+----------+
The best model is RandomForestRegressor with MAE:43.1
+-----------------------+---------------------------+-------------------+-------------------+---------------------+----------+
|         Model         | Mean Absolute Error (MAE) |        MSE        |      R2 Score     |        RMSLE        | training |
+-----------------------+---------------------------+-------------------+-------------------+---------------------+----------+
| RandomForestRegressor |     16.285532555879495    | 638.6817381814953 | 0.980729695005738 | 0.18448447784422378 | training |
+-----------------------+---------------------------+-------------------+-------------------+---------------------+----------+
+-----------------------+---------------------------+-------------------+--------------------+--------------------+---------+
|         Model         | Mean Absolute Error (MAE) |        MSE        |      R2 Score      |       RMSLE        | testing |
+-----------------------+---------------------------+-------------------+--------------------+--------------------+---------+
| RandomForestRegressor |     43.099041081703106    | 4357.338820577426 | 0.8645034951816268 | 0.4171841102984128 | testing |
+-----------------------+---------------------------+-------------------+--------------------+--------------------+---------+
```

Part-2

Solution to handle larger data.

1. Distributed Computing and Big Data Technologies: To handle large datasets, could use cloud technologies or big data platforms like Apache Hadoop, Apache Spark, and cloud-based platforms like AWS, Google Cloud, or Azure provide the infrastructure to handle and process massive amounts of data in parallel.

2. Data Storage: Distributed file systems like Hadoop Distributed File System (HDFS) or cloud-based object storage solutions provide scalable storage options.

3. Data Preprocessing: With large datasets, preprocessing becomes tricky. Technologies like Apache Spark allow for distributed data preprocessing, which is essential for data cleaning, feature engineering, and transformation. Spark's ability to cache and persist data in-memory or on disk enhances data processing speed.

4. Model Parallelization: Traditional machine learning models can't handle terabytes of data in-memory. We need to distribute the training process. Tools like TensorFlow, PyTorch, and Horovod facilitate distributed deep learning across multiple GPUs and machines.

Scaling Properties: The key scaling properties of this approach include parallelism, distribution, and efficient resource utilization. As data grows, we can add more nodes to the distributed cluster, ensuring linear scalability. Distributed data storage systems can scale horizontally to accommodate larger datasets.

Challenges and Drawbacks:

- Increased complexity: Distributed systems are more complex to set up and manage.
- Cost: Large-scale storage and computation can be expensive, especially in cloud environments and could introduce latency.
- Data quality: Ensuring data quality at scale is challenging and may require advanced data validation techniques.

In a nutshell, scaling up to handle terabyte-scale datasets requires a shift towards distributed computing, big data technologies, and scalable storage solutions. By leveraging these technologies and addressing associated challenges, we can analyze massive datasets.

I do not have hands on experience with Technologies like Apache Spark/ Hadoop but I am very much interested in learning more and implementing using these technologies.