# Report: Predict Bike Sharing Demand with AutoGluon

#### SANCHARIKA DEBNATH

## Project description

The bicycle sharing system is a method of renting bicycles. Registering members, renting a car, and returning the car will all be done automatically through a network of stations in the city. Through this system, people can rent a bicycle from one place as needed and ride to their destination. return.

In this competition, participants need to combine the usage patterns under historical weather data to predict the bicycle rental demand of the D.C. Washington Capital Bike Sharing Project.

## Initial Training

### What did you realize when you tried to submit your predictions? What changes were needed to the output of the predictor to submit your results?

The two column from train csv file were not in test csv file so we had to remove it in order to make equal columns for both training and test sets. On applying TabularPredictor in training set setting labels to count column get a minimum score of 124.419851 in WeightedEnsemble\_L2 model . Initial the kaggle score was of 1.36363.

### What was the top ranked model that performed?

WeightedEnsemble\_L2 on 3 bag folds and 3 stack level it gave the best performance by scoring 35.597325.

## Exploratory data analysis and feature creation

### What did the exploratory analysis find and how did you add additional features?

* It is found that the skewness of the data density distribution is more serious, and there is a very long tail.
* Members travel more at work and less travel on holidays, while temporary users are the opposite;
* The overall number of travelers in the first quarter was relatively small;
* The number of leases decreases as the weather level rises;
* The number of hours has a significant impact on the rental situation, with two peaks for members and a normal distribution for non-members;
* The number of leases decreases as the wind speed increases;
* Temperature and humidity have a greater impact on non-members, but less on members
* Because the time period has the greatest impact on the number of leases, this data is shown first
* There are two peaks in car usage for member users during working days, and there will be a small peak at noon. It is guessed that it may be someone who is out for lunch;
* The fluctuations for temporary users are relatively gentle, and the peak period is around 17:00;
* And the number of cars used by member users far exceeds that of temporary users.
* For non-working days, the number of leases presents a normal distribution over time, with a peak at about 14:00 and a trough at about 4, and the distribution is relatively even.
* The overall rental situation of shared bicycles in 2012 has increased compared with 2011;
* The leasing situation fluctuates significantly from month to month;
* The data fluctuated sharply from September to December 2011, and from March to September 2012;
* There are many local troughs.
* On weekdays, the number of trips of member users is large, and the number of temporary users is small;
* On weekends, the number of member user leases decreased, and the number of temporary user leases increased.
* Since the number of holidays in a year is very small, let’s first look at the number of days under the holidays each year.
* The number of vacation days accounts for a very small share of the number of days in a year, so the daily average is taken for holidays and non-holidays

### How much better did your model preform after adding additional features and why do you think that is?

After adding five new features, i.e. date, hour, year, month, day, and changing data type of weather and season to category in both train and test set resulted in better model. On applying TabularPredictor after newly adding features in training set setting labels to count column get a minimum score of 35.071134 in WeightedEnsemble\_L2 model . Which result in the kaggle score as of 0.49831. Thus on exploratory analysis the model performance increased by 0.86532.

## Hyper parameter tuning

### How much better did your model preform after trying different hyper parameters?

Different hyper tuning arguments results in difference in evaluation score. The best score was obtained on applying 3 number of bag folds and 3 stack levels by 35.597325.

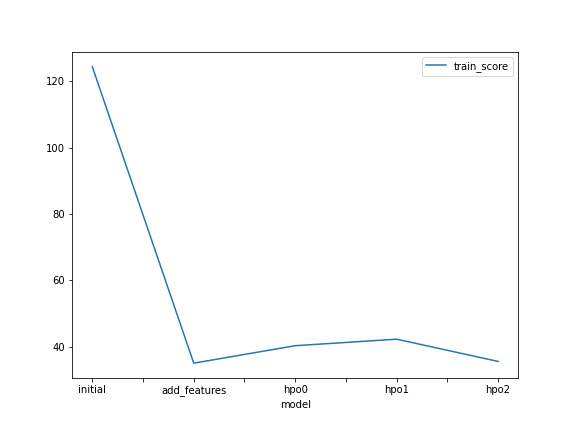
### If you were given more time with this dataset, where do you think you would spend more time?

From all the models we can say that Weighted Ensemble Model performed the best in all case, so we can use different hyperparameters on Weighted Ensemble Model for better evaluation score.

### Create a table with the models you ran, the hyperparameters modified, and the kaggle score.

| **model** | **hpo1** | **hpo2** | **hpo3** | **score** |
| --- | --- | --- | --- | --- |
| **0** | initial | Default | Default | Default |
| **1** | add\_features | Parsing of Dates | Four columns Added | feature selection |
| **2** | hpo | GBM, NN | RF ,CAT | Bagging ,Stack level |

### Create a line plot showing the top model score for the three (or more) training runs during the project.

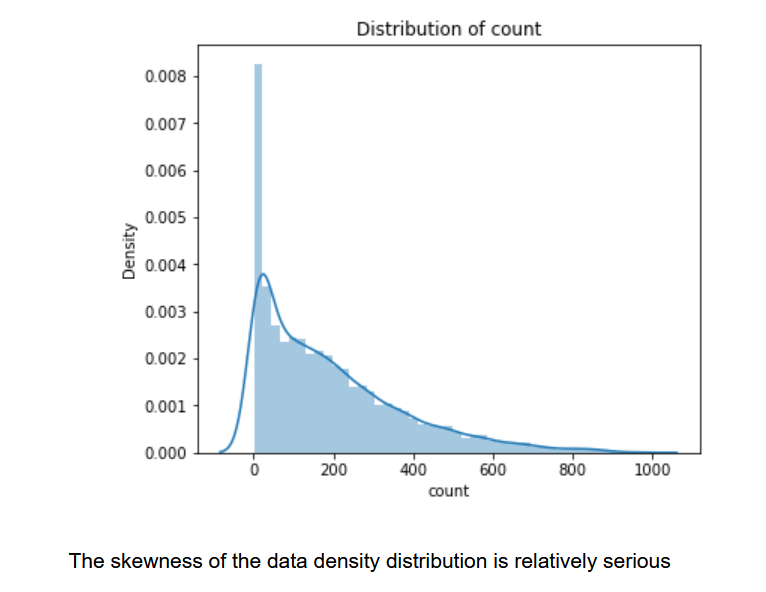


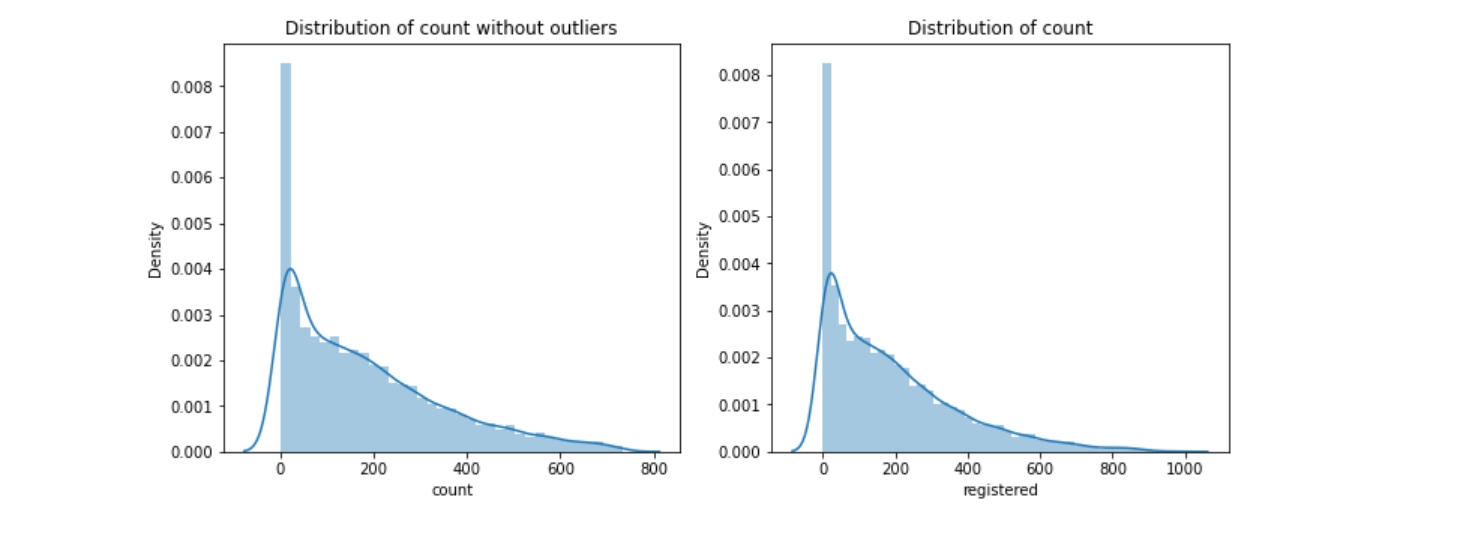
### Create a line plot showing the top kaggle score for the three (or more) prediction submissions during the project.

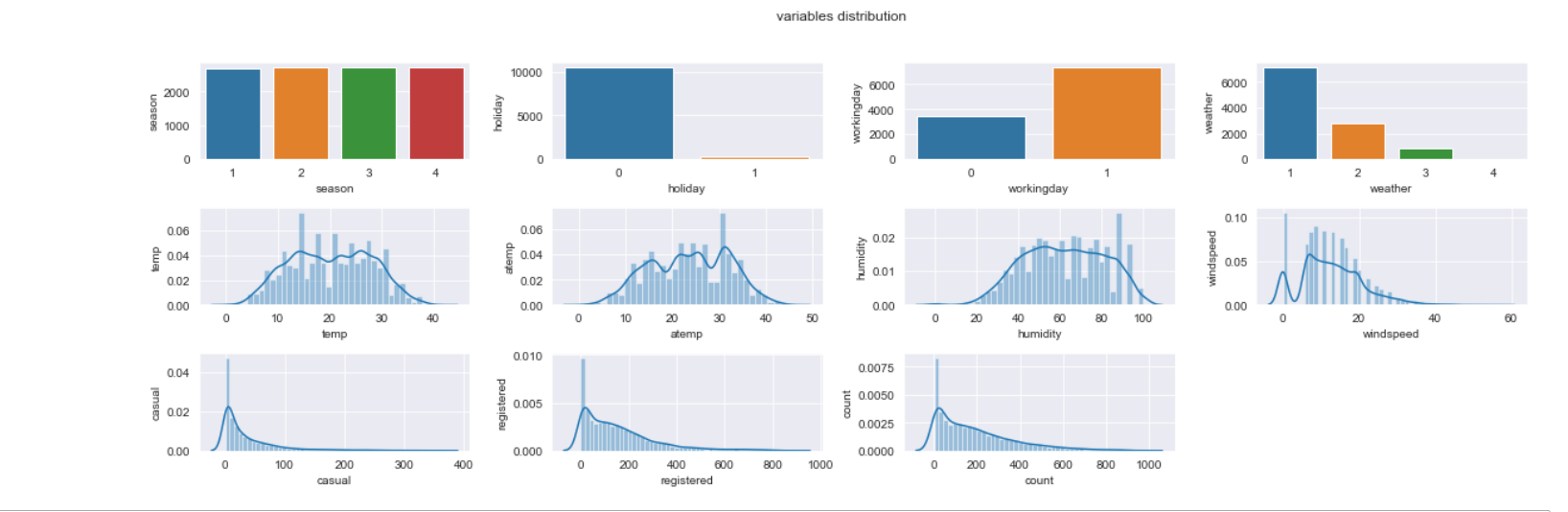
### model_test_score

**Visualization**

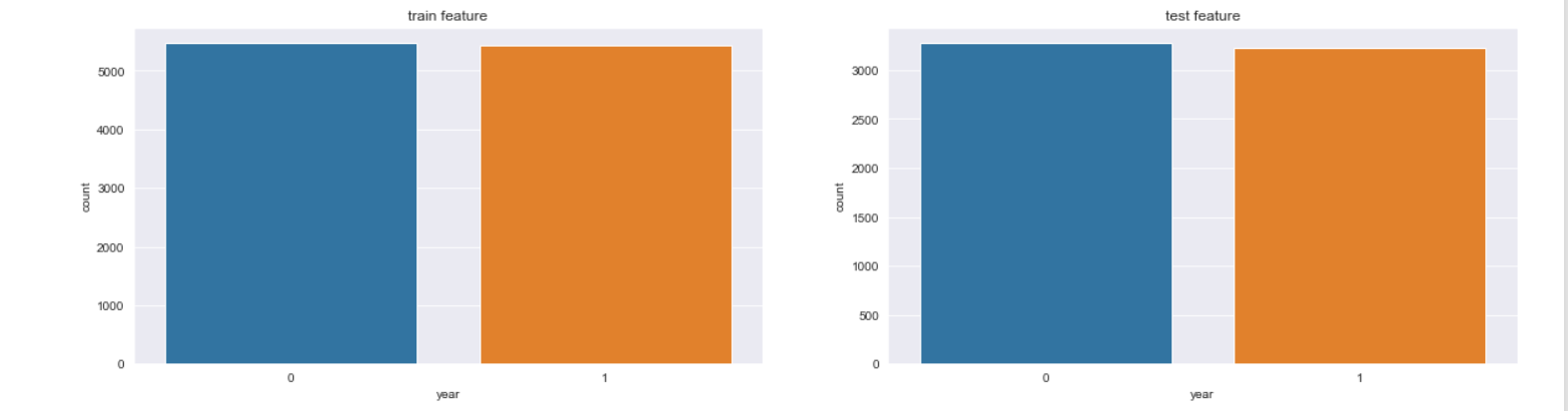
**Distribution of counts**

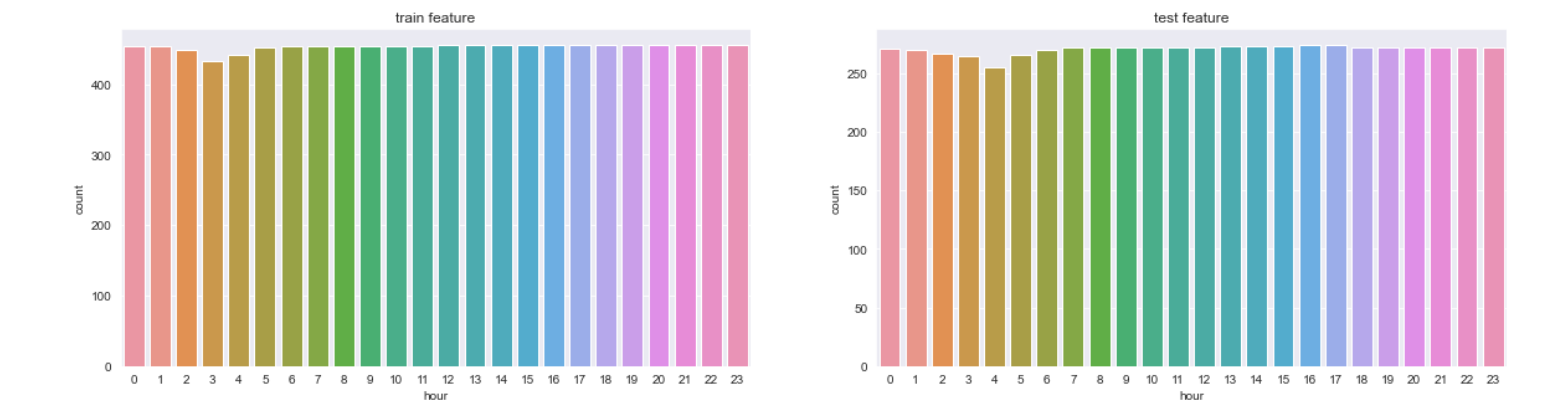
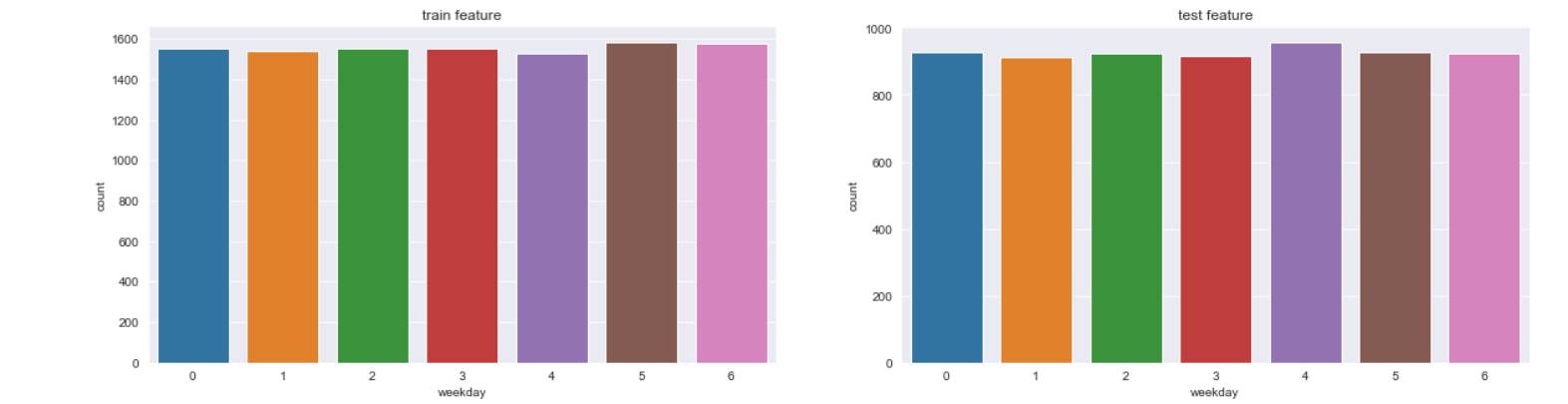
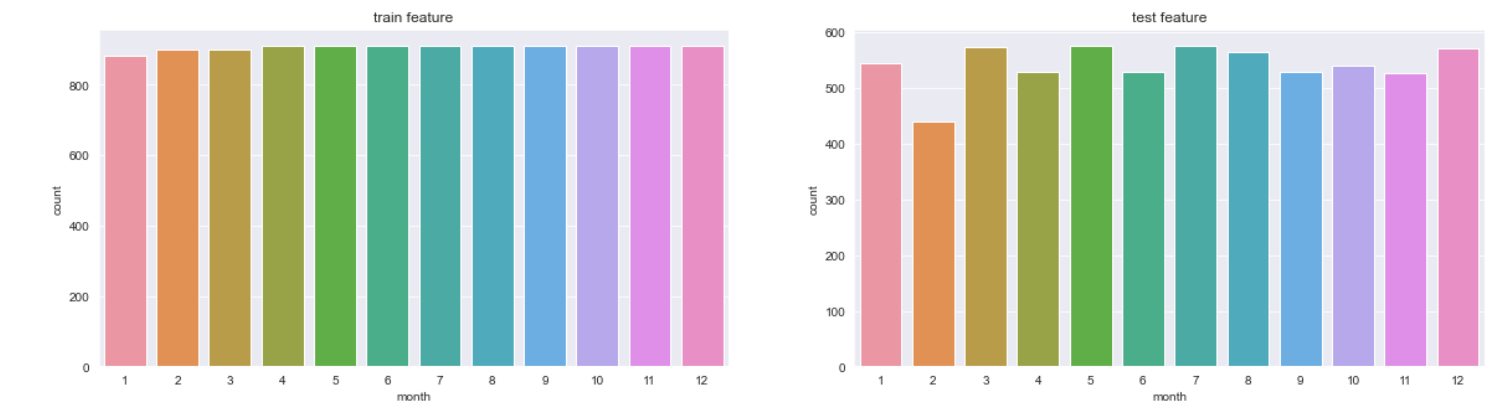






## Train Features and Test features





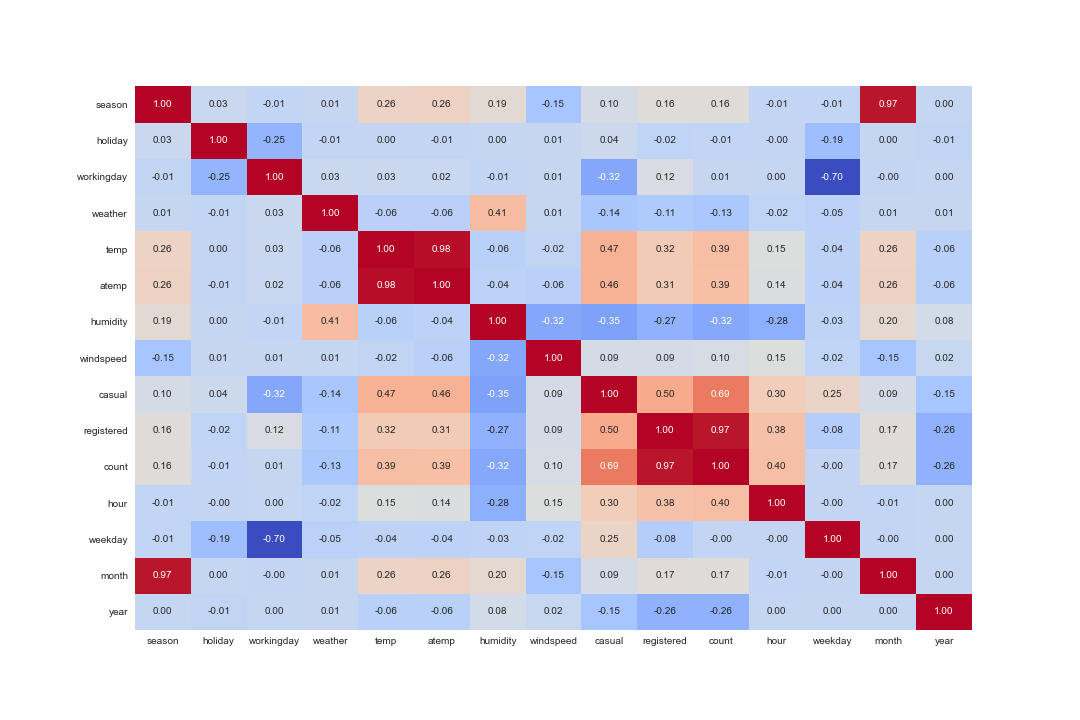
# Relationship between weekday and each count by working day and holiday

# 

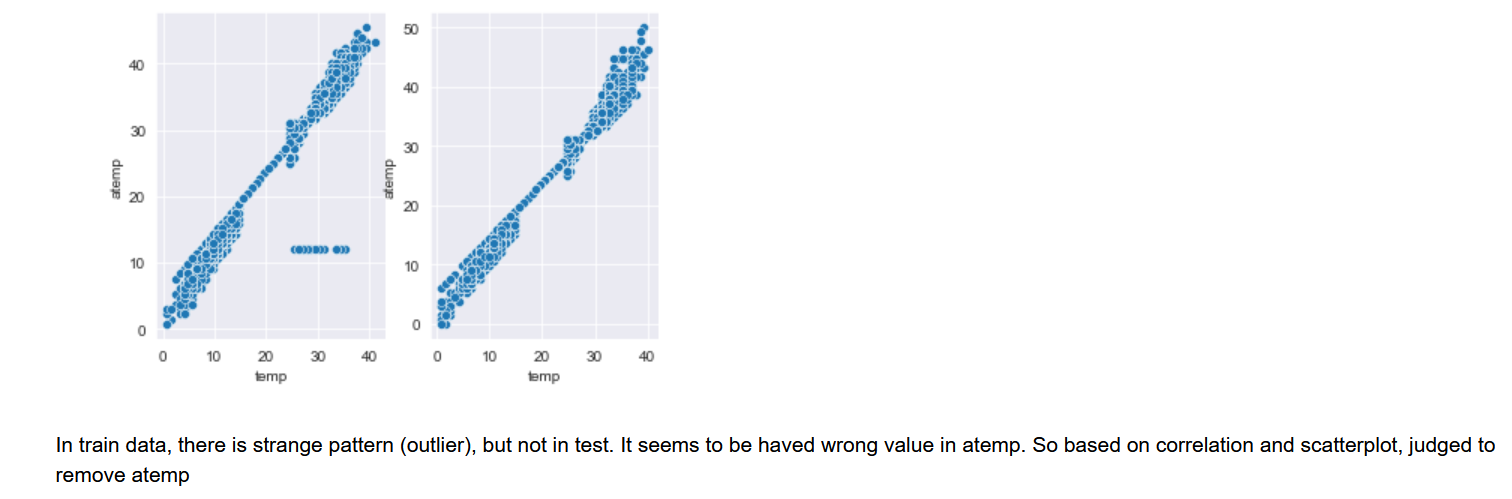
## 

## 

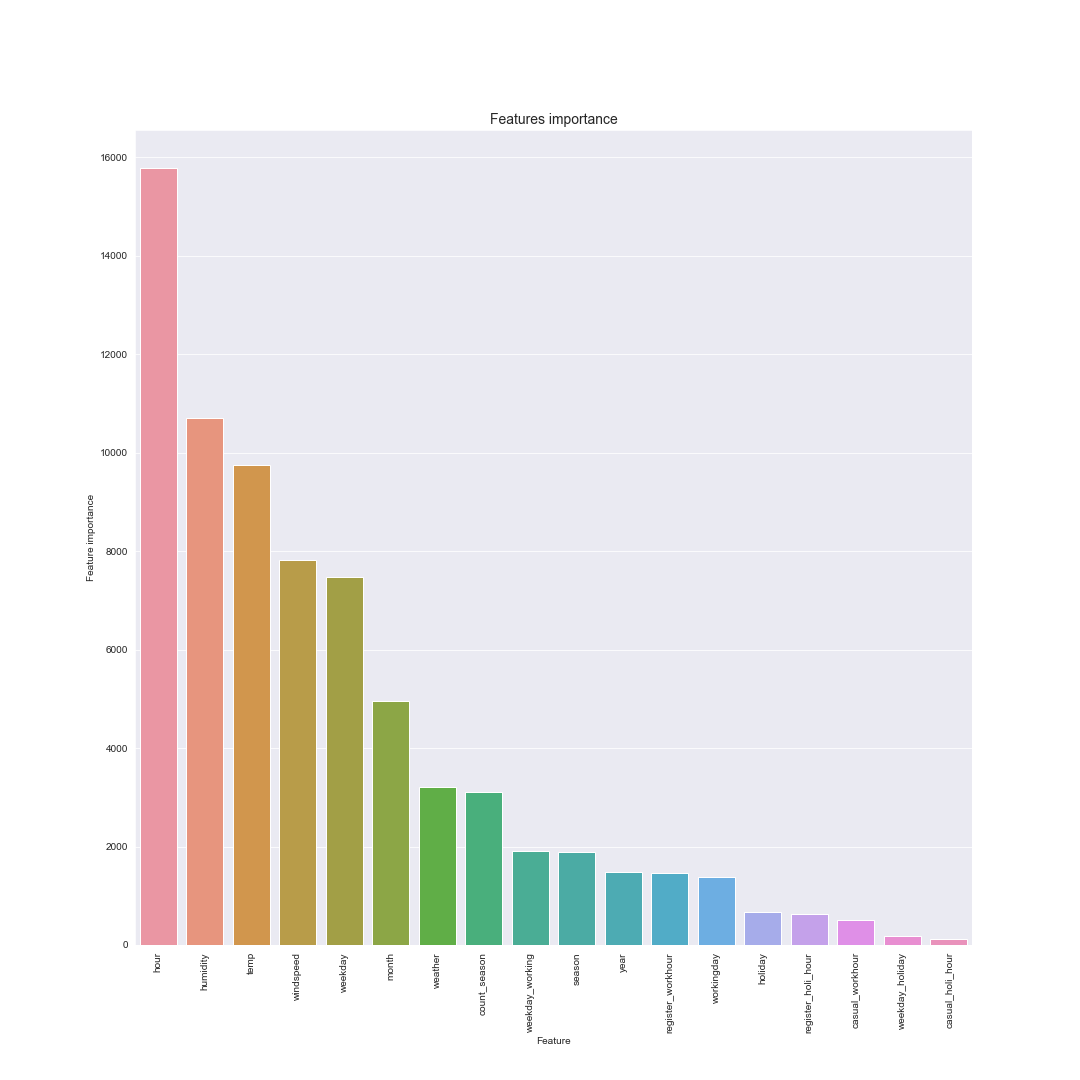
**Heatmap**

****

temp and atemp have high correlation and register and have too. And windspeed and outcomes have low correlation(<=0.1)



**Feature selection**

****

## Summary

# In this project we applied various machine learning algorithms to predict the number of member leases, the number of temporary leases and the total number of leases through the feature values and using TabularPredictor from AutoGluon library to predict the above mentioned problem then using feature selections and hyperparameters we tried to improve our model performance. And after using various techniques we successfully achieved an accuracy score of 35.071134. On participating in the kaggle competition “Bike Sharing Demand”. The best kaggle score achieved till now is 0.45939.