Bike Sharing Demand Prediction - Regression Model

Rajakumaran S
Alma Better

Abstract:

Bike sharing market is growing all over the world these years. It is meaningful to do some analysis about it because more and more companies are getting into this business. Since bike sharing is a recent phenomenon, not that many related analyses have been done upon this topic. However, more and more bike sharing companies have started to realize the importance of data driven decision making. One of the major aspects that can be addressed by data analysis is to predict the demand of bikes on any given day. Knowing the demand would help us in creating a better supply and subsequently reduce the gap between supply and demand.

Our EDA can make us understand data which variable is very important and check how every variable connected with dependent variable.

We make some models to predict the label column based on features are given.

Keywords: *- Exploratory data analysis, Correlation Analysis, Bike Sharing Demand Prediction, Regression analysis.*

1. INTRODUCTION

Bike Sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city. Using these systems, people are able rent a bike from a one location and return it to a different place on an as-needed basis.

The first bike-share programs began in 1960s Europe, but the concept did not take off worldwide until the mid-2000s. In North America, they tend to be affiliated with municipal governments, though some programs, particularly in small college towns, center on university campuses.

The typical bike-share has several defining characteristics and features, including station-based bikes and payment systems, membership, and pass fees, and per- hour usage fees. Programs are generally intuitive enough for novice users to understand. And, despite some variation, the differences are usually small enough to prevent confusion when a regular user of one city's bike-share uses another city's program for the first time.

With the onset of Industry 4.0, integration of Internet of Things (IoT) systems with bike-sharing ecosystem has eased the rental process to a significant extent. Real-time tracking of bikes, traffic density, and climate variables aids in gaining useful knowledge about trends, and patterns of renting process, thereby allowing an incisive prediction to meet future demand.

Considering the current ecosystem, bike-sharing can play a vital role in reducing the impact of carbon emissions and other greenhouse gases- major contributors in climate change. Sustainable and clean transport system, if successful, can provide a greener alternative to the traditional car-pool system, and help in reducing traffic congestion, too.

In addition to the environmental benefits, the sharing systems will impart healthier habits among commuting public, who in the hustle of tasking daily routine, often are unable to integrate optimum level of physical activity, which results in a barrage of ailments.

On a positive note, the global Bike-Sharing market size, which was sized at USD 2570.9 million in 2019, is expected to breach the USD 13780 million mark by 2026, with Compound Annual Growth Rate (CAGR) of 26.8% during 2021-2026, as per Market Analysis via MarketWatch.

For our project, we retrieved data from UCI Machine Learning Repository. The dataset contained per day Bike Rental Count with 8760 entries, possessing 14 attributes, out of which 13 variables-12 independent and one dependent- form the part of our Regression Analysis. The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information.

Date does not provide relevant information to generate a model to predict the Rental Bike Count.

The primary objective was to build a superior statistical model to predict the number of bicycles that can be rented with the availability of data and understand the trends and factors affecting the rented bike count on a particular day.

2. STEPS INVOLVED:

a. Performing EDA (exploratory data analysis)
b. Drawing conclusions from the data Training the model
c. Transforming data using MinMaxScaler.
d. Evaluating metrics of our model

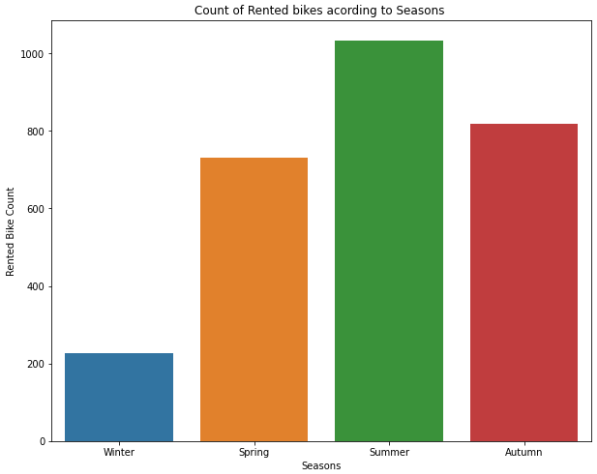
a. Performing EDA (exploratory data analysis):

- Exploring head and tail of the data to get insights on
- the given data.
- Looking for null values and removing them if it affects the performance of the model.
- Converting the data into appropriate data types to create a regression model.
- Creating data frames which help in drawing insights from the dataset.
- Creating more columns in our dataset which would be helpful for creating model.
- Encoding the string type data to better fit our regression model.
- Extracting correlation heatmap and calculating VIF to remove correlated and multicollinear variables.

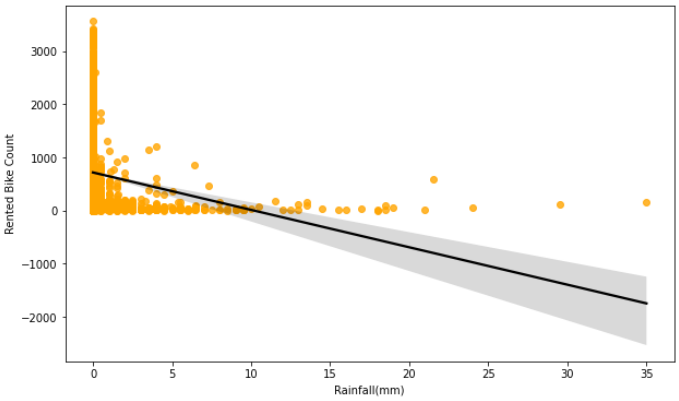
b. Drawing conclusions from the data:

Plotting necessary graphs which provides relevant information on our data like:

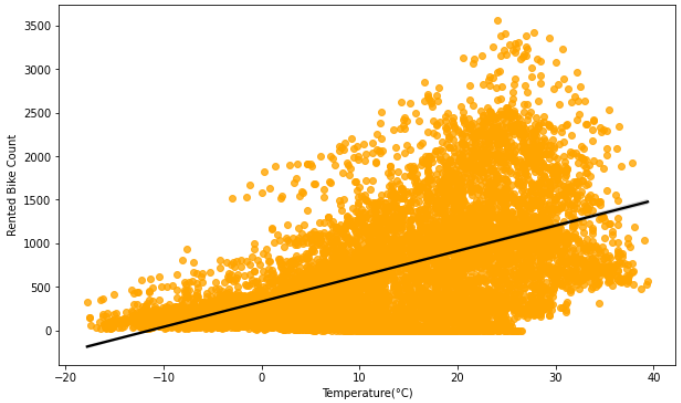
Most bikes have been rented in the summer season and Least bike rent count is in the winter season. autumn and spring seasons have almost equal amounts of bike rent count.



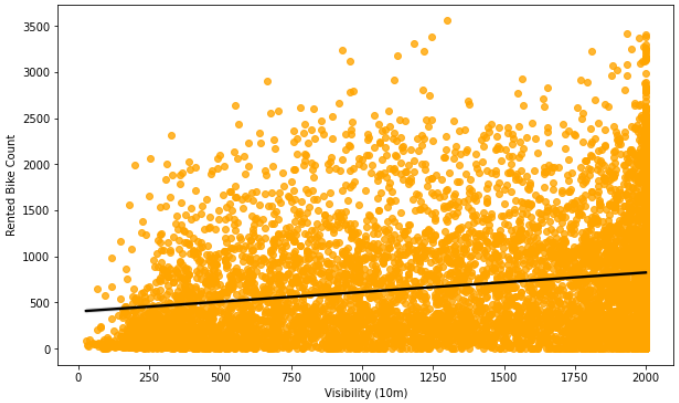
People tend to rent bikes when there are no or less rainfall.



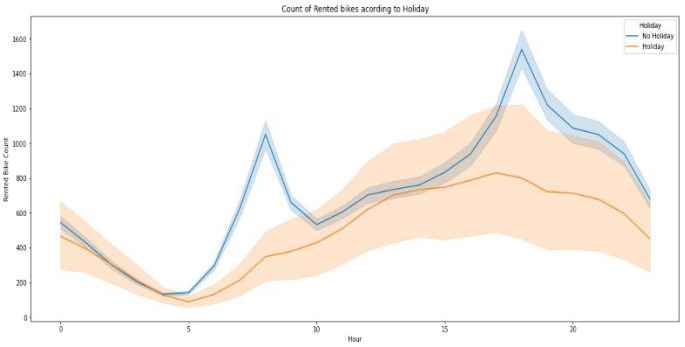
People tend to rent bikes when the temperature is between -5 to 25 degrees.



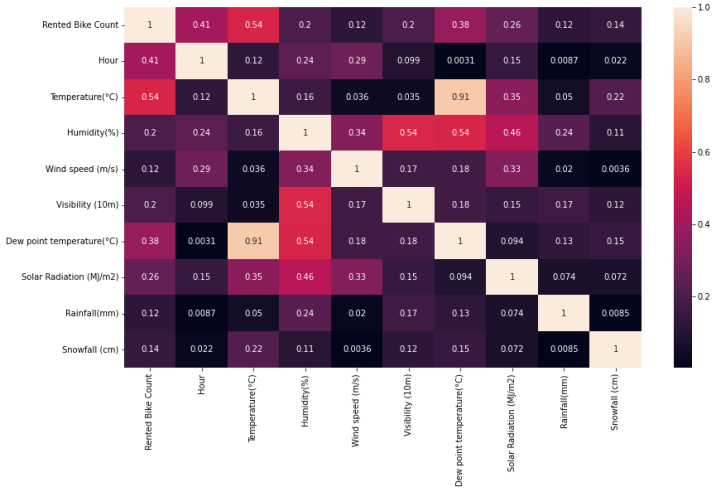
People tend to rent bikes when the visibility is between 300 to 1700.



The rentals were more in the morning and evening times in No Holiday days. This is because people not having personal vehicle, commuting to offices and schools tend to rent bikes.



Correlation Analysis



Correlational Analysis of the dataset variables from the above Correlation graph, we can observe that Temperature and Dew Point Temperature are highly correlated, thereby one of the variables would have to be removed from our Regression model, depending on the significance of each variable.

	variables	VIF
0	Hour	4.418242
1	Temperature(°C)	33.385256
2	Humidity(%)	5.371996
3	Wind speed (m/s)	4.805364
4	Visibility (10m)	9.085977
5	Dew point temperature(°C)	17.126199
6	Solar Radiation (MJ/m2)	2.881590
7	Rainfall(mm)	1.081567
8	Snowfall (cm)	1.120833

From the result of variance influence Factor of Temperature have high VIF value. So we removed temperature from our dataset.

c. Training the model:

- Assigning the dependent and independent variables.
- Splitting the model into train and test sets.
- Transforming data using MinMaxScaler.
- Fitting linear regression on train set.
- Getting the predicted dependent variable values from the model.

d. Evaluating metrics of our model:

Getting MAE, RMSE, R2-SCORE for different models used.

- MAE - It is calculated by taking the absolute difference between the predicted values and the actual values and averaging it across the dataset
- RMSE - Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are.
- R2-SCORE - R-squared (R2) is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model.

Comparing the r2 score of all models used, to get the desired prediction.

3. RESULTS AND DISCUSSIONS :

With Training Dataset having 7008 rows and 14 Columns (Features) and Testing dataset having 1752 rows and 14 columns (Features).

Final Result of Fitting all models:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
0	LinearRegression:	5.59	5.66	0.65	0.65	0.81	0.80
1	Lasso:	8.50	8.69	0.25	0.23	0.50	0.48
2	Ridge:	5.60	5.65	0.65	0.65	0.81	0.81
3	ElasticNet:	9.43	9.55	0.15	0.13	0.38	0.37
4	DecisionTreeRegressor:	0.00	3.72	1.00	0.80	1.00	0.90
5	RandomForestRegressor:	1.02	2.88	0.99	0.88	0.99	0.94
6	GradientBoostingRegressor:	3.32	3.51	0.87	0.86	0.93	0.92
7	XGBRegressor:	3.27	3.47	0.87	0.86	0.93	0.93
8	Lasso_Hyper_tuned:	5.87	6.00	0.62	0.62	0.79	0.79
9	Ridge_Hyper_tuned:	5.59	5.66	0.65	0.65	0.81	0.81
10	RandomForestReg_Hyper_tuned:	2.64	3.23	0.91	0.86	0.95	0.93
11	XGBRegressor_Hyper_tuned:	1.55	2.73	0.97	0.90	0.98	0.95
12	GradientBoostingReg_Hyper_tuned	2.57	3.00	0.91	0.89	0.96	0.94

3.1 Linear Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	LinearRegression:	5.59	5.66	0.65	0.65	0.81	0.80

On testing data R2 score is 0.65(65%) which is same to training data R2 score. Hence, we can say that our model performance is good, and overfitting is not observed. We need to improve our model performance.

3.2 Lasso Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	Lasso:	8.50	8.69	0.25	0.23	0.50	0.48
	Lasso_Hyper_tuned:	5.87	6.00	0.62	0.62	0.79	0.79

On testing data R2 score is 0.25(25%) and training data R2 score 0.23(23%). when we apply hyper parameter tuning on testing data R2 score is 0. 62(62%) which is same to training data R2 score. Hence, we can say that our model performance is good, and overfitting is not observed.

3.3 Ridge Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	Ridge:	5.60	5.65	0.65	0.65	0.81	0.81
	Ridge_Hyper_tuned:	5.59	5.66	0.65	0.65	0.81	0.81

- R2 Score 0.65(65%)

Best params are applied on the above ridge model. (Score on ridge is after hyperparameter tuning same.)

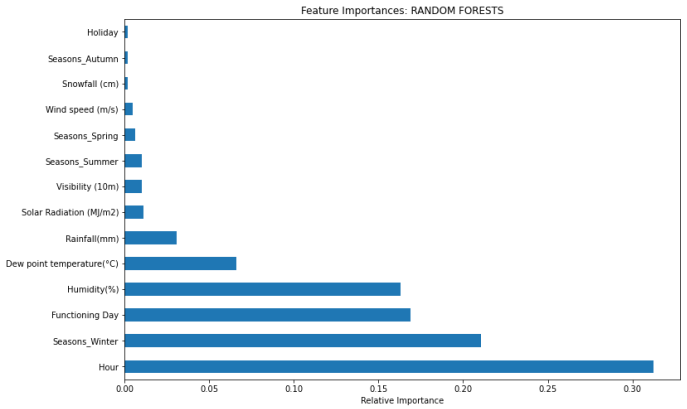
3.4 Decision Tree Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	DecisionTreeRegressor:	0.00	3.72	1.00	0.80	1.00	0.90

We have applied this best parameter to above Decision tree regressor model. overfitting is not observed.

3.5 Random Forest Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	RandomForestRegressor:	1.02	2.88	0.99	0.88	0.99	0.94
	RandomForestReg_Hyper_tuned:	2.64	3.23	0.91	0.86	0.95	0.93

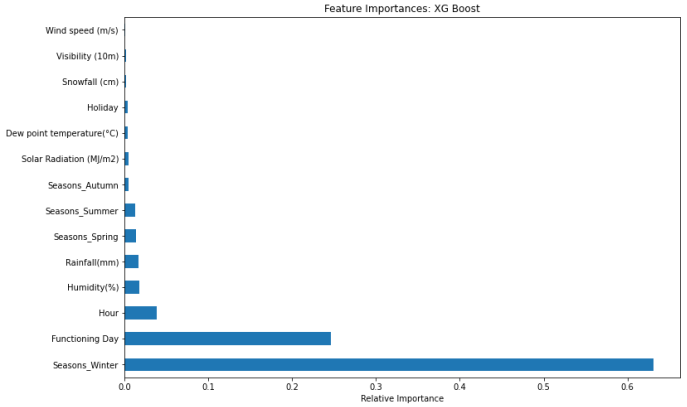


On testing data R2 score is 0.88(88%) and training data R2 score 0.99(99%). when we apply hyper parameter tuning on testing data R2 score is 0.86(86%) and training data R2 score is 0.91(91%). Hence, we can say that our model performance is good, and overfitting is observed.

By looking at the graph we can say that Hours and Seasons_Winter plays very important role on bike rentals and then Functioning day, Dew Point Temperature and so on.

3.6 XG Boost Regression:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
	XGBRegressor:	3.27	3.47	0.87	0.86	0.93	0.93
	XGBRegressor_Hyper_tuned:	1.55	2.73	0.97	0.90	0.98	0.95

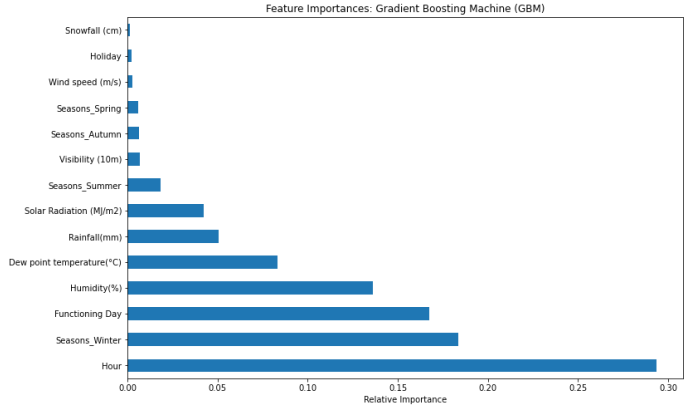


On testing data R2 score is 0.86 and training data R2 score 0.87. when we apply hyper parameter tuning on testing data R2 score is 0.90 and training data R2 score is 0.97. Hence, we can say that our model performance is good, and overfitting is observed.

By looking at the graph we can say that Seasons_Winter and Functioning day plays very important role on bike rentals and then Hours, Humidity and so on.

3.7 Gradient Boosting Machine:

	name	MAE_train	MAE_test	r2_score_train	r2_score_test	RMSE score_train	RMSE score_test
GradientBoostingRegressor:		3.32	3.51	0.87	0.86	0.93	0.92
GradientBoostingReg_Hyper_tuned		2.57	3.00	0.91	0.89	0.96	0.94



On testing data R2 score is 0.86(86%) and training data R2 score 0.87(87%). when we apply hyper parameter tuning on testing data R2 score is 0.89(89%) and training data R2 score is 0.91(91%). Hence, we can say that our model performance is good, and overfitting is not observed.

By looking at the graph we can say that Hours and Seasons_Winter plays very important role on bike rentals and then Functioning day, Dew Point Temperature and so on.

4. CONCLUSION

As we have calculated MAE, RMSE and R2 score for each model. Based on r2 score will decide our model performance.

Our assumption: if the difference of R2 score between Train data and Test is more than 5 % we will consider it as overfitting.

Regression models with low R-squared values can be perfectly good models for several reasons.

Linear, Lasso, Ridge and ElasticNet:

From The above data frame, we can see that linear, Lasso, Ridge and Elastic regression models have almost similar R2 scores (65%) on both training and test data.(Even after using GridSearchCV we have got similar results as of base models).

Decision Tree Regression:

On Decision tree regressor model, without hyperparameter tuning we got r2 score as 100% on training data and on test data (80%) it was very less compared to training r2 score. Thus, our model memorized the data. So, it was a overfitted model.

Random Forest:

On Random Forest regressor model, without hyperparameter tuning we got r2 score as 99% on training data and 88% on test data. Thus, our model memorized the data. So, it was a overfitted model, as per our assumption

After hyperparameter tuning we got r2 score as 91% on training data and 86% on test data which is very good for us.

Gradient Boosting Regression (Gradient Boosting Machine):

On Gradient Boosting Regression, without hyperparameter tuning we got r2 score as 87% on training data and 86% on test data. Our model performed well without hyperparameter tuning.

After hyperparameter tuning we got r2 score as 91% on training data and 89% on test data, thus we improved the model performance by hyperparameter tuning.

Thus, ***Gradient Boosting Regression*** (GridSearchCV) and ***Random Forest Regression*** (GridSearchCV) gives good r2 scores Compared to other models. We can deploy this model.

References:

1. *Alma better*
2. *Stack overflow*
3. *Geeks for Geeks*
4. *w3school*
5. *https://docs.python.org/*
6. *https://scikit-learn.org/*