Project – Python for data analysis

Analyse of the Seoul Bike Sharing Demand Data Set

Arthur Bonneaud
Arthur Bertrand
DIA2

Part 1 : Data-Visualisation

I/ Data-Set Information

Currently Rental bikes are introduced in many urban cities for the enhancement of mobility comfort. 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 becomes a major concern.

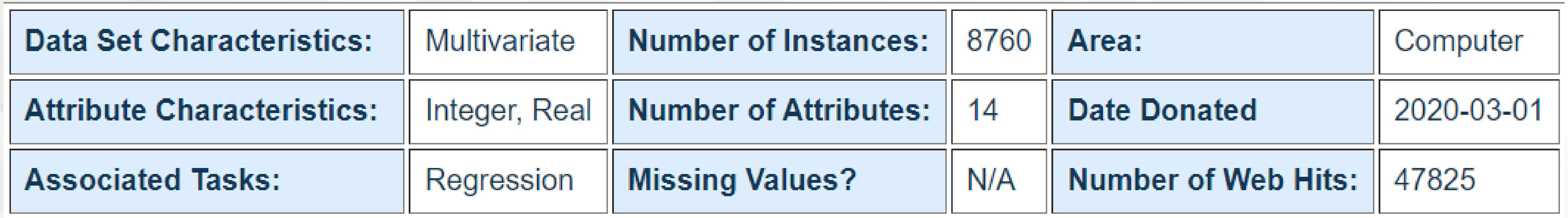
The crucial part is the prediction of bike count required at each hour for the stable supply of rental bikes.

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

Data Source : http://data.seoul.go.kr/
SOUTH KOREA PUBLIC HOLIDAYS. URL: publicholidays.go.kr

Attibute 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)

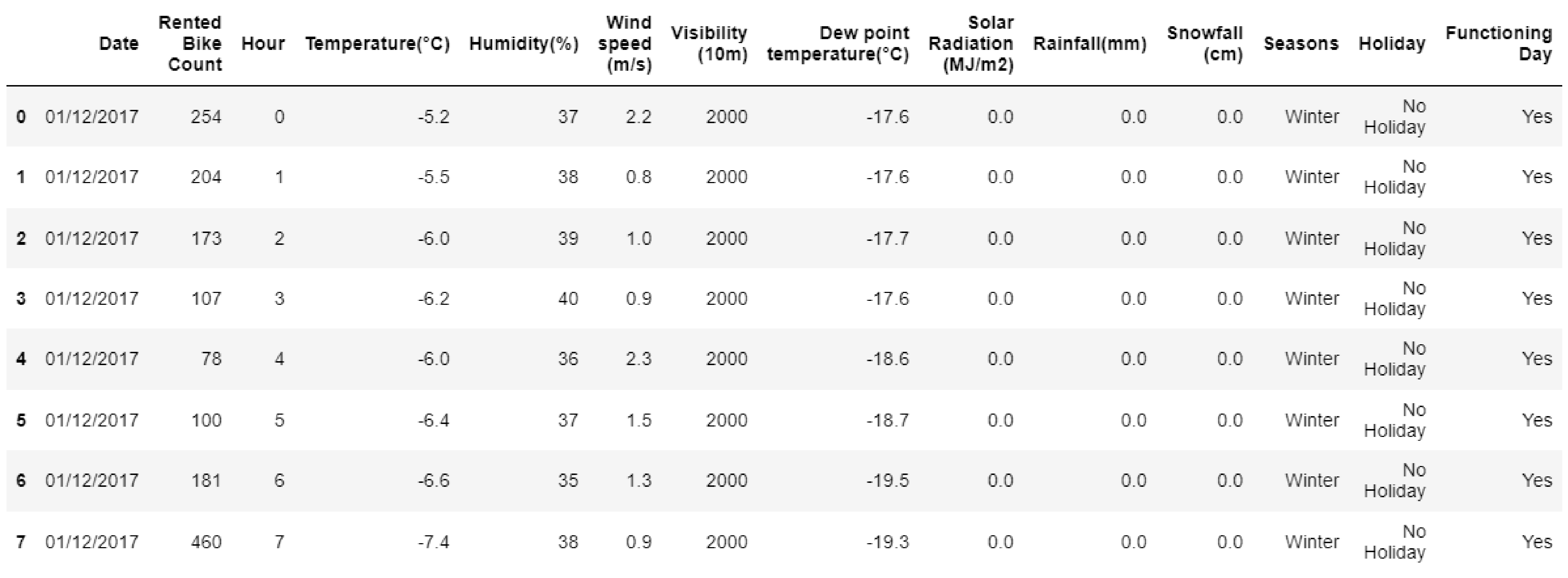


Data Set Characteristics:	Multivariate	**Number of Instances:**	8760	**Area:**	Computer
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	14	**Date Donated**	2020-03-01
Associated Tasks:	Regression	**Missing Values?**	N/A	**Number of Web Hits:**	47825

Our goal is to create a model that predicts the number of bikes rented based on the attributes given in the dataset. First, we need to visualize the data to know what attributes correlate with that target.

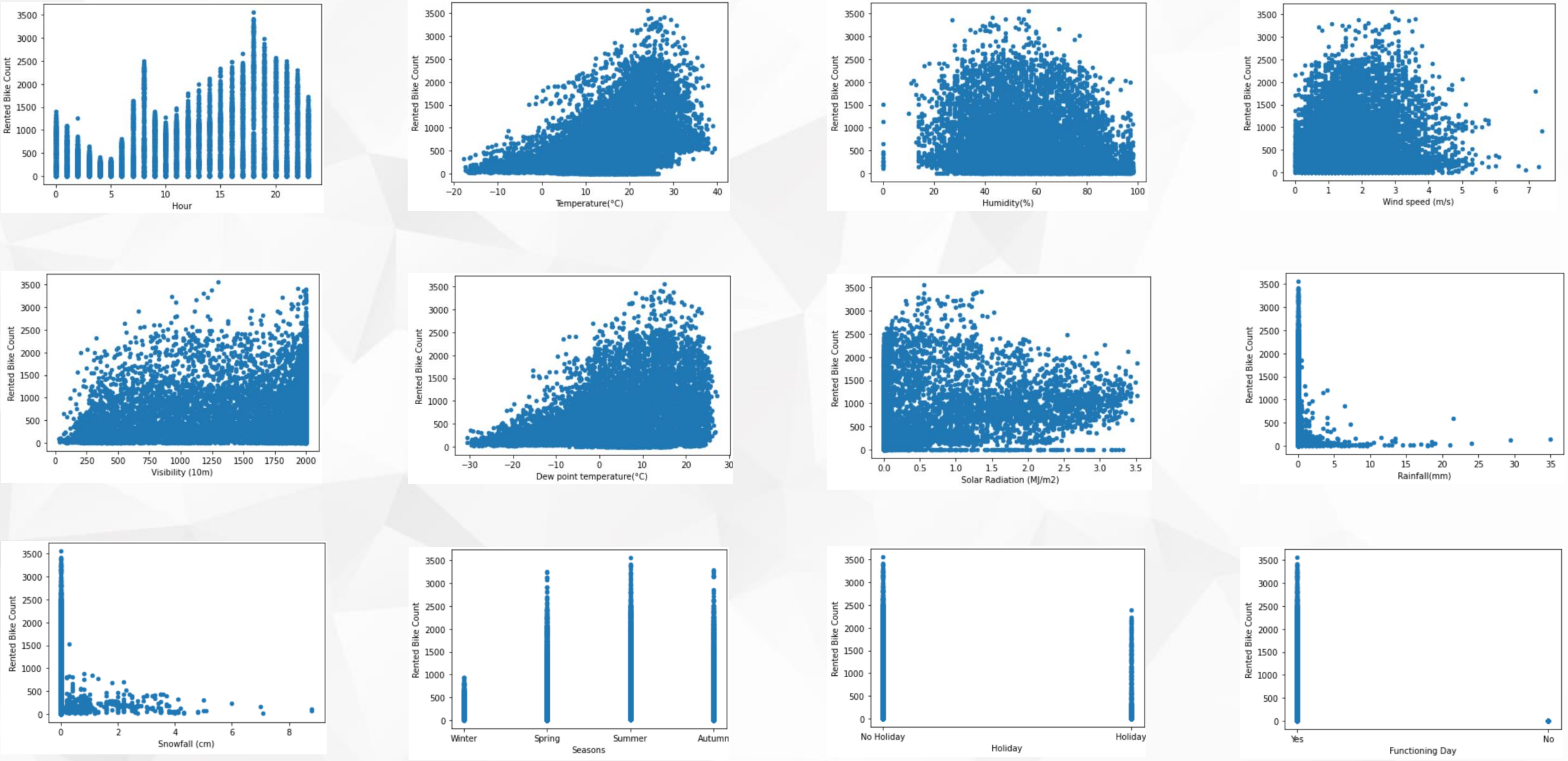
To Visualize the data, we need to plot some graphs.

Here is the head of the dataset :

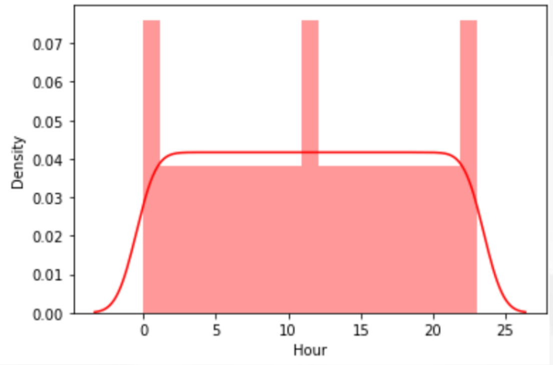


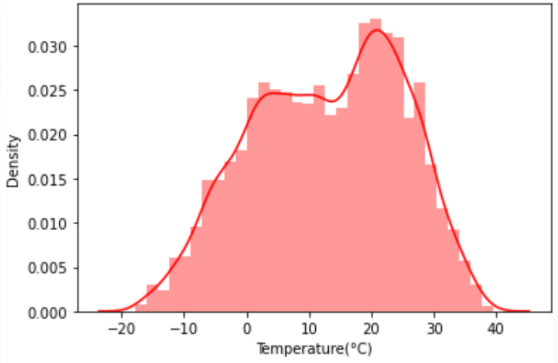
	Date	Rented Bike Count	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m2)	Rainfall(mm)	Snowfall (cm)	Seasons	Holiday	Functioning Day
0	01/12/2017	254	0	-5.2	37	2.2	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
1	01/12/2017	204	1	-5.5	38	0.8	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
2	01/12/2017	173	2	-6.0	39	1.0	2000	-17.7	0.0	0.0	0.0	Winter	No Holiday	Yes
3	01/12/2017	107	3	-6.2	40	0.9	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
4	01/12/2017	78	4	-6.0	36	2.3	2000	-18.6	0.0	0.0	0.0	Winter	No Holiday	Yes
5	01/12/2017	100	5	-6.4	37	1.5	2000	-18.7	0.0	0.0	0.0	Winter	No Holiday	Yes
6	01/12/2017	181	6	-6.6	35	1.3	2000	-19.5	0.0	0.0	0.0	Winter	No Holiday	Yes
7	01/12/2017	460	7	-7.4	38	0.9	2000	-19.3	0.0	0.0	0.0	Winter	No Holiday	Yes

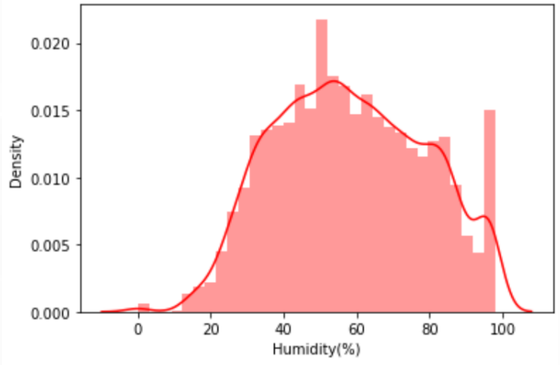
Here are the scatter plots :

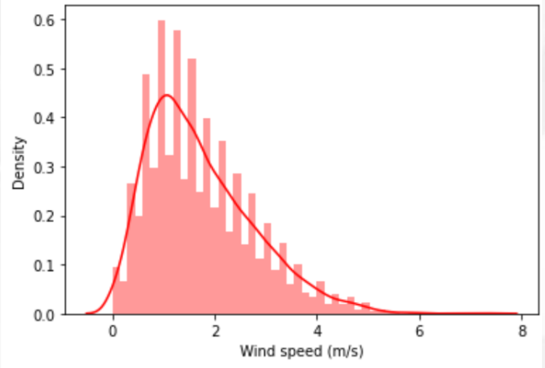


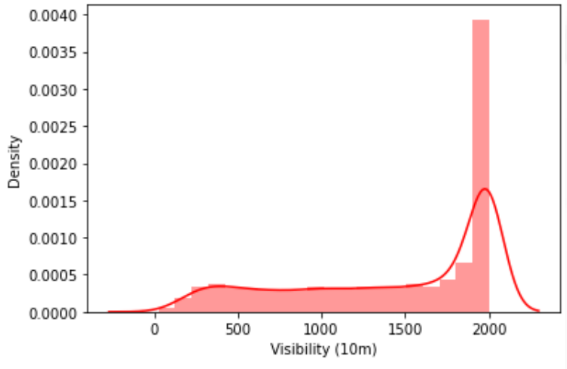
Here is an other form of display to have a better view.

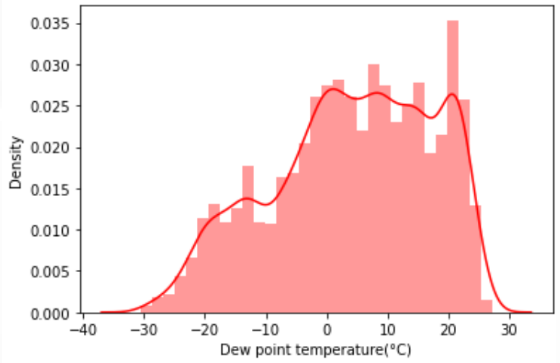


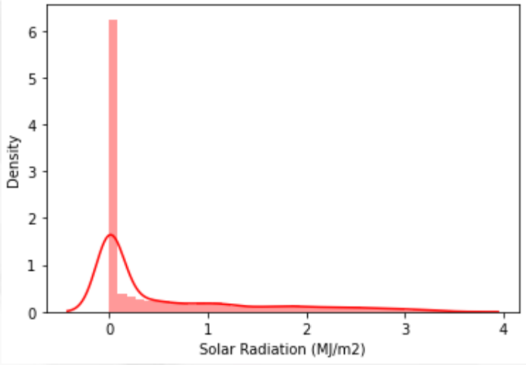


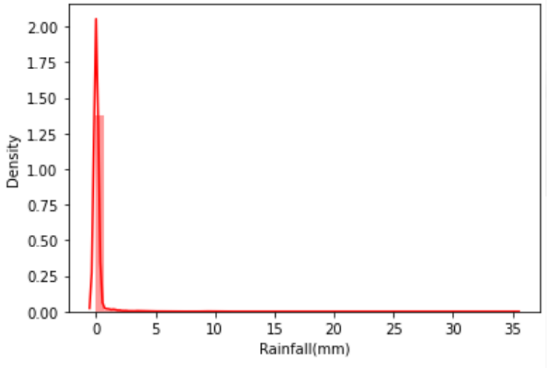


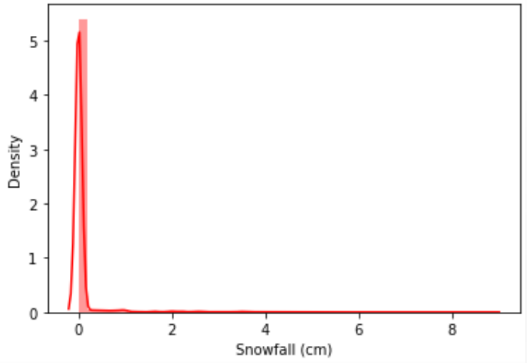




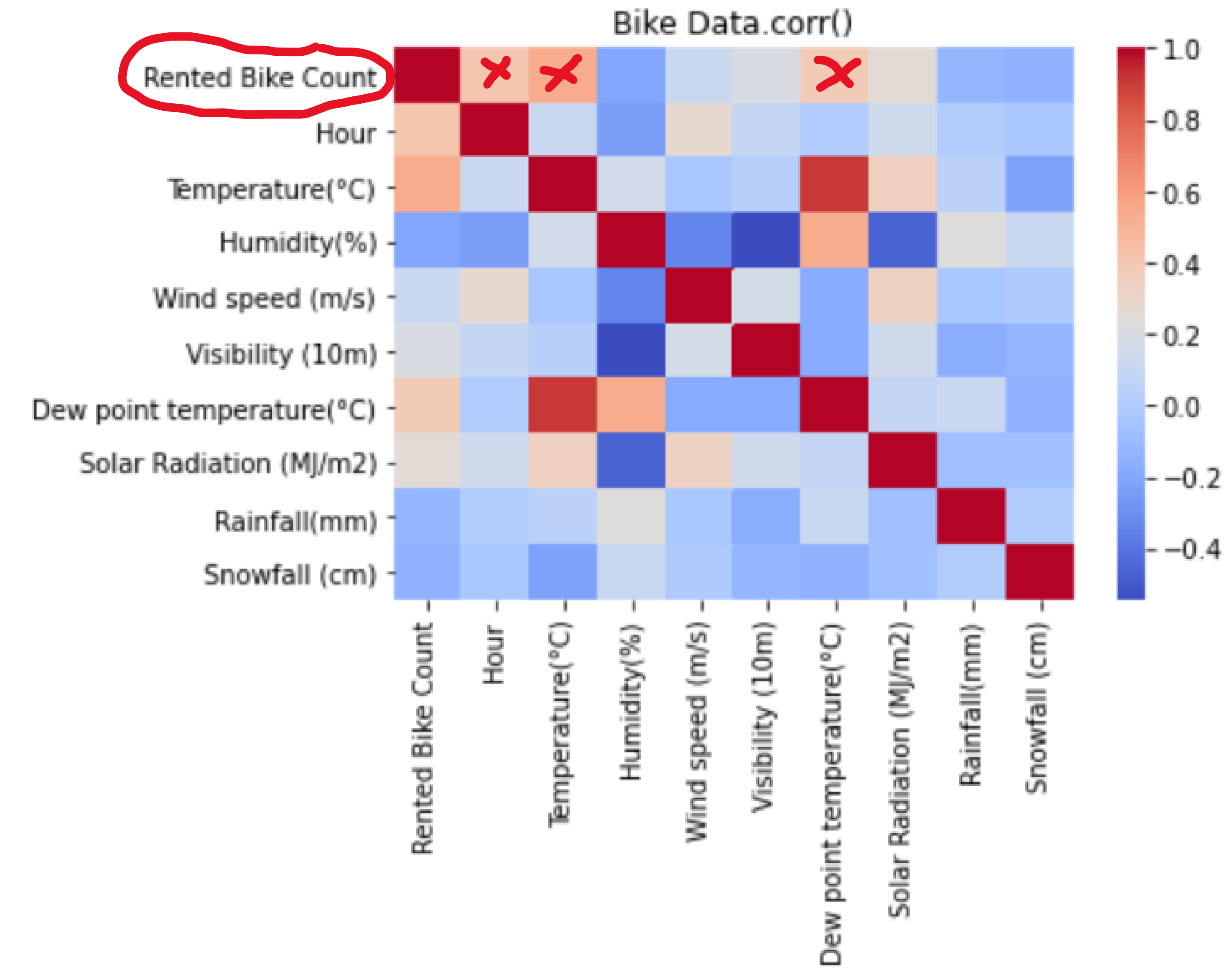








Then here is the correlation matrix to identify the most inluents factors.



The boxes checked in red are the most correlated parameters.

Part 2 : Data-Preparation

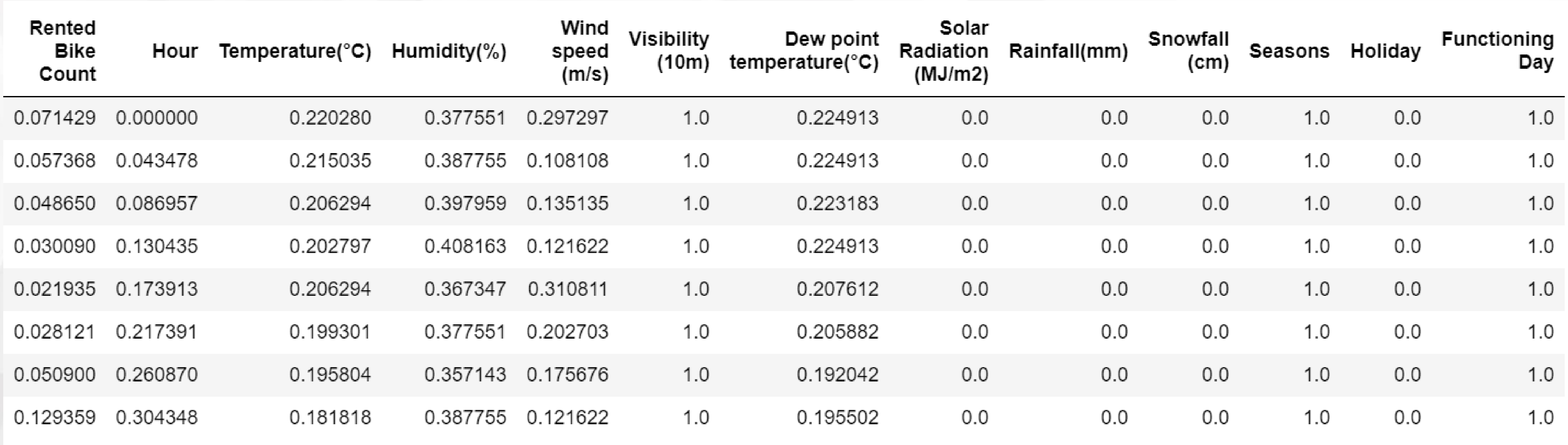
First of all, we need to prepare the dataset so that we don't have problems in training on the different models.

Concretely, it is a question of converting most of the data into digital, choosing whether or not to keep certain columns according to their relevance and perhaps creating new ones.

Moreover, we have to scale the dataset in order to have a better comparison of the values.

Next, we need to separate the dataset into two: a train set and a test set.

Here is the head of the dataset after preparation.



Rented Bike Count	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m2)	Rainfall(mm)	Snowfall (cm)	Seasons	Holiday	Functioning Day
0.071429	0.000000	0.220280	0.377551	0.297297	1.0	0.224913	0.0	0.0	0.0	1.0	0.0	1.0
0.057368	0.043478	0.215035	0.387755	0.108108	1.0	0.224913	0.0	0.0	0.0	1.0	0.0	1.0
0.048650	0.086957	0.206294	0.397959	0.135135	1.0	0.223183	0.0	0.0	0.0	1.0	0.0	1.0
0.030090	0.130435	0.202797	0.408163	0.121622	1.0	0.224913	0.0	0.0	0.0	1.0	0.0	1.0
0.021935	0.173913	0.206294	0.367347	0.310811	1.0	0.207612	0.0	0.0	0.0	1.0	0.0	1.0
0.028121	0.217391	0.199301	0.377551	0.202703	1.0	0.205882	0.0	0.0	0.0	1.0	0.0	1.0
0.050900	0.260870	0.195804	0.357143	0.175676	1.0	0.192042	0.0	0.0	0.0	1.0	0.0	1.0
0.129359	0.304348	0.181818	0.387755	0.121622	1.0	0.195502	0.0	0.0	0.0	1.0	0.0	1.0

Part 3 : Train and test

Then we train the data on the different machine learning models with X_train and y_train.

And in the same time, we test the prediction to obtain the accuracy.

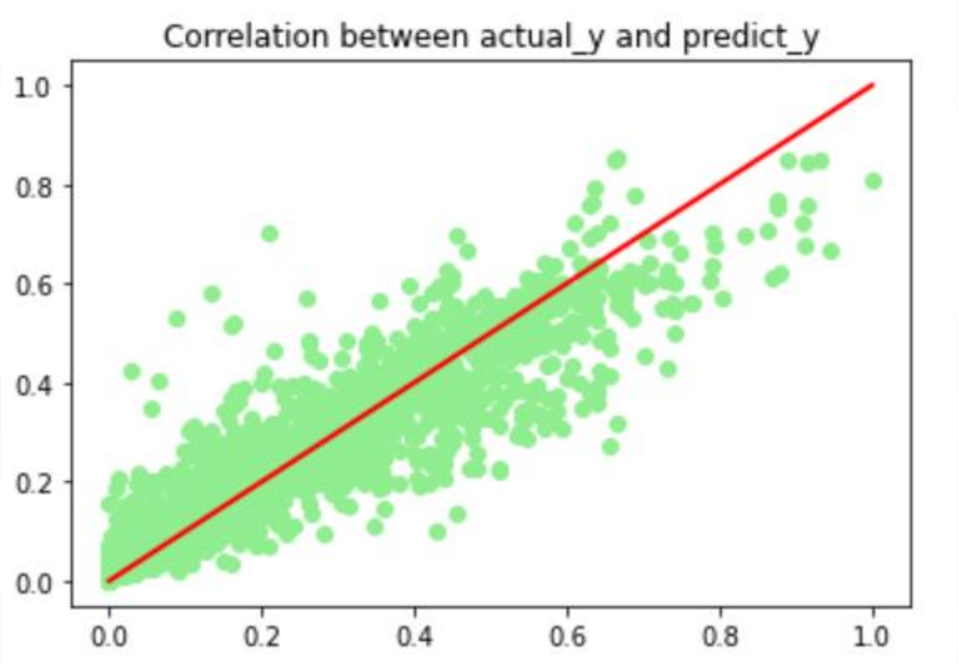
Machine learning models used :

- Linear Regression ➔ acc : 51%
- Random forest ➔ **acc : 87%**
- KNeighbors Regression ➔ acc : 78%
- Decision Tree ➔ acc : 73%

Also, we can use hyper parameters to improve the results.

Conclusion : Best model

To conclude, our best model is random forest with an accuracy of 87%.


Correlation between actual_y and predict_y