Solution of the assignment
BlueYonder GmbH
Data Scientist Position (m/f)

Submitted By – Niraj Dev Pandey

Dear BlueYonder GmbH,

Here is the solution of the provided programming task. This document is to explain the opted approach and to include some graphical plots which will help you to understand the exploratory data analysis part of the solution. The python script is attached which can be seen for crosschecking the outcomes presented here.

Note: There are two scripts, one is about clean code challenge which is BlueYonder.py and another one is just to visualize the data and other statistics called BlueYonder_data.ipynb (dirty code ;) For clean code I have adopted python PEP-8 standard and this includes followings.

1. Correct user defined function naming
2. Choosing clear variable names
3. Helper for all the functions
4. Spaces and pronunciations as per PEP-8 standard
5. Raising clear exceptions
6. Etc.....

Unittest - Standard way in software development is to use python unittest module to verify if everything is working as expected but there is also few check-post we can raise as an exception to let the user know what is wrong in there. Moreover, sklearn.utils module also has some functionality which can be used for unittesting. The *BlueYonder.py* includes sklearn.utils way of checking some functions and the data, raising exceptions are the other thing you can see there. In addition to this, I have tried Python Unittest module for few functions that can be seen in *BikeTest.py* file. In task like regression there is less to test except few user defined functions like, evaluation metric, data-set shapes, one hot encoding functions etc. Nevertheless, without wasting any more time, let's see what this task was all about and dive a bit deeper to see explanatory data analysis.

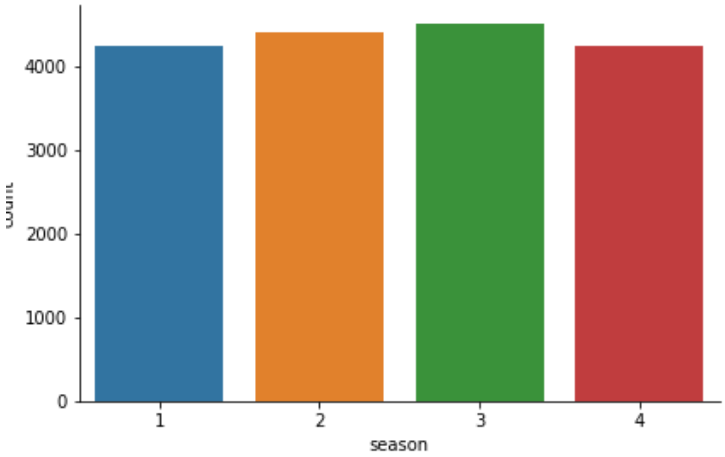
Data-set - Bike-sharing rental process is highly correlated to the environmental and seasonal settings. For instance, weather conditions, precipitation, day of week, season, hour of the day, etc. can affect the rental behaviors. The core data set is related to the two-year historical log corresponding to years 2011 and 2012 from Capital Bikeshare system, Washington D.C., USA which is publicly available.

Task - Regression: Predication of bike rental count hourly or daily based on the environmental and seasonal settings. We are going to see the hourly count prediction.

Here is how data look like.

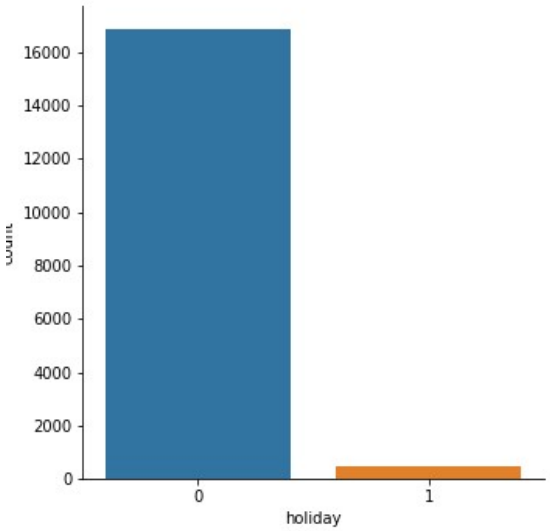
	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
0	1	2011-01-01	1	0	1	0	0	6	0	1	0.24	0.2879	0.81	0.0	3	13	16
1	2	2011-01-01	1	0	1	1	0	6	0	1	0.22	0.2727	0.80	0.0	8	32	40
2	3	2011-01-01	1	0	1	2	0	6	0	1	0.22	0.2727	0.80	0.0	5	27	32
3	4	2011-01-01	1	0	1	3	0	6	0	1	0.24	0.2879	0.75	0.0	3	10	13
4	5	2011-01-01	1	0	1	4	0	6	0	1	0.24	0.2879	0.75	0.0	0	1	1

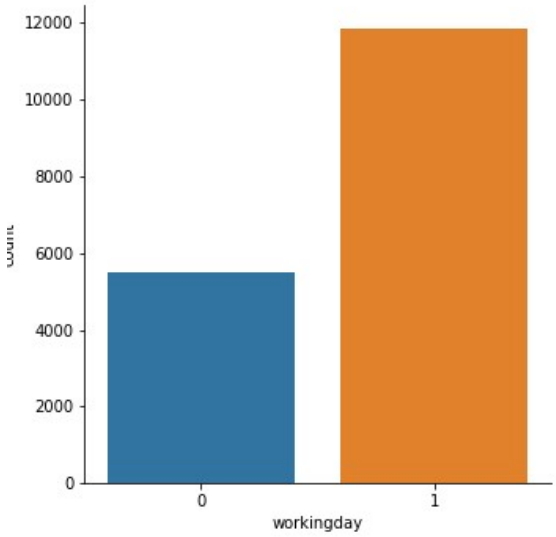
Let’s plot some of the statistics and see more about the dependent and independent (features) variables. Let’s plot the count of all the seasons and see that which season has more count in our data set.



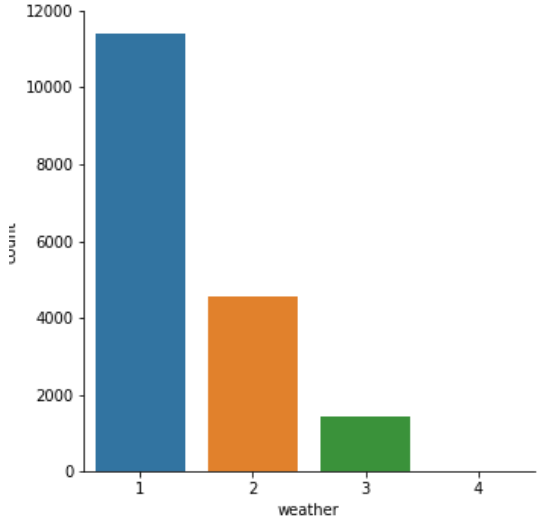
Here we can see that the all 4 seasons have almost equal count.

How about the holidays and working days in the data-set.





Here is two plots showing the number of holidays and working days in the data-set. It is obvious that there would be less holidays in year and more normal days. The plot in the right depicts working days against not working days (Saturday & Sunday).

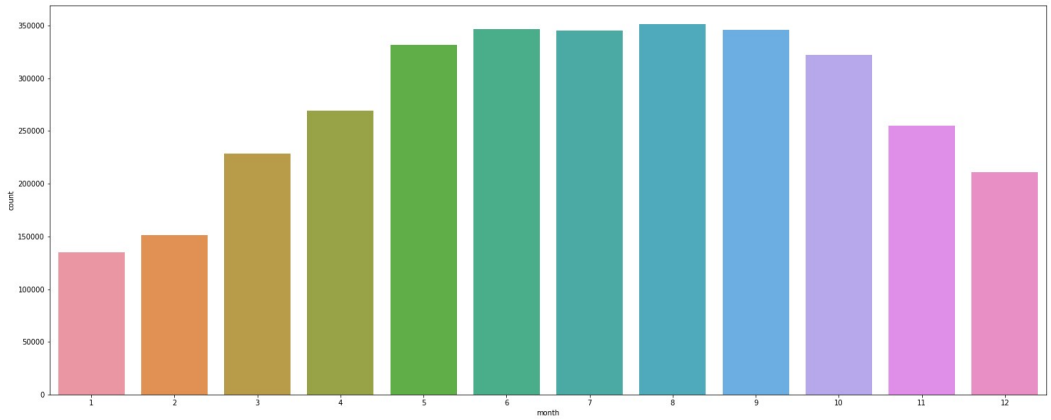


Well, how was the weather in those 2 years? Was is mostly cloudy, sunny etc. Remember that these numbers on the x axis has longer meaning which can be seen in the description below -

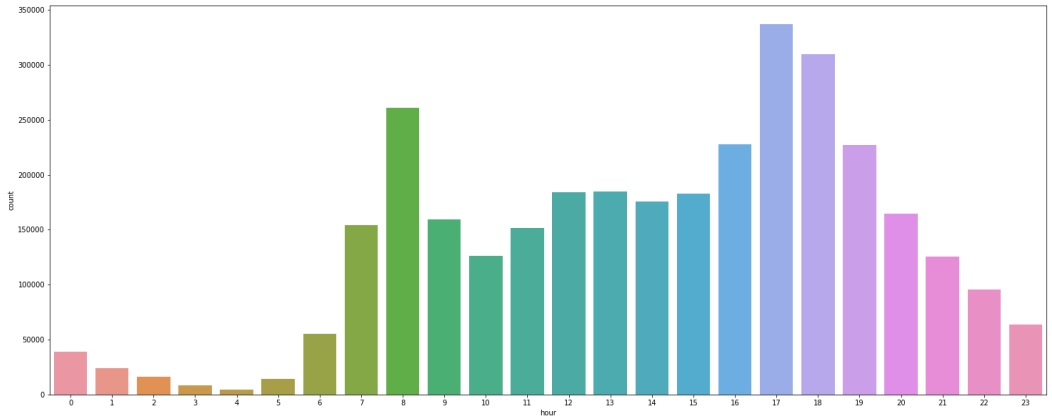
1: Clear, Few clouds, Partly cloudy, Partly cloudy
2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered Cloud
4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog

So, this plot shows that the weather mostly fall in the category of class 1 and 2. Class 4 has almost o count. This means that there were no thunderstorm and Ice pallets etc in those two years of time. Anyways, these weather conditions are rare in weather conditions.

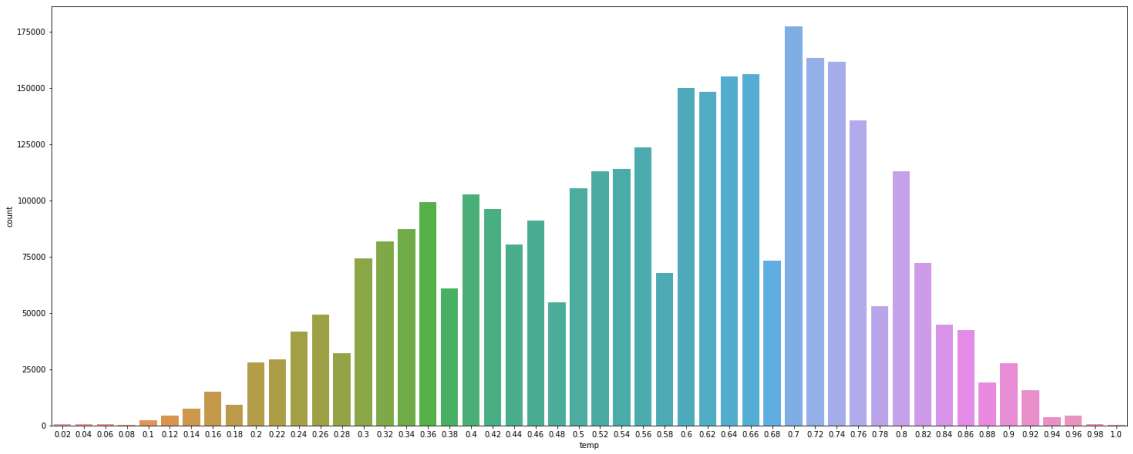
Which month had the highest demand? People like to ride bikes in warmer temperature but not to put my own bias here. Human biases are already in controversy for machine learning and AI models :) let's see the statistics instead generalizing ourselves.



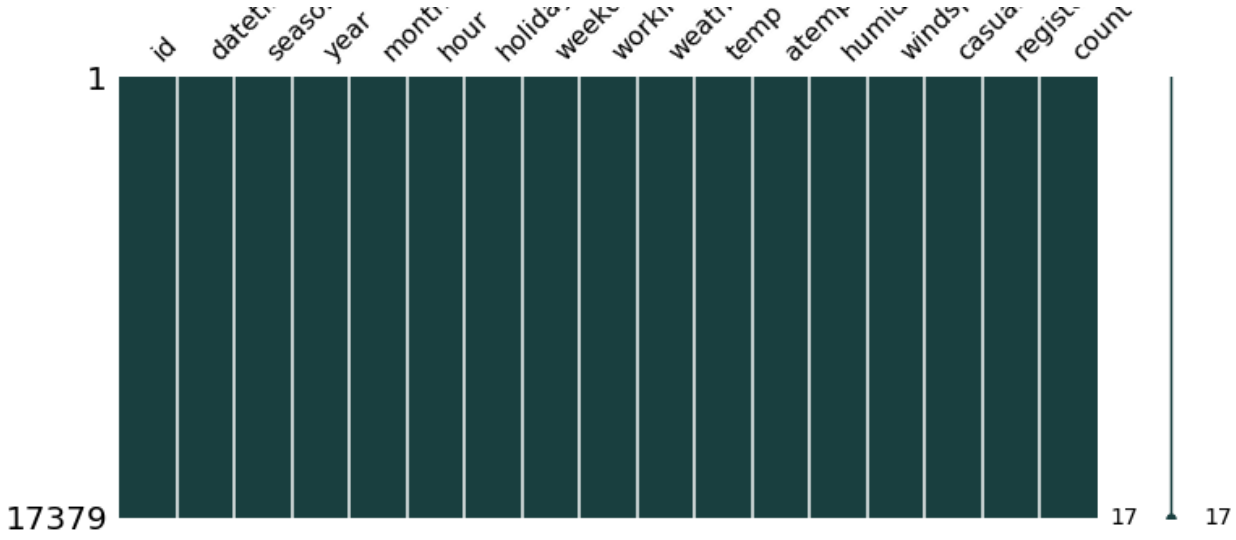
X axis is about 12 months and y axis is having number of count. You can see that the month July to October has very high demand. The highest demand was spotted in the month August. What was peak hour when riders rented bike?



Here x axis is about the number of hour in a day which is 0 to 23 and y axis has number of count. You see, morning 8 O'clock and evening 5-6 got high demand for bikes. Reason could be office commuters basically. Now, which temperature was best preferred by riders to rent a bike?

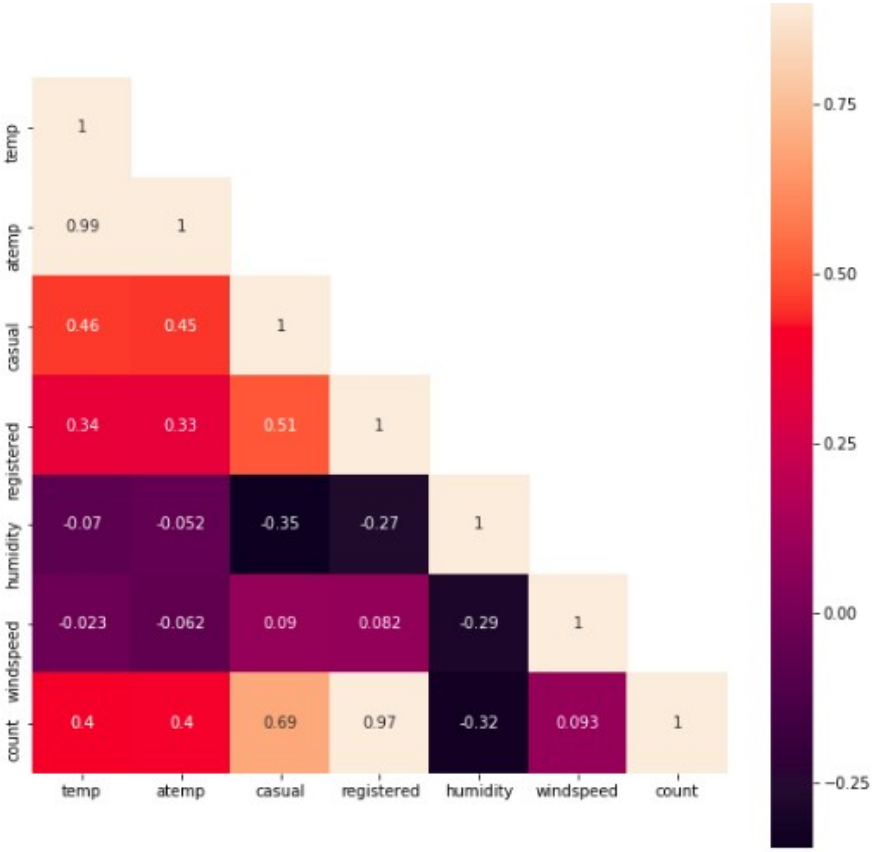


You can see in the picture above that the higher temperature was more preferred for riding. Also, remember that these temperatures are normalizes Celsius values. We also have another feature column called *[atemp]* which is real feel of the recorded temperature. Now, let's see if there is any missing value or Nan in data-set. This will create obstacle if there would be any.



In total we have 17379 rows and 17 columns. This picture shows that there is no empty or Nan value. If found we generally fill that place with 0 but best with the mean of the data points.

Correlation Analysis - To understand how a dependent variable is influenced by features (numerical) is to get a correlation matrix between them. Lets plot a correlation plot between "count" and ["temp","atemp","humidity","windspeed"].



temp and humidity features has got positive and negative correlation with count respectively. Although the correlation between them are not very prominent still the count variable has got little dependency on "temp" and "humidity".

windspeed is not gonna be really useful numerical feature and it is visible from it correlation value with "count"

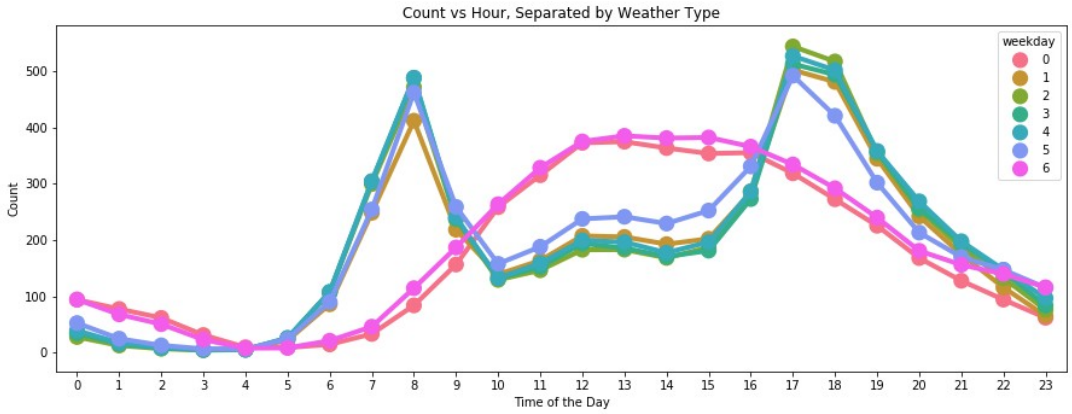
"atemp" is variable is not taken into since "atemp" and "temp" has got strong correlation with each other. During model building any one of the variable has to be dropped since they will exhibit multicollinearity in the data.

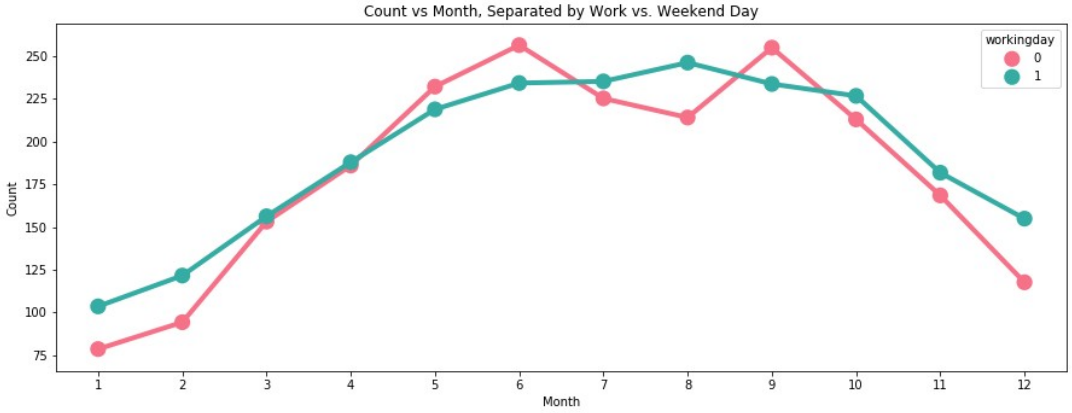
"Casual" or "Registered" are also not taken into account since they are leakage variables in nature and need to dropped during model building.

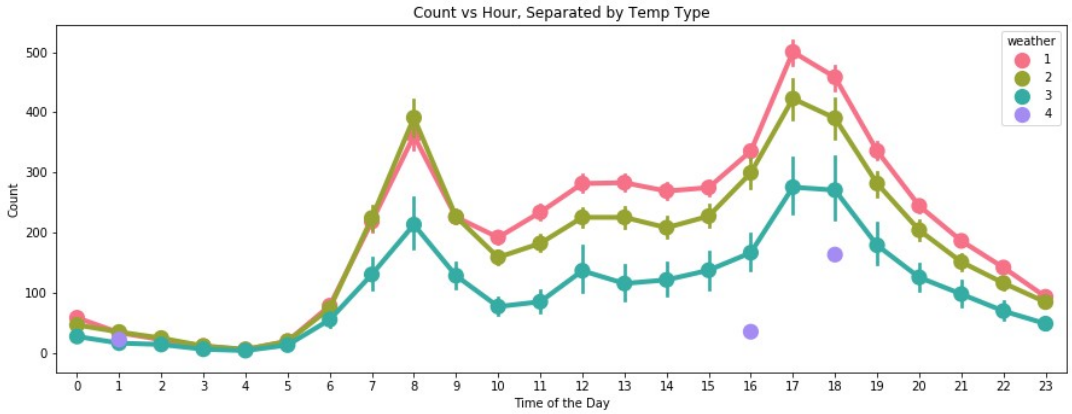
In plot below (next page) It is quiet obvious that people tend to rent bike during summer season since it is really conducive to ride bike at that season. Therefore June, July and August has got relatively higher demand for bicycle.

On weekdays more people tend to rent bicycle around 7AM-8AM and 5PM-6PM. As I mentioned earlier this can be attributed to regular school and office commuters.

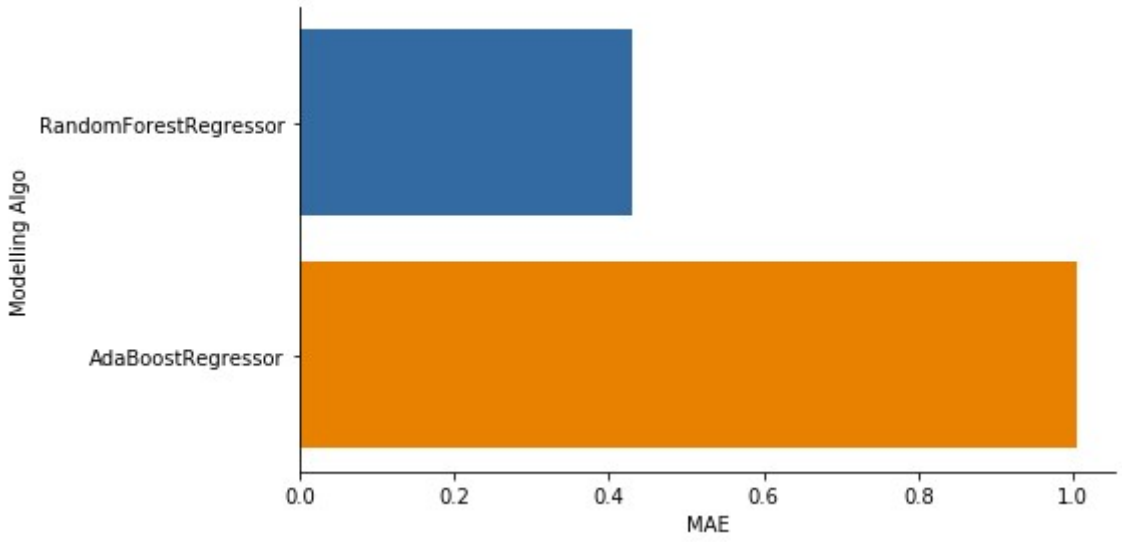
Above pattern is not observed on "Saturday" and "Sunday".More people tend to rent bicycle between 10AM and 4PM.







Model Building - A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the data-set and uses averaging to improve the predictive accuracy and control over-fitting.



• Reasons for choosing this model- There are so many different algorithms do do the same task. Be it regression, classification or any other. We choose models based on their performances. No one knows that how xyz model will perform until you try and test it on your data, having a performance metric. I tried many different algorithm to improve the accuracy of the prediction but RandomForestRegression gave me the least error. That's the reason for choosing the model. Also, if the nature of the data is volatile then you can't stick with one model with same parameters.

•Assume that the code you are writing is used in production in a daily prediction service and maintained by your colleagues (what could that mean?)
This means to me that the code must be clean and so organized that my colleagues will find it easy to follow. Every function must contains a helper argument. Do not name the classes, variable or functions randomly but as such that they contain a clear meaning. PEP-8 is really nice thing to follow.

• What are the scaling properties of your model, if you assume that the amount of data you need to handle go up to several terabytes? Do you see any problems?
Machine Learning models are highly sensitive to the size of the data. If data will go till terabytes then model will still be able to make predictions but given a powerful machine or translate this model with Big Data techniques. Such as, Hadoop, Spark etc.

• How would you address these problems? Are there technologies for data storage/predictive modeling you can build upon?
Spark's RDD is a good solution for such scenario when we can scale the model by using distributed data. But on server another approach could be, to use cloud services. Such as, AWS , Azure, Google cloud etc….

• What are the limits and drawbacks for your new approach?
Scaling the model could be costly process and have limitations too. For example AWS has some limits on, size of the training data, batch prediction, number of variable in a data file etc. Please find all the limits for ML model here.
• Do you have hands-on experience with such technologies? Which ones? For how long?

I have a bit of experience with Big Data technologies. Such as, Pyspark and Hadoop. About cloud, I know Dockers and have little knowledge of AWS instances like S3.

Thanks a lot. It was a simple but good task to learn