Project Name - Bike Renting

Somesh Ugar

28 June 2019

Chapter 1 : Introduction

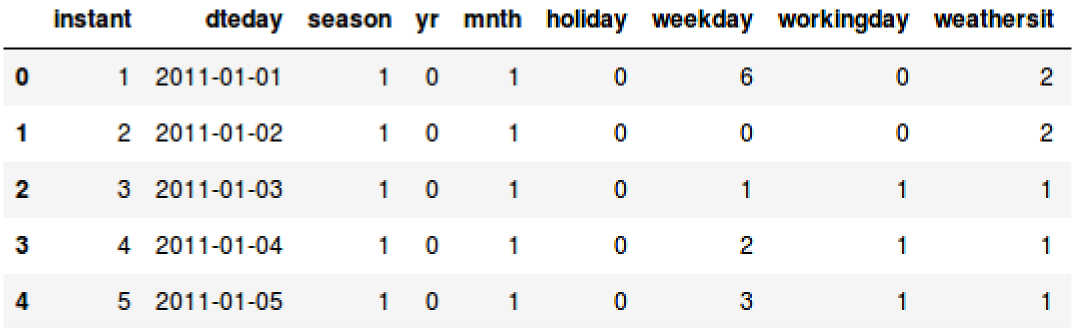
1.1 Problem Statement

The objective of this Case is to Predication of bike rental count on daily based on the environmental and seasonal settings.

1.2 Data

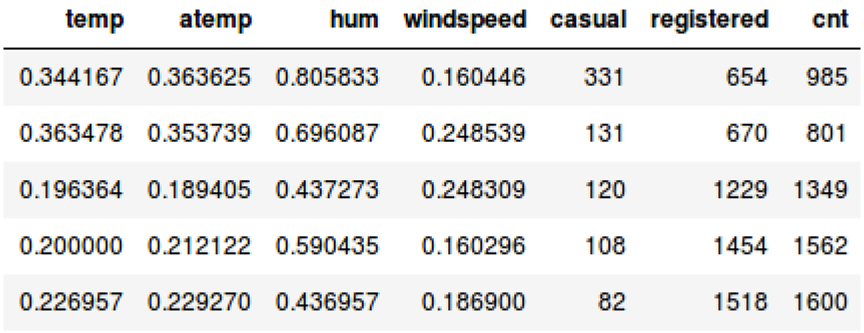
Our task is to build a models which will predict the bike rental count on daily based on the environmental and seasonal settings. Given below is a sample of the data set that we are using to predict the rental count:

Table 1.1: Bike Rental Sample Data (Columns: 1-9)



	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit
0	1	2011-01-01	1	0	1	0	6	0	2
1	2	2011-01-02	1	0	1	0	0	0	2
2	3	2011-01-03	1	0	1	0	1	1	1
3	4	2011-01-04	1	0	1	0	2	1	1
4	5	2011-01-05	1	0	1	0	3	1	1

Table 1.2: Bike Rental Sample Data (Columns: 10-16)



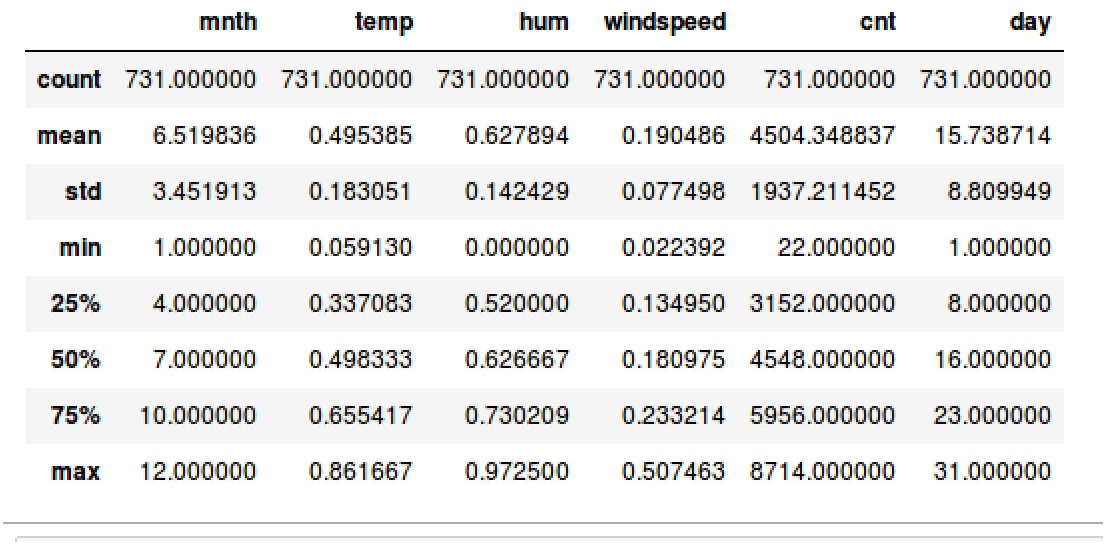
temp	atemp	hum	windspeed	casual	registered	cnt
0.344167	0.363625	0.805833	0.160446	331	654	985
0.363478	0.353739	0.696087	0.248539	131	670	801
0.196364	0.189405	0.437273	0.248309	120	1229	1349
0.200000	0.212122	0.590435	0.160296	108	1454	1562
0.226957	0.229270	0.436957	0.186900	82	1518	1600

Bike-sharing rental process is highly correlated to the environmental and seasonal settings. For instance, weather conditions,temperature, day of the week, season etc. can affect the rental behaviors.
The original data set contains 16 variables and 731 observations in total.

1.3 Data Types of all variables

```
'data.frame':   731 obs. of  16 variables:
 $ instant   : int  1 2 3 4 5 6 7 8 9 10 ...
 $ dteday    : Factor w/ 731 levels "2011-01-01","2011-01-02",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ yr        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ mnth      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ weekday   : int  6 0 1 2 3 4 5 6 0 1 ...
 $ workingday: int  0 0 1 1 1 1 1 0 0 1 ...
 $ weathersit: int  2 2 1 1 1 1 2 2 1 1 ...
 $ temp      : num  0.344 0.363 0.196 0.2 0.227 ...
 $ atemp     : num  0.364 0.354 0.189 0.212 0.229 ...
 $ hum       : num  0.806 0.696 0.437 0.59 0.437 ...
 $ windspeed : num  0.16 0.249 0.248 0.16 0.187 ...
 $ casual    : int  331 131 120 108 82 88 148 68 54 41 ...
 $ registered: int  654 670 1229 1454 1518 1518 1362 891 768 1280 ...
 $ cnt       : int  985 801 1349 1562 1600 1606 1510 959 822 1321 ...
```

1.4 Summary of all variables

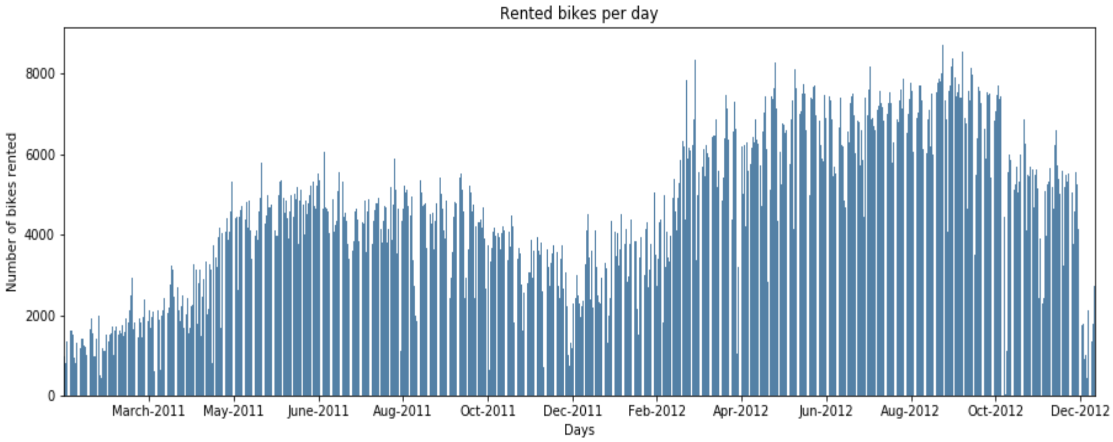


	mnth	temp	hum	windspeed	cnt	day
count	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000
mean	6.519836	0.495385	0.627894	0.190486	4504.348837	15.738714
std	3.451913	0.183051	0.142429	0.077498	1937.211452	8.809949
min	1.000000	0.059130	0.000000	0.022392	22.000000	1.000000
25%	4.000000	0.337083	0.520000	0.134950	3152.000000	8.000000
50%	7.000000	0.498333	0.626667	0.180975	4548.000000	16.000000
75%	10.000000	0.655417	0.730209	0.233214	5956.000000	23.000000
max	12.000000	0.861667	0.972500	0.507463	8714.000000	31.000000

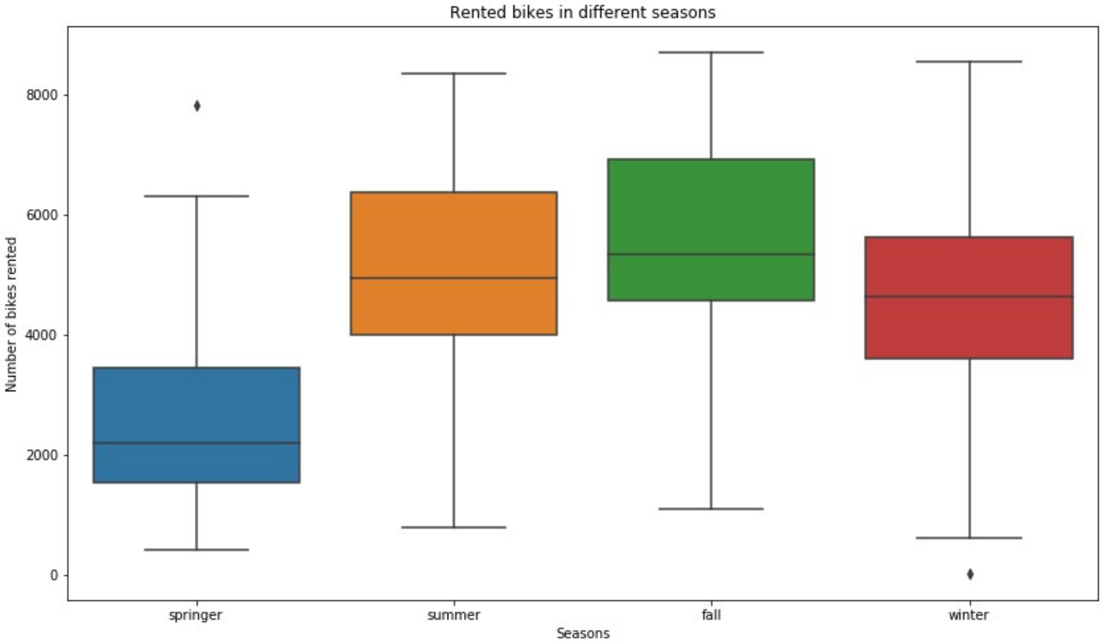
Chapter 2 : Exploratory Data Analysis :

2.1 Data Visualization

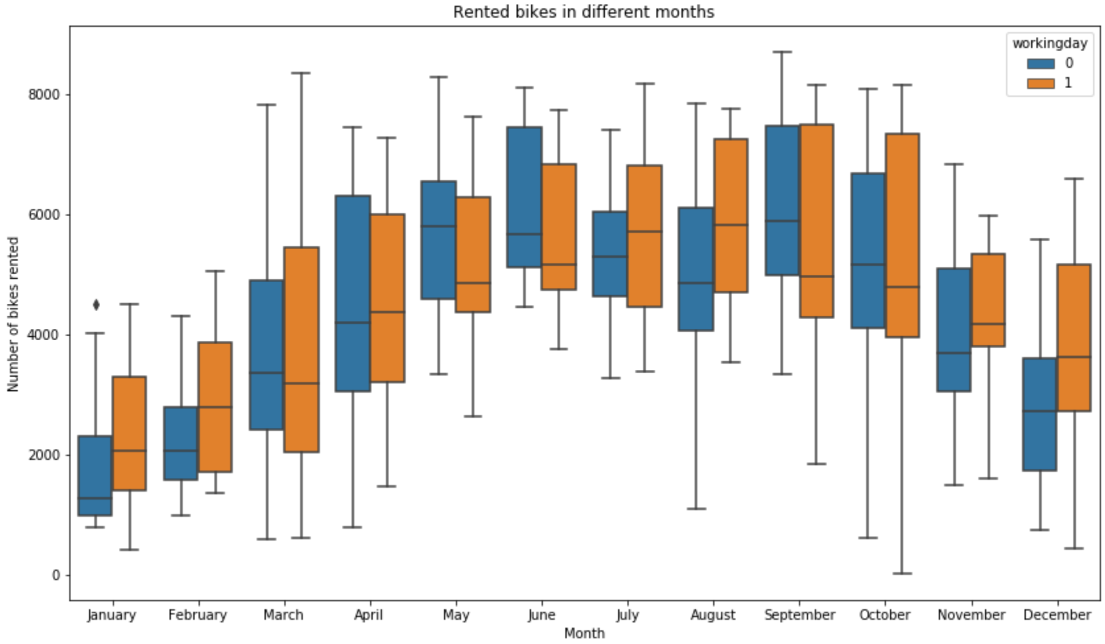
Bike sharing utilization over the two years The objective of this Case is to Predication of bike rental count on daily based on the environmental and seasonal settings.



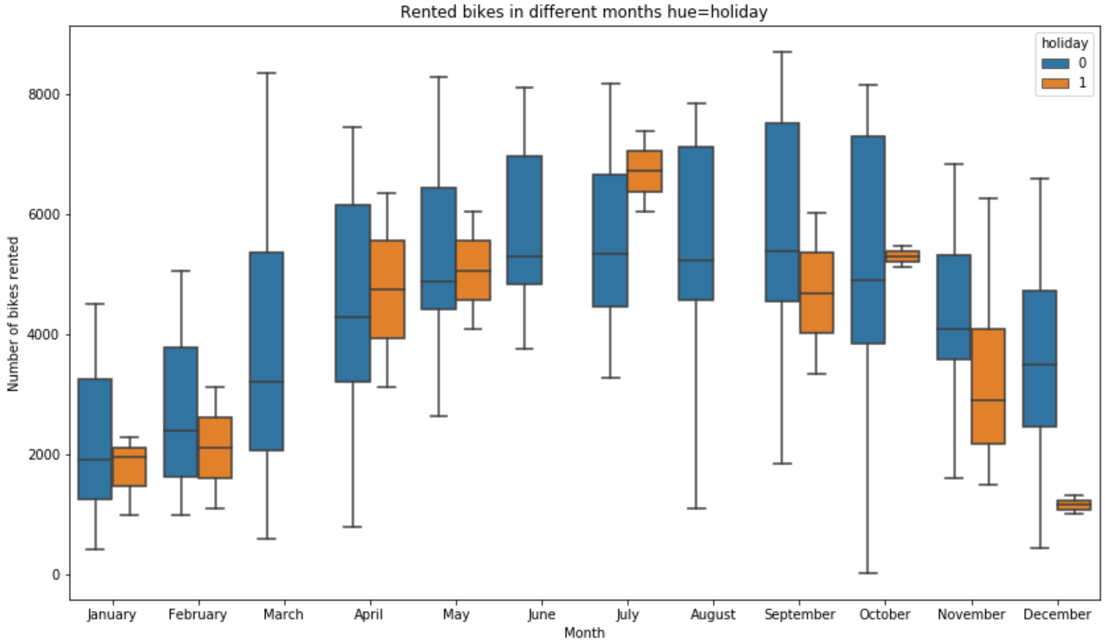
From the above figure, it seems that the number of total rented bikes follow a nearly normal distribution. The first plot shows the relationship between cnt variable and date. We can see that the overall trend increased during the two-year time span. And within each year, there are huge amount of bike rentals during summer and fall seasons.
Next, we looked at the relationship between the response variable and each explanatory variable. We selected some plots with obvious patterns as shown below.



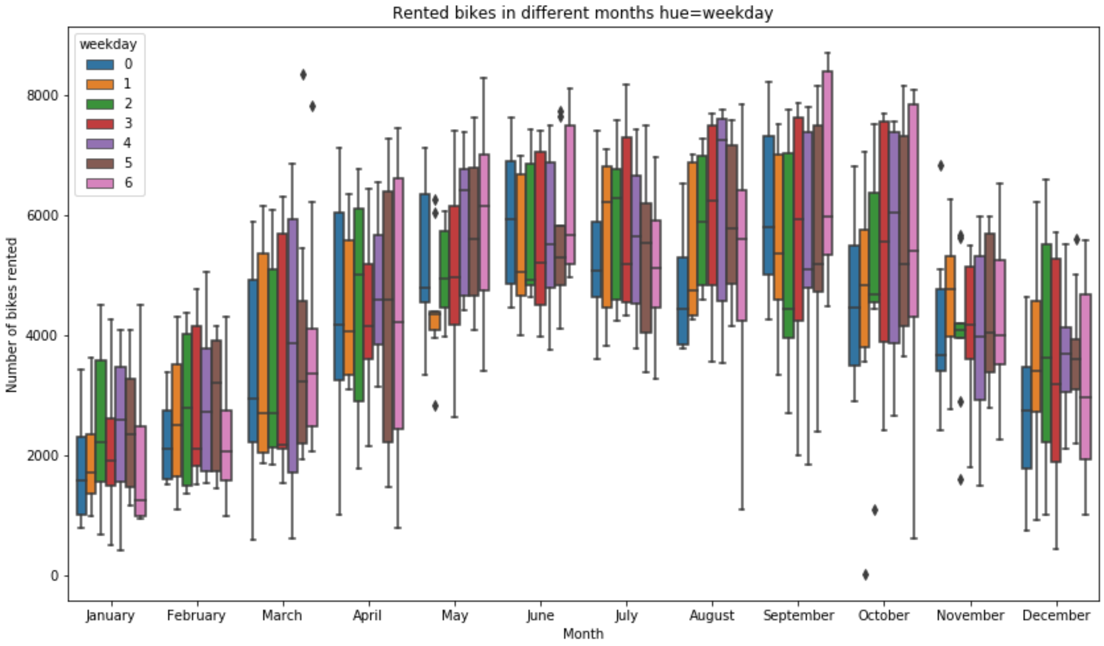
The second plot shows the relationship between cnt variable and season, which confirms the conclusion we got above. The average numbers of bike rentals are the highest during summer and fall.



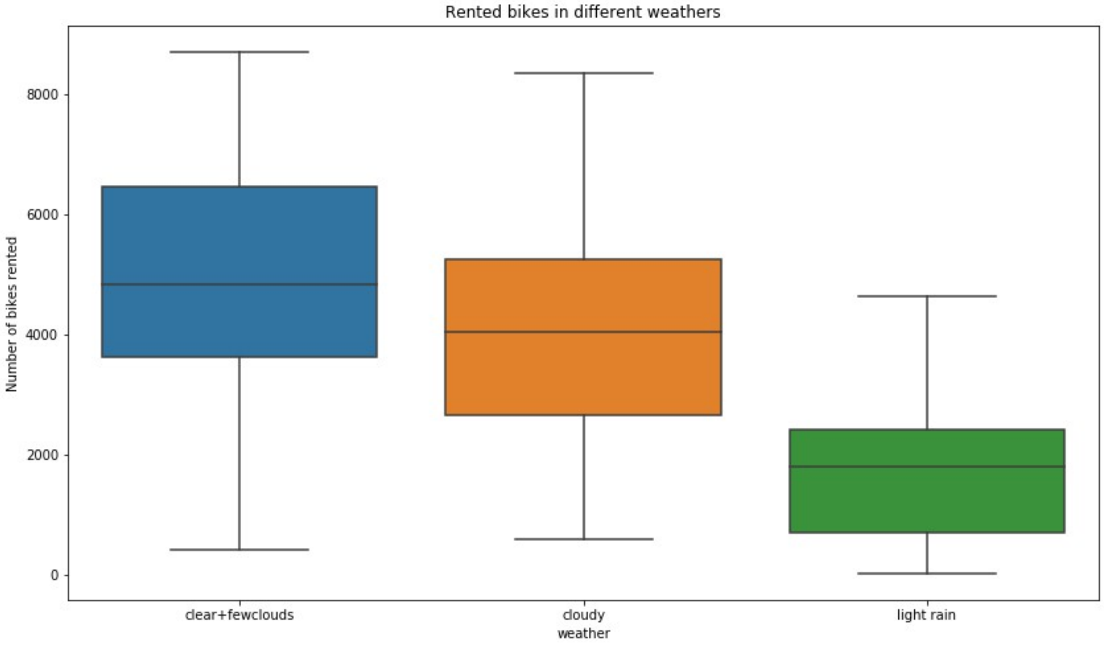
The third plot shows the relationship between cnt variable and month with hue as working day. We can see that the average number of bike rentals on working-day is higher than non-working-day, bu has more variability as well



The fourth plot shows the relationship between cnt variable and month with hue as holiday. We can see that the average number of bike rentals on non-holiday is higher than holiday.



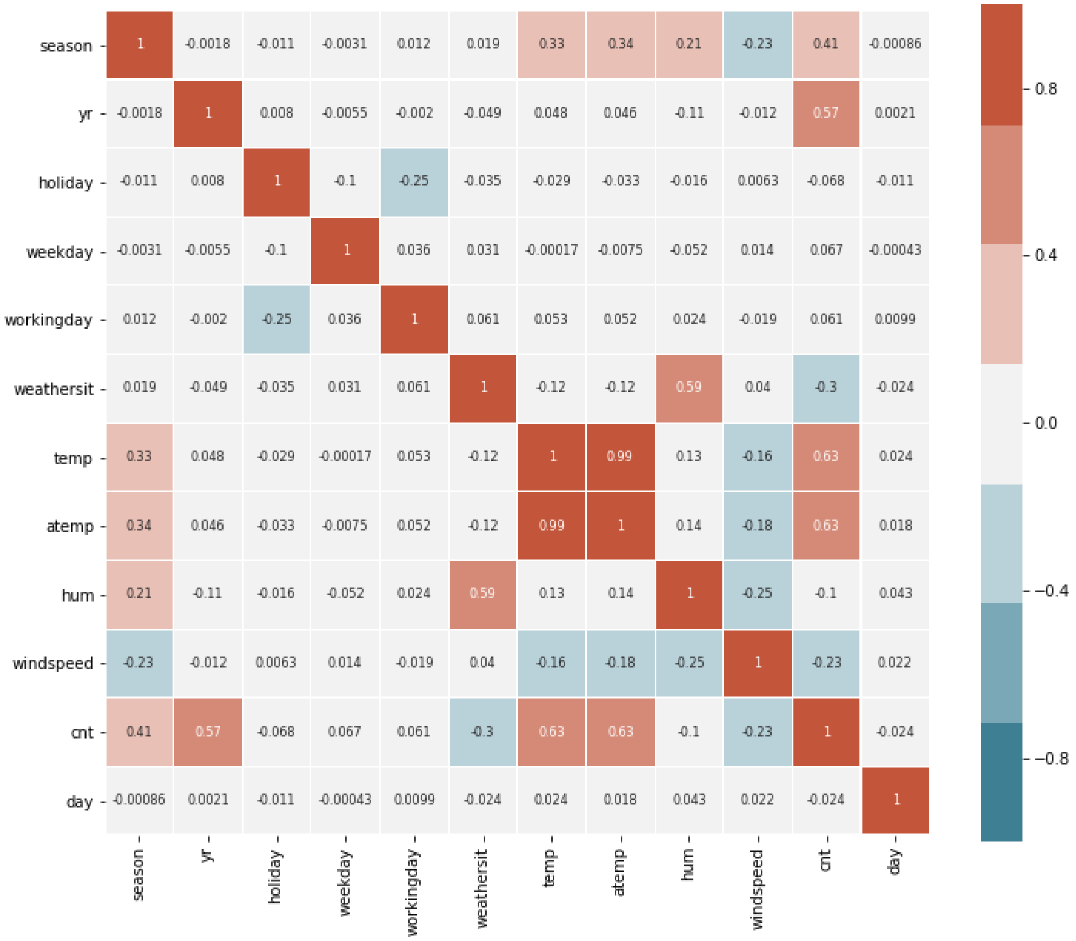
The fifth plot shows the relationship between cnt variable and month with hue as weekday. We can see that the average number of bike rentals on non-holiday is higher than holiday.



The sixth plot shows the relationship between cnt variable and weather. There is a clearly decreasing trend of bike rentals when weather conditions grow worse

2.2 Correlation Analysis:

A correlation analysis will allow to identify relationships between the dataset variables. A plot of their distributions highlighting the value of the target variable might also reveal some patterns.

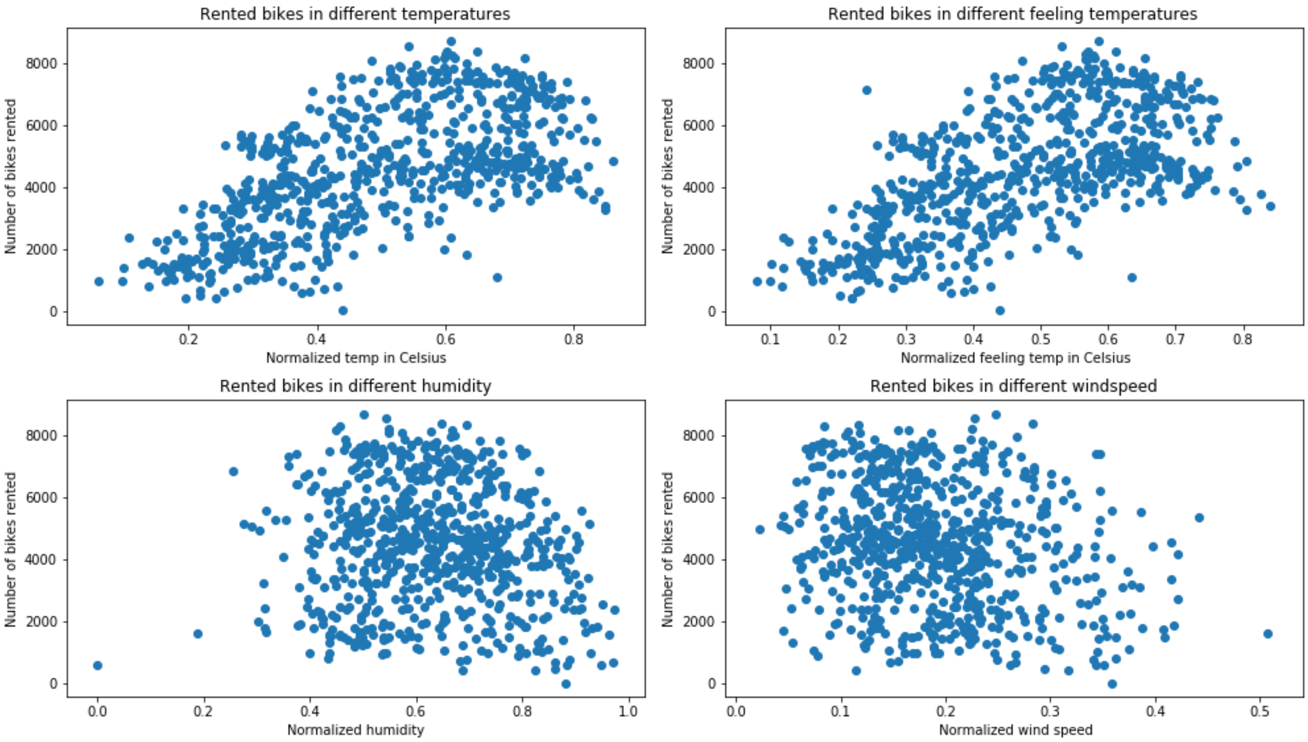


	season	yr	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	cnt	day
season	1	-0.0018	-0.011	-0.0031	0.012	0.019	0.33	0.34	0.21	-0.23	0.41	-0.00086
yr	-0.0018	1	0.008	-0.0055	-0.002	-0.049	0.048	0.046	-0.11	-0.012	0.57	0.0021
holiday	-0.011	0.008	1	-0.1	-0.25	-0.035	-0.029	-0.033	-0.016	0.0063	-0.068	-0.011
weekday	-0.0031	-0.0055	-0.1	1	0.036	0.031	-0.00017	-0.0075	-0.052	0.014	0.067	-0.00043
workingday	0.012	-0.002	-0.25	0.036	1	0.061	0.053	0.052	0.024	-0.019	0.061	0.0099
weathersit	0.019	-0.049	-0.035	0.031	0.061	1	-0.12	-0.12	0.59	0.04	-0.3	-0.024
temp	0.33	0.048	-0.029	-0.00017	0.053	-0.12	1	0.99	0.13	-0.16	0.63	0.024
atemp	0.34	0.046	-0.033	-0.0075	0.052	-0.12	0.99	1	0.14	-0.18	0.63	0.018
hum	0.21	-0.11	-0.016	-0.052	0.024	0.59	0.13	0.14	1	-0.25	-0.1	0.043
windspeed	-0.23	-0.012	0.0063	0.014	-0.019	0.04	-0.16	-0.18	-0.25	1	-0.23	0.022
cnt	0.41	0.57	-0.068	0.067	0.061	-0.3	0.63	0.63	-0.1	-0.23	1	-0.024
day	-0.00086	0.0021	-0.011	-0.00043	0.0099	-0.024	0.024	0.018	0.043	0.022	-0.024	1

From the above correlation plots actual temperature is more correlated with bike rentals count, humidity and wind speed are also slightly correlated. Temperature and actual are highly correlated.

Another problem are our registered and casual variables. Think about it: imagine that you are trying to predict the total number of users for tomorrow. While you can have data such as the month, the weekday, the temperature and the weather condition, it's impossible to have the number of registered and casual users because this is exactly what you are trying to predict.

Also, since the count is a decomposition of these two variables, we could have problems if they remain on the data set. So, let's get rid of them too.



These four plots show the relationship between cnt variable temperature,actual temperature,humidity and windspeed. There seems to be a linear relationship between bike rentals count and temperature, which means more people will rent bikes when it gets warmer. However, the data seems to be scattered with a lot of variability, so the linear relationship might be weak if there is any.

Chapter 3 : Encoding Categorical Data

You will now learn different techniques to encode the categorical features to numeric quantities. To keep it simple, you will apply these encoding methods only on the column. However, the same approach can be extended to all columns.

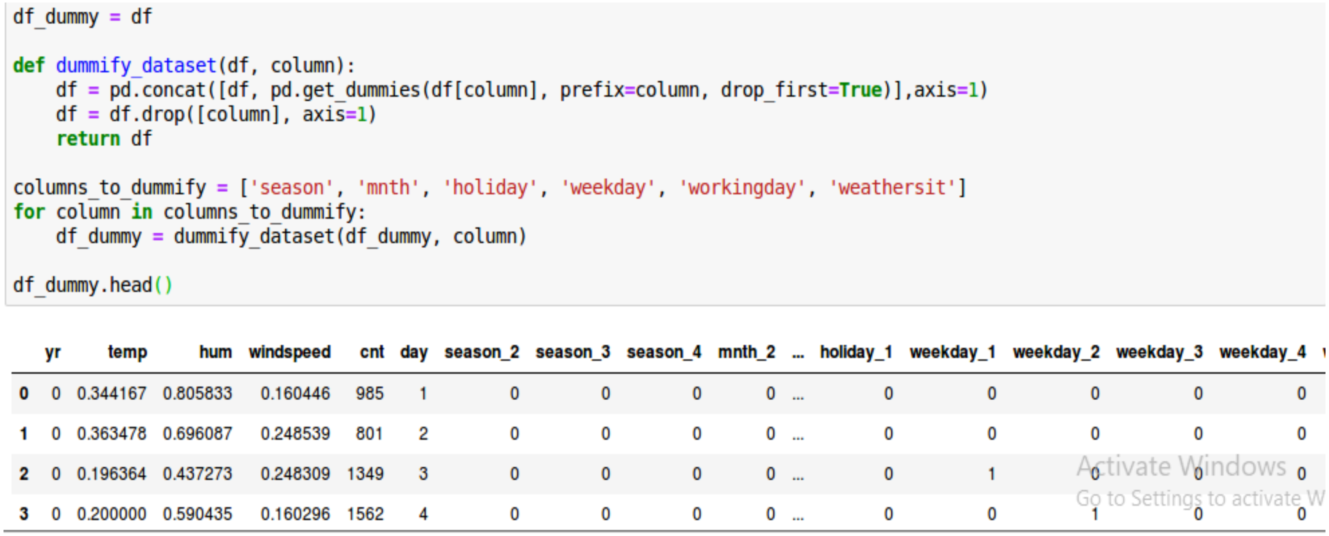
The techniques that you'll cover are the following:

- Replacing values
- Encoding labels
- One-Hot encoding
- Binary encoding
- Backward difference encoding
- Miscellaneous features

3.1 One-Hot encoding

Since we have categorical values in our data set, we need to ‘tell’ our algorithm that classes have equal weight for our analysis. For instance: our weekdays are represented by numbers from 0 to 6. But we can’t really say that a 6 is better than a 5 here.

A way to change this perspective is using the one hot encoding technique. This is a process by which we convert categorical variables into binary categories. By the way, when we apply one hot encoding, it’s important to left one variable out to avoid multicollinearity.



```
df_dummy = df

def dummify_dataset(df, column):
    df = pd.concat([df, pd.get_dummies(df[column], prefix=column, drop_first=True)],axis=1)
    df = df.drop([column], axis=1)
    return df

columns_to_dummify = ['season', 'mnth', 'holiday', 'weekday', 'workingday', 'weathersit']
for column in columns_to_dummify:
    df_dummy = dummify_dataset(df_dummy, column)

df_dummy.head()
```

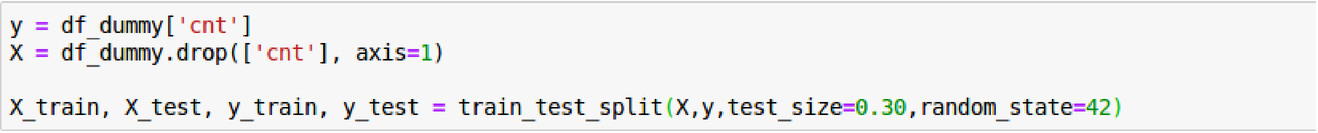
	yr	temp	hum	windspeed	cnt	day	season_2	season_3	season_4	mnth_2	...	holiday_1	weekday_1	weekday_2	weekday_3	weekday_4
0	0	0.344167	0.805833	0.160446	985	1	0	0	0	0	...	0	0	0	0	0
1	0	0.363478	0.696087	0.248539	801	2	0	0	0	0	...	0	0	0	0	0
2	0	0.196364	0.437273	0.248309	1349	3	0	0	0	0	...	0	1	0	0	0
3	0	0.200000	0.590435	0.160296	1562	4	0	0	0	0	...	0	0	1	0	0

Chapter 4: Choosing the algorithm

When the data is not normally distributed, we can apply a non-linear transformation to try to fix this issue. We can also use a non parametric algorithm on these cases. By the way, we call non parametric the algorithms that do not make strong assumptions about the form of the mapping function.

We can quickly test algorithms performance by using the model_selection.cross_val_score function. Let’s see how some algorithms perform by measuring their mean squared error.

But before, we need to split our data into training and test. Here we can use train_test_split to split the data by importing sklearn.model_selection module.



```
y = df_dummy['cnt']
X = df_dummy.drop(['cnt'], axis=1)

X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.30,random_state=42)
```

4.1 Model Selection

DecisionTreeRegressor algorithm is good, because tree methods are generally insensitive to outliers. We'll do a performance test of both the models with the same number of variables. Let's check how the model behaves by checking its Root Mean Squared Logarithm Error . We see that we have an RMSLE error of DecisionTreeRegressor is 0.4343.

4.2 Conclusion :

- As we found the correlation plots against bike rentals with humidity and windspeed were slightly related, we created a DecisionTreeRegressor and found that the Root Mean Squared Logarithm Error is 0.4343 which is better performer.
- MSE = 812783.8227272728
- RMSE = 901.5452416419671
- Variance score: 0.80
- RMSLE: 0.4343134750547236
- Though, checking the residual plot , we can see that the residuals have a pattern, and are not normally distributed, which means the model doesn't fit the data so well.
- Bike ridership is maximun between summer and spring quater.
- Number of registered riders are higher in comparision to casual riders.
- For the time series analysis, only time series is not enough for prediction. Other features should also be considered.
- The corr-plot gave us the idea about which features are highly correlated to the response variable (cnt).
- DecisionTreesRegressor model is descent enough for prediction as well, although there are lot of outliers in prediction.
- The Random Forest Model with the features from corr-plot increased the accuracy of the prediction but still the spikes and drops are not that well represented by the model.