**Bike Renting**

**By: - Kushagra Shroff**

**24-03-2020**

* **INTRODUCTION**

A bike renting system is a service or a framework in which users can rent a bike and use bicycles available for shared use on a short term basis for a price or free depends on the system. Currently, there are over 500 bike-sharing programs around the world and growing with a fast pace all over the globe as a measure for a sustainable environment. Such systems usually aim to reduce traffic jams, noise pollution, and air pollution by providing free/affordable access to bicycles for short-distance trips in an urban area as opposed to motorized vehicles. Our goal is to predict the bike rental counts based on environmental and seasonal data.

Our dataset is a time-based so we will do this as a time series project. A time series is a series of data points indexed in time order. Most commonly, a time series is a sequence taken at successive equally spaced points in time.

* **PROBLEM STATEMENT**

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

* **DATA**

The details of data attributes in the dataset are as follows – • instant: Record indexdteday: Dateseason:

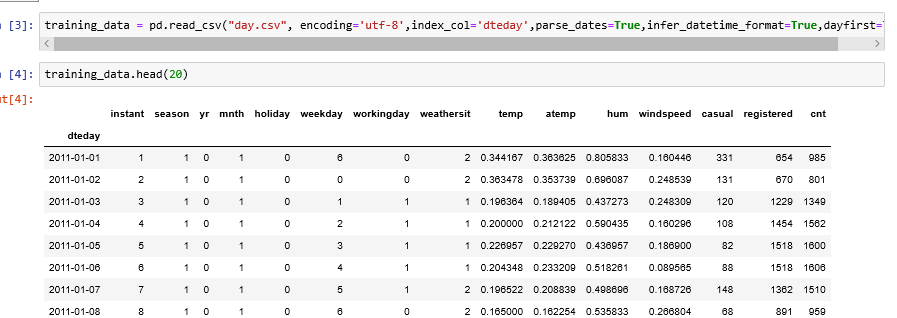
* Season (1:springer, 2:summer, 3:fall, 4:winter)
* yr: Year (0: 2011, 1:2012)
* mnth: Month (1 to 12)hr: Hour (0 to 23)
* holiday: weather day is holiday or not (extracted fromHoliday Schedule)
* weekday: Day of the week
* workingday: If day is neither weekend nor holiday is 1, otherwise is 0.
* weathersit: (extracted fromFreemeteo) o 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 +

Scatteredclouds o 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog

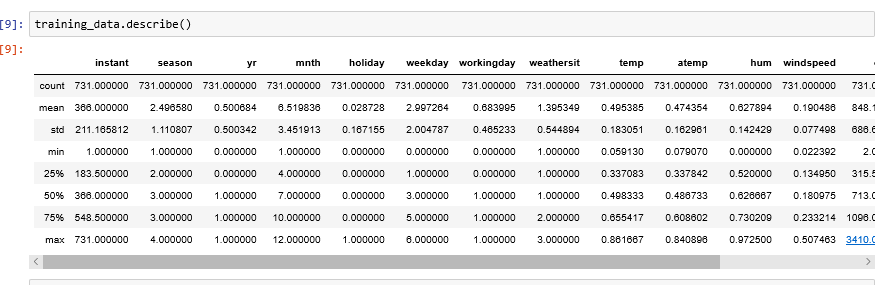
* temp: Normalized temperature in Celsius. The values are derived via(t-t\_min)/(t\_maxt\_min),t\_min=-8, t\_max=+39 (only in hourly scale)
* atemp: Normalized feeling temperature in Celsius. The values are derived via(tt\_min)/(t\_max-t\_min), t\_min=-16, t\_max=+50 (only in hourly scale)
* hum: Normalized humidity. The values are divided to 100 (max)
* windspeed: Normalized wind speed. The values are divided to 67 (max)
* casual: count of casual users
* registered: count of registered users
* cnt: count of total rental bikes including both casual and registered

***DESIGN AND IMPLEMENTATION***

1. **Loading the data**

****

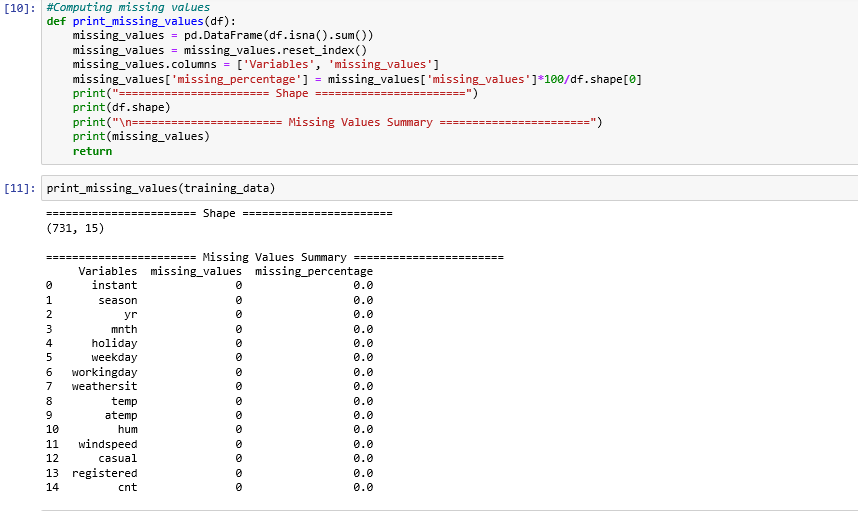
While loading we are parsing dates into date-time object and making date as an index column.



1. **Data Pre-Processing**
2. ***Missing Value Analysis***

Missing information can happen in view of nonresponse, no data is accommodated at least one things or for an entire unit. Once in a while missing qualities are brought about by the analyst. For instance, when the information assortment is done inappropriately or mix-ups are made in information passage, it is called Human Error.

Not all far observed data is an outlier but they can be good for analysis of an unseen data. Thus using common sense and basic knowledge we can interpret this far data entries.



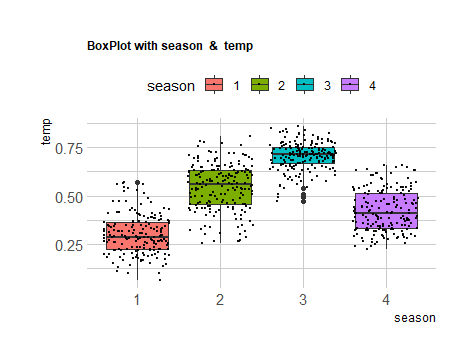
1. ***Outlier Analysis***

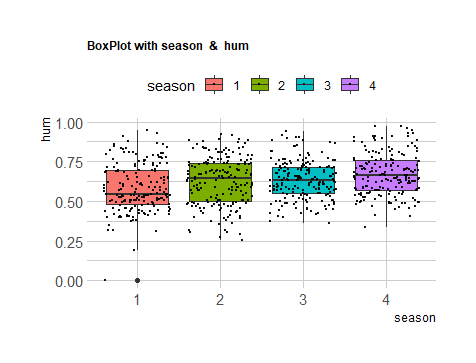
An Outlier is an item that veers off essentially from the remainder of the articles. They can be brought about by estimation or execution mistake. The investigation of exception information is alluded to as anomaly examination or anomaly mining.

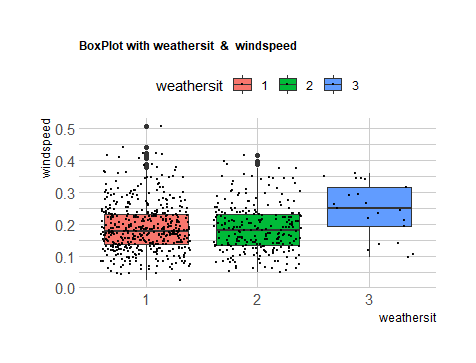
Not all far observed data is an outlier but they can be good for analysis of an unseen data. Thus using common sense and basic knowledge we can interpret this far data entries.

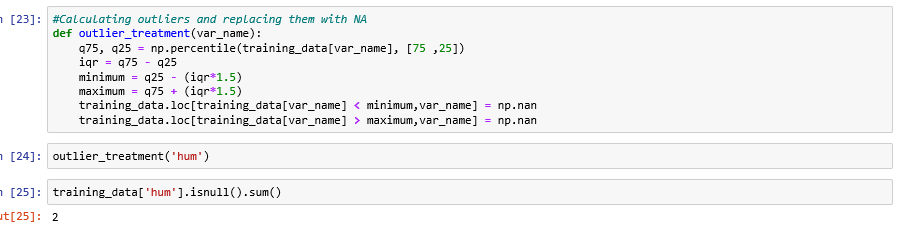
A basic method to approach to distinguish exception is to utilize boxplot.

On the off chance that there is a lot of information in anomaly, at that point erasing them isn't a productive way, since we will lose parcel of information.

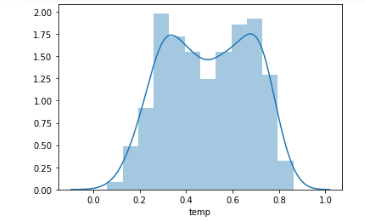


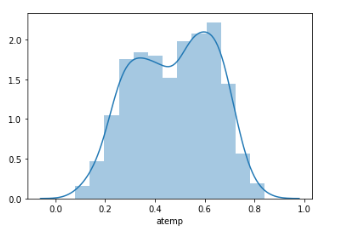


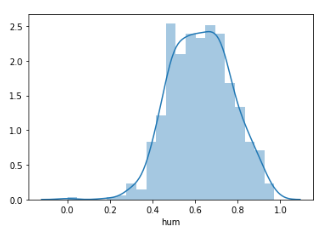


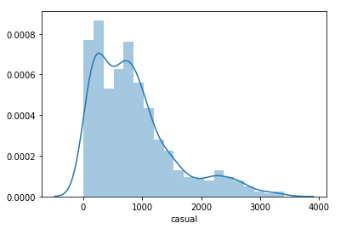


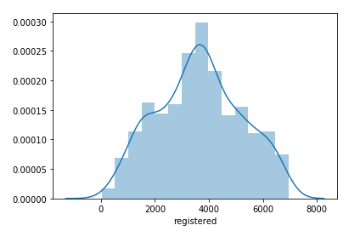
* ***Plotting distplots for various features.***
* ***A distplot plots a univariate distribution of observations.***

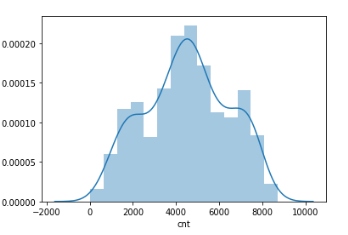






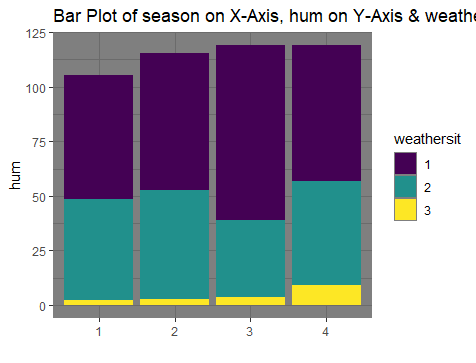


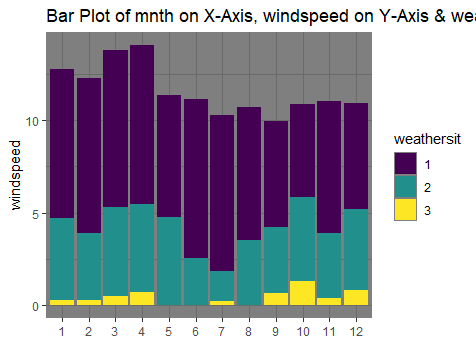


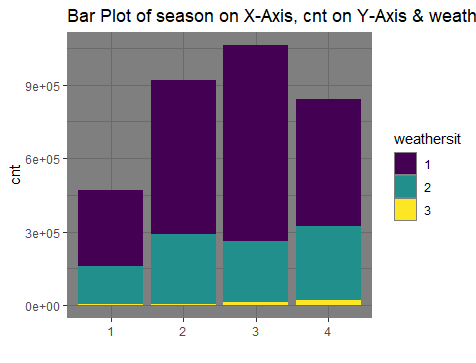


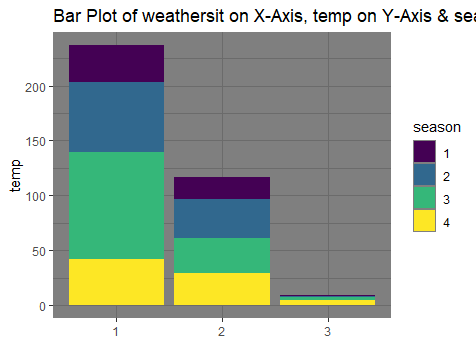
***Plotting stacked bar graph.***

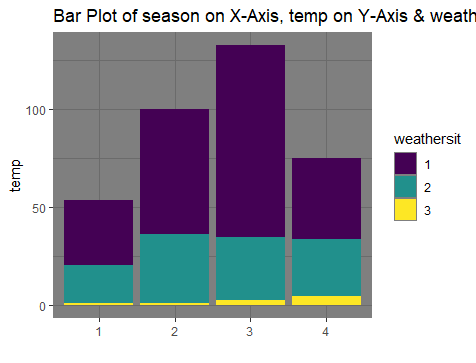
***A stacked bar graph is a chart that uses bar to show comparisons between categories of data, but with ability to break down and compare parts of a whole.***

******

******

******

******

******

***#Observations***

***'''***

***1.> There are few outliers in hum and wind speed which I’ll take into consideration later.***

***2.> Data is not noisy though.***

***3.> No outliers in temp and atemp***

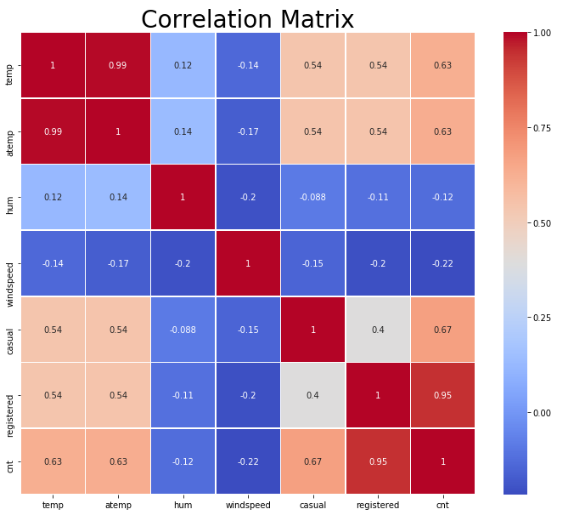
***4.> Little skewness***

***'''***

1. ***Feature Selection***

For modelling, we have to select the columns independent variables that will highly contribute to the dependent variable.

If there is no dependent variable, we have to find the collinearity between all the variables and if there is high collinearity between two variables, we can remove any one variable and keep the other.



***# Observations***

***# 'temp' and 'atemp' are very highly correlated with each other.***

***# Similarly, 'registered' and 'cnt' are highly correlated with each other***

***# Chi-Square test for independence of categorical variable***

***Hypothetical Test :***

***Null Hypothesis: 2 variables are independent.***

***Alternate Hypothesis: 2 variables are not independent.***

***If p-value is less than 0.05 then we reject the null hypothesis saying that 2 variables are dependent.***

***And if p-value is greater than 0.05 then we accept the null hypothesis saying that 2 variables are independent.***

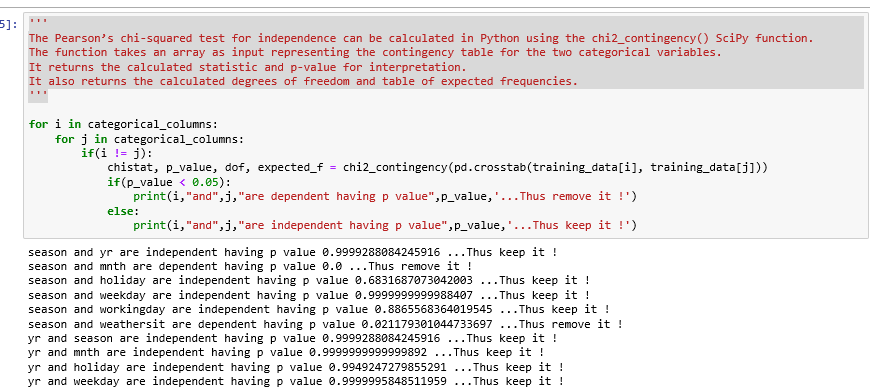
***There should be no dependencies between independent variables.***

***So we will remove that variable whose p-value with other variable is low than 0.05.***

***And we will keep that variable whose p-value with other variable is high than 0.05***

***'''***

* The Pearson’s chi-squared test for independence can be calculated in Python using the chi2\_contingency() SciPy function.
* The function takes an array as input representing the contingency table for the two categorical variables.
* It returns the calculated statistic and p-value for interpretation.
* It also returns the calculated degrees of freedom and table of expected frequencies.



* holiday and weekday-8.56e-11
* hoilday and workingday-4.033e-11
* workingday and holiday-4.033e-11
* workingday and weekday-6.77e-11
* weathersit and season-0.0211
* weekday and holiday-8.56e-11
* weekday and workingday-6.77e-136
* season and weathersit-0.0211
* season and month-0
* weathersit and mnth-0.014
* mnth and season-0
* mnth and weathersit-0.014
* So besides season,holiday and yr we will remove weekday,weathersit,workingday,mnth.

A variance inflation factor(VIF) detects multicollinearity in regression analysis.

Multicollinearity is when there’s correlation between predictors (i.e. independent variables) in a model.

It’s presence can adversely affect your regression results.

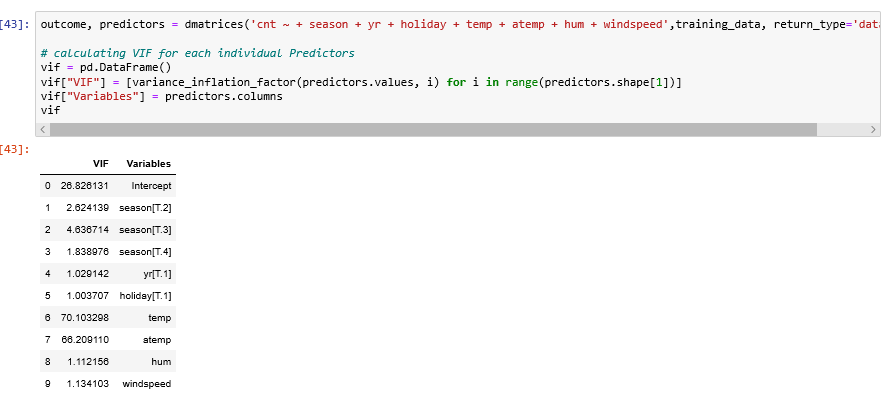
The VIF estimates how much the variance of a regression coefficient is inflated due to multicollinearity in the model.

A rule of thumb for interpreting the variance inflation factor:

1 = not correlated.

Between 1 and 5 = moderately correlated.

Greater than 5 = highly correlated.



1. ***Feature Scaling***

Feature Scaling is a step of Data Pre Processing which is applied to independent variables or features of data. It basically helps to normalise the data within a particular range. Sometimes, it also helps in speeding up the calculations in an algorithm.

* Rescaling (min-max normalization)
* Mean Normalization
* Standardization (Z-score Normalization)
* Scaling to Unit etc.

Standardization has to be applied when the data is normally/uniformly distributed that is graph is having a bell shaped curve.

In our case we are normalizing 4 features.

num\_v=['temp','atemp','hum','windspeed']

#Normalisation

for i in num\_v:

print(i)

training\_data[i] = ( training\_data[i] - min( training\_data[i]))/(max( training\_data[i]) - min( training\_data[i]))

1. ***Feature Extraction***

* From above Dataframe we see that there is Multicollinearity in our Data.
* temp and atemp has highest VIF value.
* Instead of dropping one variable i'm combining those two as mean value as a part of feature engineering.

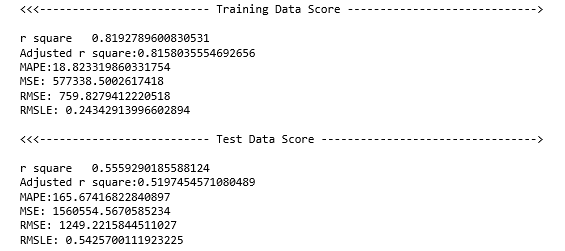


1. **Model Creation**

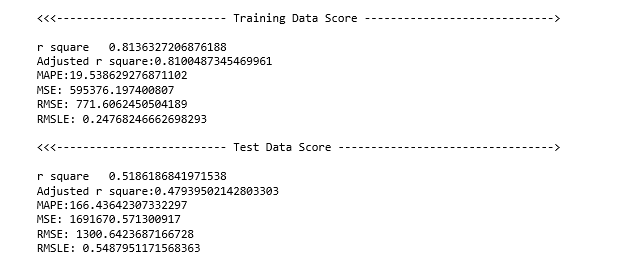
Some of the time the information or the business goals loan themselves to a particular calculation or model. Different occasions the best methodology isn't so obvious. As you investigate the information, run the same number of calculations as you can; think about their yields. Base your decision of the last model on the general outcomes. Here and there you're in an ideal situation running a troupe of models all the while on the information and picking a last model by looking at their yields.

**Results** (referred from python code file name pypro2)

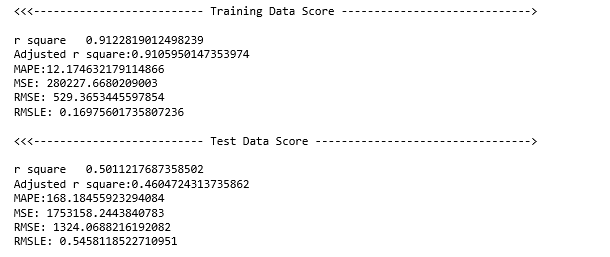
1. **Linear Regression**



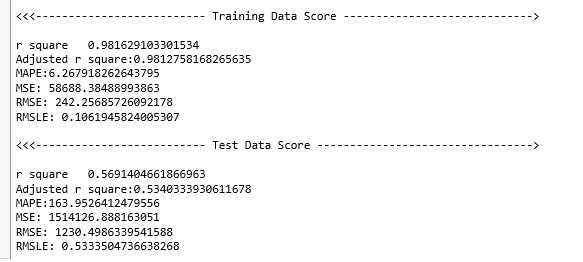
1. **Ridge Regression**



1. **Decision Tree**

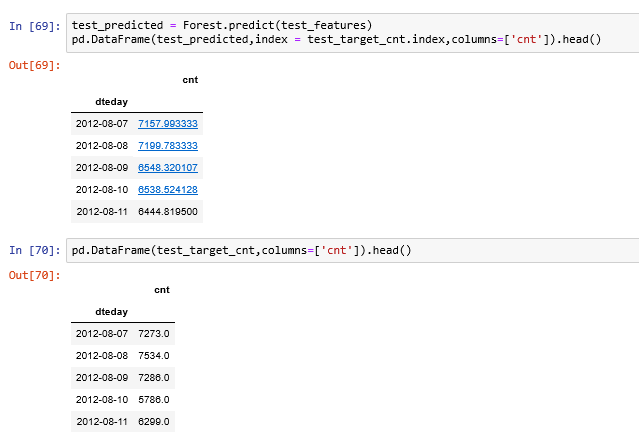
****

1. **Random Forest**



**Conclusion**

**We have selected Random Forest as our best model.**

****

**Our distribution is almost identical.**

****

**###########################################################################**

**THANKS.**