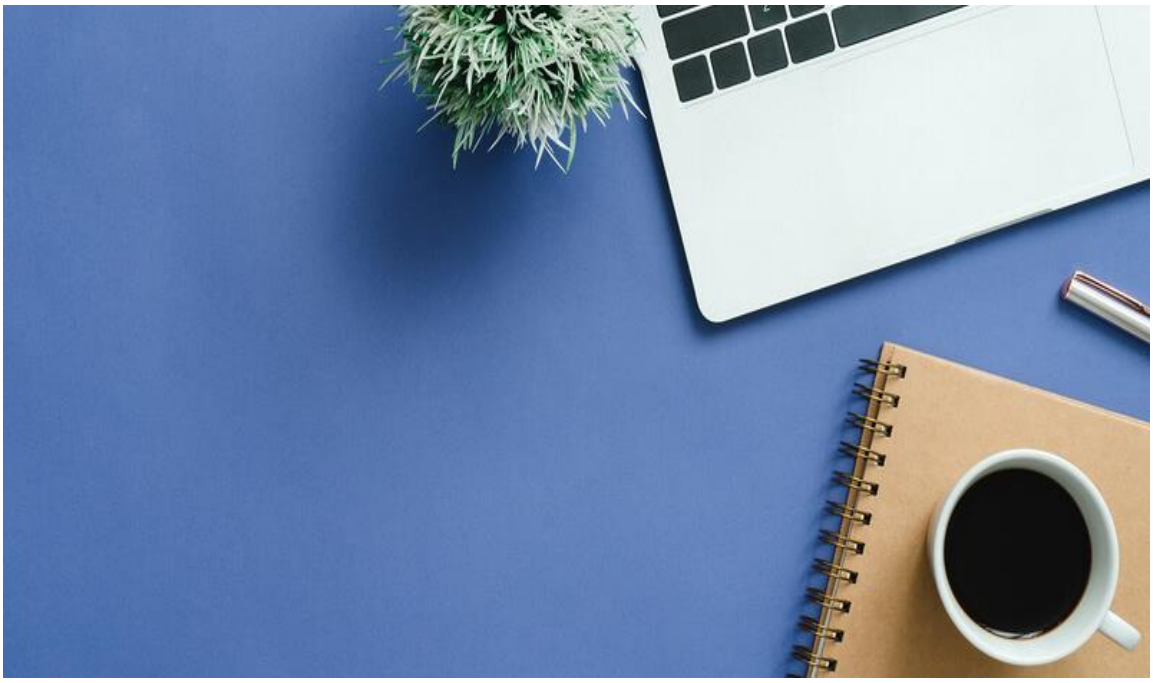
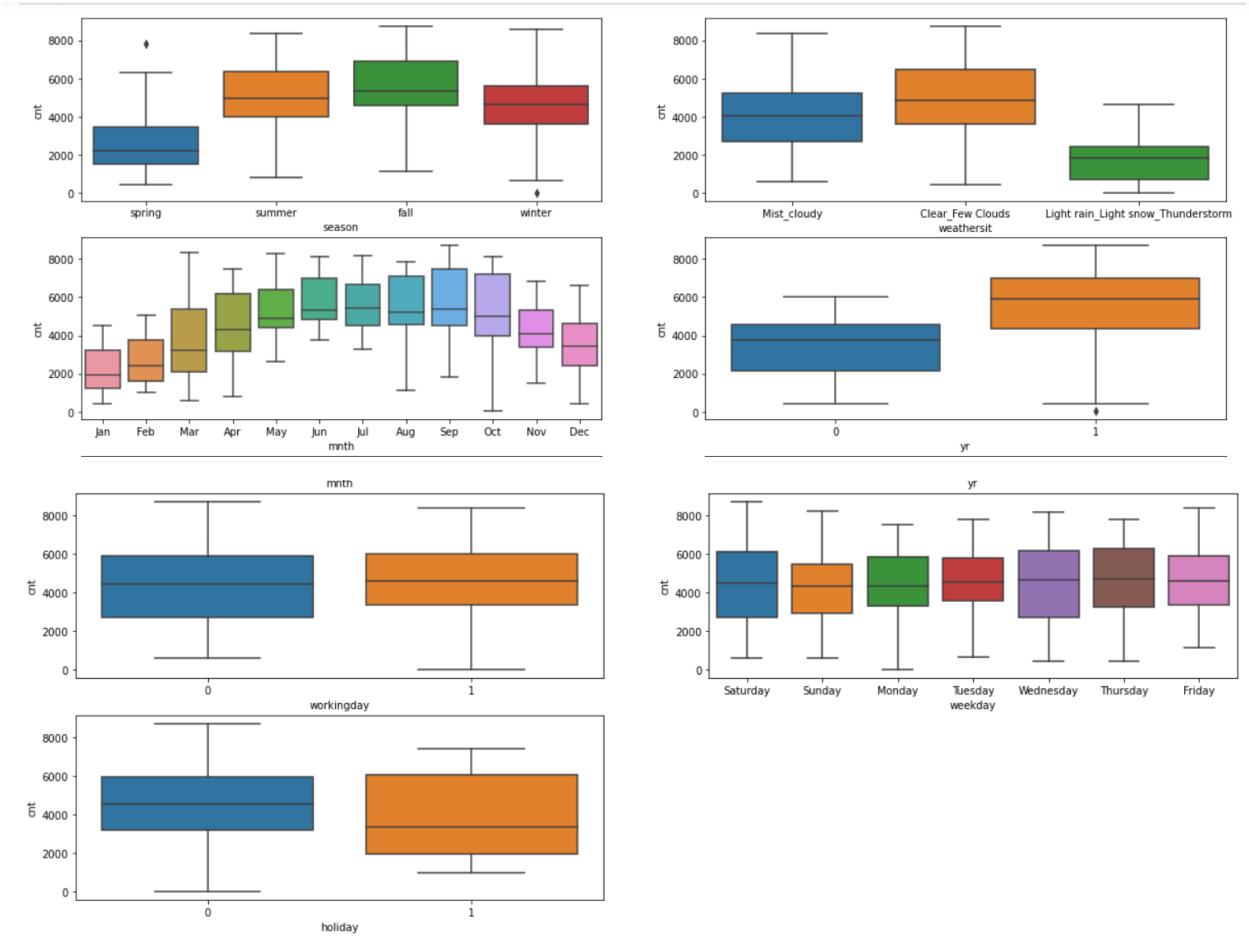
TABLE OF CONTENTS



ASSIGNMENT-BASED SUBJECTIVE

1. **From your analysis of the categorical variables from the dataset, what could you infer about their effect on the dependent variable?**

Ans. To analyze effect of these categorical variables on the dependent variable (cnt in this case), we plotted the sub plots of box plot. Each box plot shows the spread of each category with respect to cnt. There are following categorical variables in our dataset: Season, weathersit, mnth, yr, workingday, weekday, holiday. Refer below plots.



Inferences:

- Season: Shared bikes usage is maximum in fall season and minimum in spring. It seems summer and fall seasons are favoring the usage of bikes.
- Mnth: Similar pattern as with season, is observed when we see the cnt with respect to months. We have assumed that mnth 1, 2, 3 so on corresponds to the month Jan, Feb, and March. September month has the maximum count of shared bike usage.
- Weathersit: Clear weather with few clouds, is favorable weather for using bikes. It is also visible in the plot as there is maximum usage of bikes during this weather situation.
- Yr: Bike usage has increased in the year 2019 as compared to 2018.

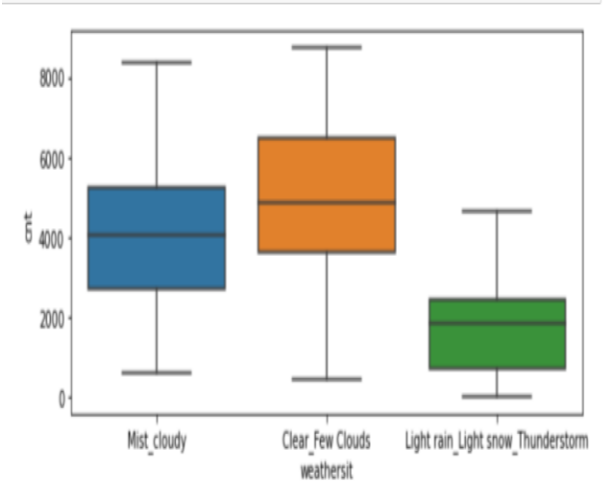
- Workingday: There is no impact on bike usage due to working or non-working days.
- Weekday: Bike usage seems unimpacted on all weekdays, as the median value is almost similar for all weekdays.
- Holiday: If it is a holiday, fewer users are using shared bikes.

2. **Why is it important to use drop_first=True during dummy variable creation?**

Ans. Pandas pd.get_dummies() converts the categorical column into indicator columns i.e. columns of 0s and 1s. Its parameter drop_first by default has the value False. It is important to use drop_first=True for dropping the first categorical variable. It is possible because if every other dummy column is 0, then this means your first value would have been 1.

E.g. If we have a categorical column that has 5 unique categorical values, using pd.getdummies() we first convert these into a binary vector of 5 columns. For a row, the True value indicated in a column is represented by 1, and in other columns, False is represented by 0. So, if we specify drop_first =true, then a binary vector of 4 columns is created. And the value 0 for all these 4 columns in a row, will represent that the 5th label is True.

In the bike-sharing case study, the **weathersit** categorical column had 3 distinct values. When we created dummy variables for this, specifying drop_first= True, we end up having 2 new columns, instead of 3.



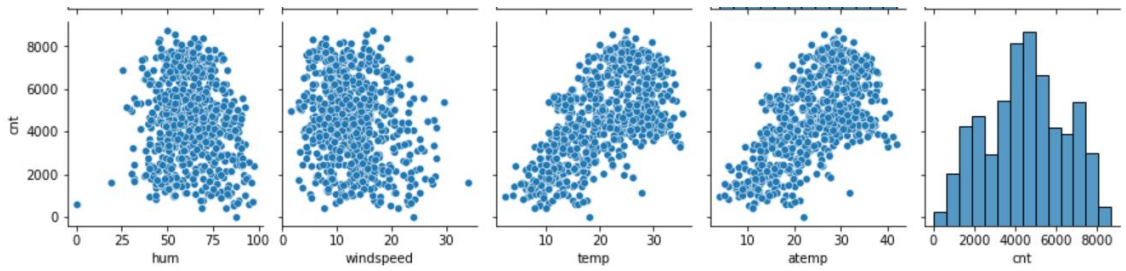

```
1 Weather_condition
```

	Light rain_Light snow_Thunderstorm	Mist_cloudy
0	0	1
1	0	1
2	0	0
3	0	0
4	0	0
...	...	...
725	0	1
726	0	1
727	0	1
728	0	0
729	0	1

730 rows × 2 columns

3. **Looking at the pair-plot among the numerical variables, which one has the highest correlation with the target variable?**

Ans. Looking at the below pair plot, we can say that temp and atemp have the highest correlation with the target variable 'cnt'. Later due to multicollinearity, we dropped column temp, hence we can conclude that the numeric variable atemp has the highest correlation with the target variable.

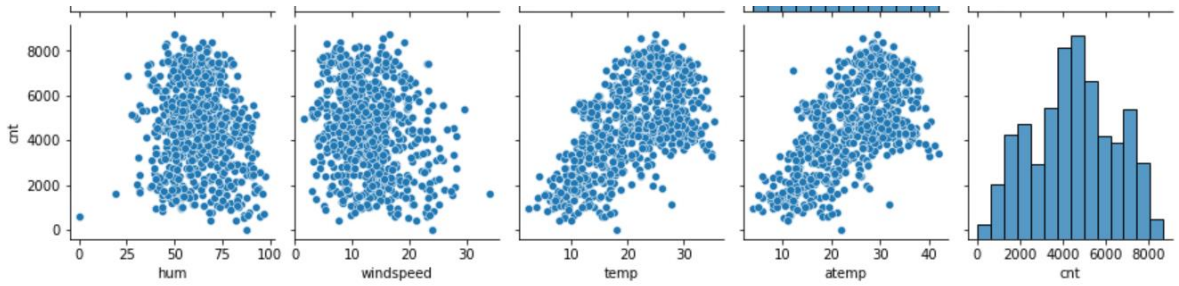


4. **How did you validate the assumptions of Linear Regression after building the model on the training set?**

Ans. We validated the assumptions of Linear Regression as follow:

- **Linear Regression Assumption 1 — Linear relationship**

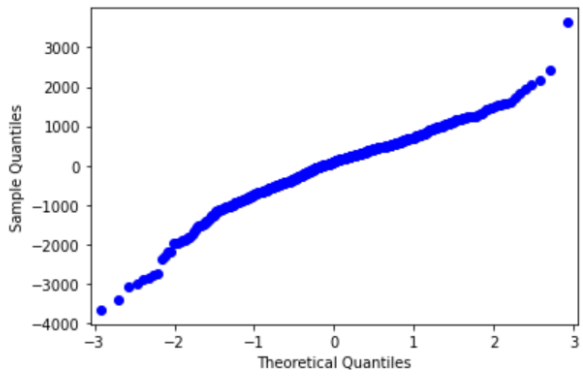
  By plotting pair plot between each numeric feature and target variable.



- **Linear Regression Assumption 2 — Normality of the residuals**

By plotting Q-Q plot

```
# QQ Plots - let's see QQ Plots of the residuals
sm.qqplot(residuals)
plt.show()
```



- **Linear Regression Assumption 3 — No or little Multicollinearity**

While removing or adding features during perfect Model Building, we checked at every step RIF value of all selected features. We insured the following thing for every independent variable:

VIF stays less than 5 and, R^2 >=0.8.

```
#Check VIF Again
checkVIF(X_train_rfe)
```

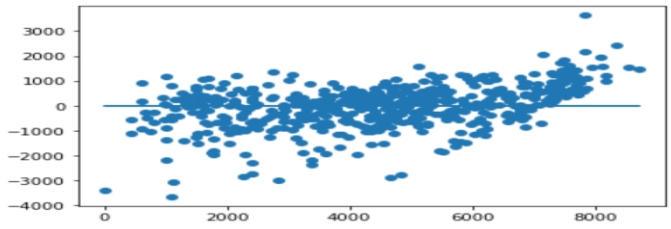
	Features	VIF
2	atemp	2.30
0	yr	1.99
5	Mist_cloudy	1.44
3	spring	1.20
6	Sep	1.15
1	holiday	1.03
4	Light rain_Light snow_Thunderstorm	1.03

- **Linear Regression Assumption 4 — Homoscedasticity**

We validated this by plotting a scatter plot between Y_train data and residuals.

```
plt.scatter(y_train, residuals)
plt.plot(y_train, [0]*len(y_train))
```

```
[<matplotlib.lines.Line2D at 0x1c68b795520>]
```



5. **Based on the final model, which are the top 3 features contributing significantly towards explaining the demand for the shared bikes?**

Ans. The linear equation of our linear model is:

cnt=2329.80+2034.9703$*yr$−873.5290$*holiday$+3421.5839$*atemp$−1424.4364$*spring$−2561.0004$*LightrainLightsnowThunderstorm$−649.5445$*MistCloudy$+612.9308$*Sep$

The top features that are contributing significantly towards explaining the demand for shared bikes are as below:

S.No.	Feature Name	Coefficient Value
1.	atemp (feeling temperature in Celsius)	3421.5839
2.	Year	2034.9703
3.	$LightrainLightsnowThunderstorm$	-2561.0004

- The demand for shared bikes increasing with increase in temperature and Year.
- The demand of shared bikes drops when there is unfavorable weather like Light, rain, snow or thunderstorm.

GENERAL SUBJECTIVE QUESTIONS

1. **Explain the linear regression algorithm in detail.**

Ans. The linear regression algorithm is based on the significance of an equation of a straight line.

$$Y=mX+ b$$

The linear regression algorithm finds out the best fit straight line describing a linear relationship between a dependent (y) and one or more independent (x) variables. This algorithm is a method to do predictive analysis, where linear relation exists among target and independent variables.

Predictor variable. (Y): The independent variable
Dependent variables(X): The output variables.

As we also have target variable or labeled data, in the available historic data set, this algorithm is categorized as **supervised Learning Algo**.

Cost Function:
The best-fit line is calculated by minimizing a quantity called Residual Sum of Squares (RSS). Various optimization algorithms can be used to minimize the RSS. These algorithms are known as Cost Functions or loss functions or error functions.

- Commonly used methods for minimizing Cost Function:
  Differentiation
  Gradient descent method

The strength of a linear regression model is explained by R^2.

$$R^2 = 1 - (RSS / TSS)$$

RSS: Residual Sum of Squares

TSS: Total Sum of Squares

Linear regression algorithms can be categorized into two categories depending upon the number of independent variables:

Simple linear regression: The number of independent variables is 1.

Multiple linear regression: The number of independent variables is more than 1. The linear Equation in this case will be as below:

$$Y = \beta0 + \beta1.X1 + \beta2.X2 + \beta3.X3 + \beta4.X4 + \beta5.X5 + \beta5.X6 + \epsilon$$

β0 = Y-intercept (always a constant)
β1, β2, β3, β4, β5 = regression coefficients
ϵ = Error terms (Residuals)

Steps to be followed in Linear Regression Algorithm:

1. Reading and understanding the data
2. Visualizing the data (Exploratory Data Analysis)
    - Visualizing numerical variables using scatter or pair plots.
    - Visualizing categorical variables using box plots or bar plots.
3. Data Preparation
4. Data Splitting into training and test sets and rescaling
5. Building a linear model:
    - Forward Selection: We start with empty model and add variables one by one.
    - Backward Selection: We add all the variables at once and then eliminate variables based on high multicollinearity or insignificance.
    - RFE or Recursive Feature Elimination: Automated Approach.
6. Residual analysis of the training data: It tells us how much the errors (y_real — y_pred) are distributed across the model.
7. Making predictions using the final model.
8. Model Evaluation

2. Explain the Anscombe's quartet in detail.

Ans. Anscombe's quartet is an intentionally created set of 4 data sets to depict the importance of data visualization by a famous statistician Francis Anscombe. It was created in the year 1973.

He illustrated the importance of plotting the graphs before analyzing and model building through these data sets. He also depicted how a regression algorithm can be fooled by similar-looking Data sets.

These four data sets have almost **identical simple descriptive statistics** but have very different distributions when graphed.

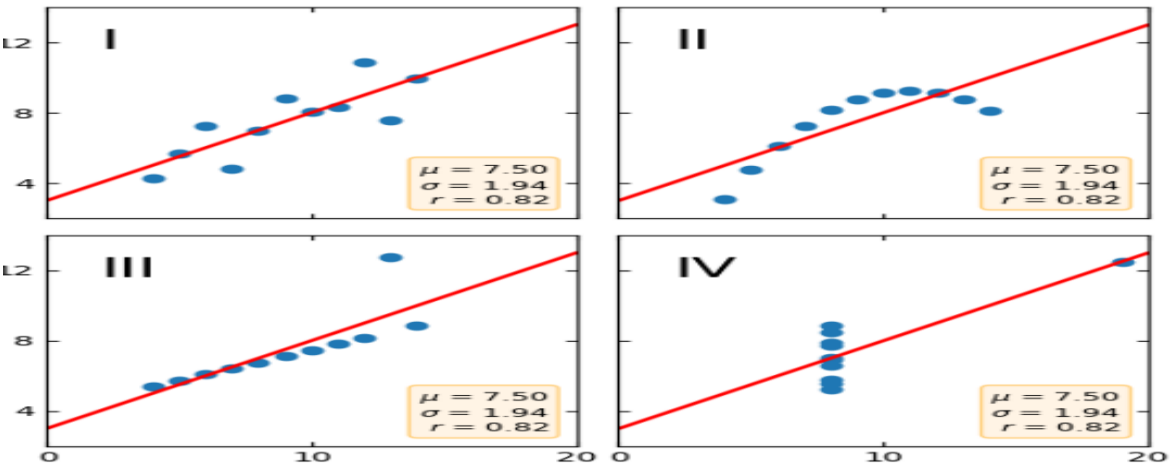
Each data set consists of 11 data points. The x values are the same for the first three datasets. The datasets are as follows.

A	B	C	D	E	F	G	H	I
	I		II		III		IV	
	10	8.04	10	9.14	10	7.46	8	6.5
	8	6.95	8	8.14	8	6.77	8	5.7
	13	7.58	13	8.74	13	12.74	8	7.7
	9	8.81	9	8.77	9	7.11	8	8.8
	11	8.33	11	9.26	11	7.81	8	8.4
	14	9.96	14	8.1	14	8.84	8	7.0
	6	7.24	6	6.13	6	6.08	8	5.2
	4	4.26	4	3.1	4	5.39	19	12.
	12	10.84	12	9.13	12	8.15	8	5.5
	7	4.82	7	7.26	7	6.42	8	7.9
	5	5.68	5	4.74	5	5.73	8	6.8

Below is a simple identical descriptive statistic for all four data sets.

Property	Value	Accuracy
Mean of x	9	exact
Sample variance of x : s_x^2	11	exact
Mean of y	7.50	to 2 decimal places
Sample variance of y : s_y^2	4.125	±0.003
Correlation between x and y	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively
Coefficient of determination of the linear regression : R^2	0.67	to 2 decimal places

Below is a graphical representation of the above datasets.



Explanation of the above graphical representations:

- In the top left of the scatter plot, you can see that there seems to be a linear relationship between x and y.
- In the top right of the scatter plot, you can see that there is a non-linear relationship between x and y.
- In the bottom left of the scatter plot, you can see a perfect linear relationship for all the data points except an outlier.
- In the bottom right of the scatter plot, is an example where one high-leverage point is sufficient to produce a high correlation coefficient.

Hence, we can conclude that the data features must be plotted to see the distribution of the samples that can help you identify the various anomalies present in the data like outliers, diversity of the data, linear separability of the data, etc.

3. **What is Pearson's R?** (3 marks)

Ans. The Pearson's r is also called, **Pearson's Correlation Coefficient, Pearson product-moment correlation coefficient (PPMCC), or bivariate compound**.

- It is a statistic tool to measures the relationship between two variables or features or sets of data.
- It can only measure direct relationships between two variables. It can not differentiate dependent or independent variables.

Mathematically, Pearson's correlation coefficient is calculated by ratio of the covariance of two variables and the product of their standard deviation.

Formula

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

r = correlation coefficient
x_i = values of the x-variable in a sample
$\bar{x}$ = mean of the values of the x-variable
y_i = values of the y-variable in a sample
$\bar{y}$ = mean of the values of the y-variable

Range of R value: -1 to 1

The value of Pearson's r would range from **–1 to 1**.
−1 means there exist a perfect negative linear relationship between variables.
0 means there exist no linear relationship between variables.
1 means there is a perfect positive linear relationship between variables.

4. **What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling?**

Ans. In Data Science, Scaling is a technique for features in the dataset, to standardize their value in a fixed range. It is done during the data pre-processing stage, to fix the issue with variables with highly varying magnitudes or values or units.

Why is scaling performed: If feature scaling is not done, then an ML algorithm ignoring the units of features, tends to weigh greater values, as higher and smaller values as the lower.

For example, If the ML algorithm is not performing the feature scaling, then it will interpret the value 500 meters to be greater than 1 km which is incorrect. In such cases, the algorithm will give incorrect predictions.

So, we use Feature Scaling to bring all values to the same magnitudes and thus, handle this issue.

Techniques to perform Feature Scaling:
There are two most used feature scaling techniques:

- **Min-Max Normalization:** This technique re-scales a feature or observation value with a distribution value between 0 and 1. This is calculated as below:

$$X_{\text{new}} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

- **Standardization:** This method re-scales a feature value so that it has distribution with 0 mean value and variance equals 1. It can be calculated as below:

$$X_{\text{new}} = \frac{X_i - X_{\text{mean}}}{\text{Standard Deviation}}$$

5. **You might have observed that sometimes the value of VIF is infinite. Why does this happen?**

Ans. Variance inflation factor (VIF) is a measure of the amount of multicollinearity in a set of multiple regression variables.

Variance Inflation Factor or VIF, gives a basic quantitative idea about how much the feature variables are correlated with each other. It is an extremely important parameter to test our linear model. The formula for calculating `VIF` is:

$$VIF_i = \frac{1}{1 - R_i^2}$$

- A large value of (VIF) on an independent variable indicates a highly collinear relationship to the other variables.
- An infinite VIF value indicates that the variable can be expressed exactly by a linear combination of other variables.
- VIF becomes infinity when R square = 1
- **Why Does this happen**: It has happened because the corresponding variable is duplicate or can be exactly expressed by the combination of other variables in your dataset. View your independent variables and eliminate terms that are duplicates.
- For example, in the bike-sharing data set data distribution among **seasons** and **months** will be similar. If we will keep both in our model, then it might result in infinite VIF. We should just keep one of these variables in the model and drop the other one. It will fix the VIF for the remaining terms and now all the VIF factors are within the threshold limits.

6. **What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression.**

Ans. Quantile-Quantile (Q-Q) plot: It is a graphical tool to plot and compare probability distributions (Normal, exponential, or Uniform) of two variables. It is called a Quantile-Quantile plot because it compares variables by plotting their quantiles against each other.

- If the distributions of variables being compared are similar, the points in the Q–Q plot will lie on the line y = x.
- If the distributions of variables being compared are linearly related, the points in the Q–Q plot will lie on a line, but the line might not be y=x.

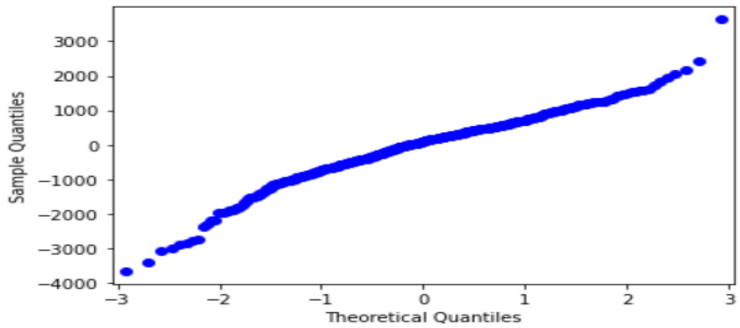
Use:

- It helps to determine if two data sets come from populations with a common distribution.
- It is used to compare the shapes of distributions.
- It is used to provide a graphical view of how properties such as location, scale, and skewness are similar or different in the two distributions.

Importance: Q-Q plot can be used to verify the assumption of Linear Regression that the residuals should follow a normal distribution.

In our bike-sharing model, we obtained the residuals from the model, then visualized the distribution of Residuals on the Q-Q plot. We plotted the theoretical quantiles or known as the standard normal variate (a normal distribution with mean=0 and standard deviation=1) on the x-axis and the residuals on the y-axis.

```
# QQ Plots - let's see QQ Plots of the residuals
sm.qqplot(residuals)
plt.show()
```



We see that the points plotted on the graph perfectly lie on a straight line so we can clearly say that this distribution is Normally distribution because it is evenly aligned with the standard normal variate which is the simple concept of the Q-Q plot.