SUBJECTIVE QUESTIONS AND ANSWERS : BOOM BIKES

Assignment-based Subjective Questions

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

Categorical variable participates in the model such that their presence effects the overall fitness of the model:

Categorical variables:

A. Season: Season are divided into Fall , summer, spring and winters. It is observed that Fall effects the "cnt" (dependent variable) the most and Spring season provides not much on the dependent variable.

B. Month : Months are futher divided as Jan, Feb, March, April……..Dec. In these all months the month will highly effects the bikes renting is August, while the month with least positive effect on the business is January.

C. Weekdays: On weekdays Sunday constitute to the minimum renting of bikes while "Friday" being the top day for bike renting. This infer that the beginning of weekend pace up the renting activity and end of the weekend reduces the renting.

D. Weathersit: Clear Weather supports bikes renting. While Heavy rain and snow will impact negatively on the renting business.

Question2 : Why is it important to use drop_first=True during dummy variable creation?

It is advisable to Drop first =True, because the variables can be easily explained by k-1 terms. Keeping or not keeping them depends upon your case, whether to get k-1 dummies out of k categorical levels by removing the first level. Please note default = False, meaning that the reference is not dropped and k dummies created out of k categorical levels. Keeping it will just be additional redundant feature.

We can substitute *one dummy variable* in configuration -1 with its value in Configuration -2. This actually means that (at least) one of the features we are working with is redundant- that feature could be any one of the three, since configuration-2 could be written with any one of them in the left-hand side. So, we are making our model learn an additional weight which is not really needed. This consumes computational power and time. This also gives an optimisation objective that might not be very reasonable and might also be difficult to work with. Too many independent variables may lead to Curse of Dimensionality. If multicollinearity also comes along with that, things become worse.

We not only want our model to predict well, but we also want it to be interpretable.

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

3. Note: Casual and Registered are part of "cnt" variable. If it has to be considered then "registered" is most correlated. Else**

***(Most correlated is "atemp")

Over Excel :

	season	yr	mnth	holiday	weekday	workingd	weathersi	temp	atemp	hum	windspee	casual	registered	cnt
season	1													
yr	-2E-17	1												
mnth	0.83103	-2E-17	1											
holiday	-0.0109	0.0082	0.0189	1										
weekday	-0.0031	-0.0055	0.00952	-0.102	1									
workingd	0.01376	-0.0029	-0.0047	-0.2529	0.0358	1								
weathersi	0.02131	-0.0503	0.04561	-0.0344	0.03111	0.06024	1							
temp	0.33336	0.04879	0.21908	-0.0288	-0.0002	0.05347	-0.1195	1						
atemp	0.34201	0.04722	0.22643	-0.0327	-0.0075	0.05294	-0.1206	0.9917	1					
hum	0.20822	-0.1125	0.22494	-0.0157	-0.0523	0.0232	0.59028	0.12856	0.14151	1				
windspee	-0.2296	-0.0116	-0.208	0.00626	0.01428	-0.0187	0.03977	-0.1582	-0.1839	-0.2485	1			
casual	0.20874	0.2505	0.12121	0.05405	0.05998	-0.5176	-0.246	0.54273	0.54336	-0.0752	-0.168	1		
registered	0.41031	0.59691	0.29195	-0.1091	0.05743	0.30544	-0.259	0.53944	0.54368	-0.0892	-0.2179	0.39414	1	
cnt	0.40458	0.56973	0.27819	-0.0688	0.06753	0.06254	-0.2959	0.62704	0.63069	-0.0985	-0.2351	0.67212	0.945410612	1

Over Python:

	season	yr	mnth	holiday	weekday	workingd	weathersi	temp	atemp	hum	windspee	casual	registered	cnt
season	1.0000 00e+00	- 2.2481	8.3103 21e-01	- 0.0108	- 0.0030	0.0137 62	0.0213 06	0.3333 61	0.3420 14	0.2082 20	- 0.2296	0.2087 37	0.4103 10	0.4045 84
yr	- 2.2481	1.0000 00e+00	- 2.4701	0.0081 95	- 0.0054	- 0.0029	- 0.0503	0.0487 89	0.0472 15	- 0.1125	- 0.0116	0.2505 02	0.5969 11	0.5697 28
mnth	8.3103 21e-01	- 2.4701	1.0000 00e+00	0.0189 05	0.0095 23	- 0.0046	0.0456 13	0.2190 83	0.2264 30	0.2249 37	- 0.2080	0.1212 08	0.2919 52	0.2781 91
holiday	- 1.0868	8.1953 45e-03	1.8904 83e-02	1.0000 00	- 0.1019	- 0.2529	- 0.0343	- 0.0287	- 0.0327	- 0.0156	0.0062 57	0.0540 55	- 0.1091	- 0.0687
weekday	- 3.0811	- 5.4663	9.5229 69e-03	- 0.1019	1.0000 00	0.0358 00	0.0311 12	- 0.0001	- 0.0075	- 0.0522	0.0142 83	0.0599 78	0.0574 27	0.0675 34
workingda	1.3761 78e-02	- 2.9453	- 4.6879	- 0.2529	0.0358 00	1.0000 00	0.0602 36	0.0534 70	0.0529 40	0.0232 02	- 0.0186	- 0.5176	0.3054 37	0.0625 42
weathersit	2.1306 36e-02	- 5.0322	4.5613 35e-02	- 0.0343	0.0311 12	0.0602 36	1.0000 00	- 0.1195	- 0.1205	0.5902 77	0.0397 69	- 0.2460	- 0.2590	- 0.2959
temp	3.3336 07e-01	4.8789 19e-02	2.1908 33e-01	- 0.0287	- 0.0001	0.0534 70	- 0.1195	1.0000 00	0.9916 96	0.1285 65	- 0.1581	0.5427 31	0.5394 36	0.6270 44
atemp	3.4201 39e-01	4.7215 19e-02	2.2643 02e-01	- 0.0327	- 0.0075	0.0529 40	- 0.1205	0.9916 96	1.0000 00	0.1415 12	- 0.1838	0.5433 62	0.5436 78	0.6306 85
hum	2.0821 96e-01	- 1.1254	2.2493 68e-01	- 0.0156	- 0.0522	0.0232 02	0.5902 77	0.1285 65	0.1415 12	1.0000 00	- 0.2485	- 0.0752	- 0.0892	- 0.0985
windspeed	- 2.2960	- 1.1624	- 2.0801	0.0062 57	0.0142 83	- 0.0186	0.0397 69	- 0.1581	- 0.1838	- 0.2485	1.0000 00	- 0.1679	- 0.2179	- 0.2351
casual	2.0873 73e-01	2.5050 17e-01	1.2120 79e-01	0.0540 55	0.0599 78	- 0.5176	- 0.2460	0.5427 31	0.5433 62	- 0.0752	- 0.1679	1.0000 00	0.3941 37	0.6721 23
registered	4.1031 02e-01	5.9691 06e-01	2.9195 16e-01	- 0.1091	0.0574 27	0.3054 37	- 0.2590	0.5394 36	0.5436 78	- 0.0892	- 0.2179	0.3941 37	1.0000 00	0.9454 11
cnt	4.0458 38e-01	5.6972 85e-01	2.7819 09e-01	- 0.0687	0.0675 34	0.0625 42	- 0.2959	0.6270 44	0.6306 85	- 0.0985	- 0.2351	0.6721 23	0.9454 11	1.0000 00

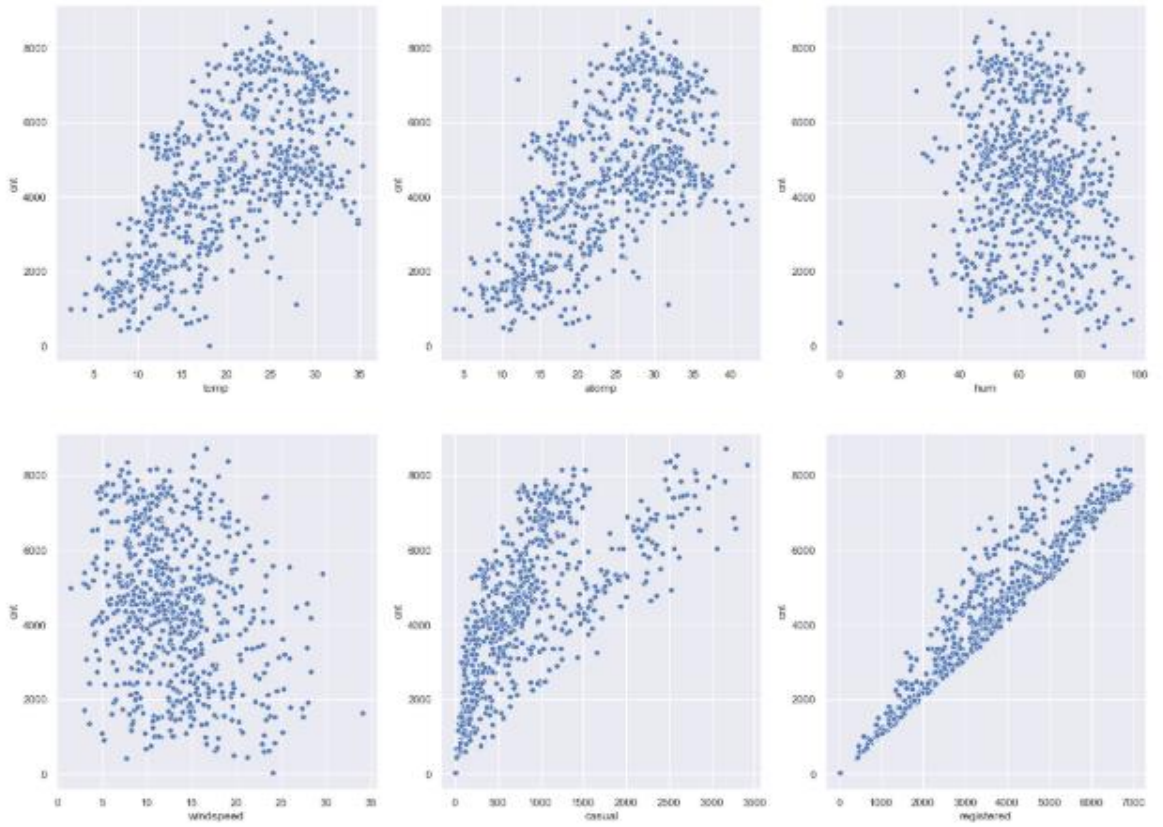
Finding correlation and then dropping the most co-related variables.

boom_bikes.corr()

Visualizing using the "cnt"as Target Variable

In [12]:
```
plt.figure(figsize=(20,15))
plt.subplot(2,3,1)
sns.scatterplot(x='temp', y='cnt' ,data=boom_bikes)
plt.subplot(2,3,2)
sns.scatterplot(x='atemp', y='cnt', data=boom_bikes)
plt.subplot(2,3,3)
sns.scatterplot(x='hum', y='cnt', data=boom_bikes)
plt.subplot(2,3,4)
sns.scatterplot(x='windspeed', y='cnt', data=boom_bikes)
plt.subplot(2,3,5)
sns.scatterplot(x='casual', y='cnt', data=boom_bikes)
plt.subplot(2,3,6)
sns.scatterplot(x='registered', y='cnt', data=boom_bikes)
plt.show()

```



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

Validation on training set depends on the following factors:

1. R2 and adjusted is towards higher side.
2. P-Value should not be greater than 0.05
3. VIF should not exceed value 5.
4. Correlation of any two independent variable is not too much.

Question 5: Based on the final model, which are the top 3 features contributing significantly towards explaining the demand of the shared bikes?

Top Three features contributing significantly towards explaining the demand of the shared bikes are:

1. Atemp : On the addition of this variable the R2 of the model greatly improved
2.

General Subjective Questions

Question 1. Explain the linear regression algorithm in detail.

Answer: *Linear regression is used for finding linear relationship between target and one or more predictors. There are two types of linear regression:*

a. Simple Linear Regression

b. Multiple Linear Regression

Simple linear regression is useful for finding relationship between two continuous variables. One is predictor or independent variable and other is response or dependent variable. It looks for statistical relationship but not deterministic relationship. Relationship between two variables is said to be deterministic if one variable can be accurately expressed by the other.

For doing this we predict a set of datapoints such that they are placed closed to the actual data points such that the predicted datapoints form a best fit line represented by:

Y=bo+b1X, where Y is the dependent variable and X are the independent variables. bo & b1 are intercept and Slope respectively.

Question 2. Explain the Anscombe's quartet in detail.

Anscombe's quartet : These are called quartet because it comprises four datasets that have nearly identical simple statistical description, yet have very different distributions and appear very different when graphed.

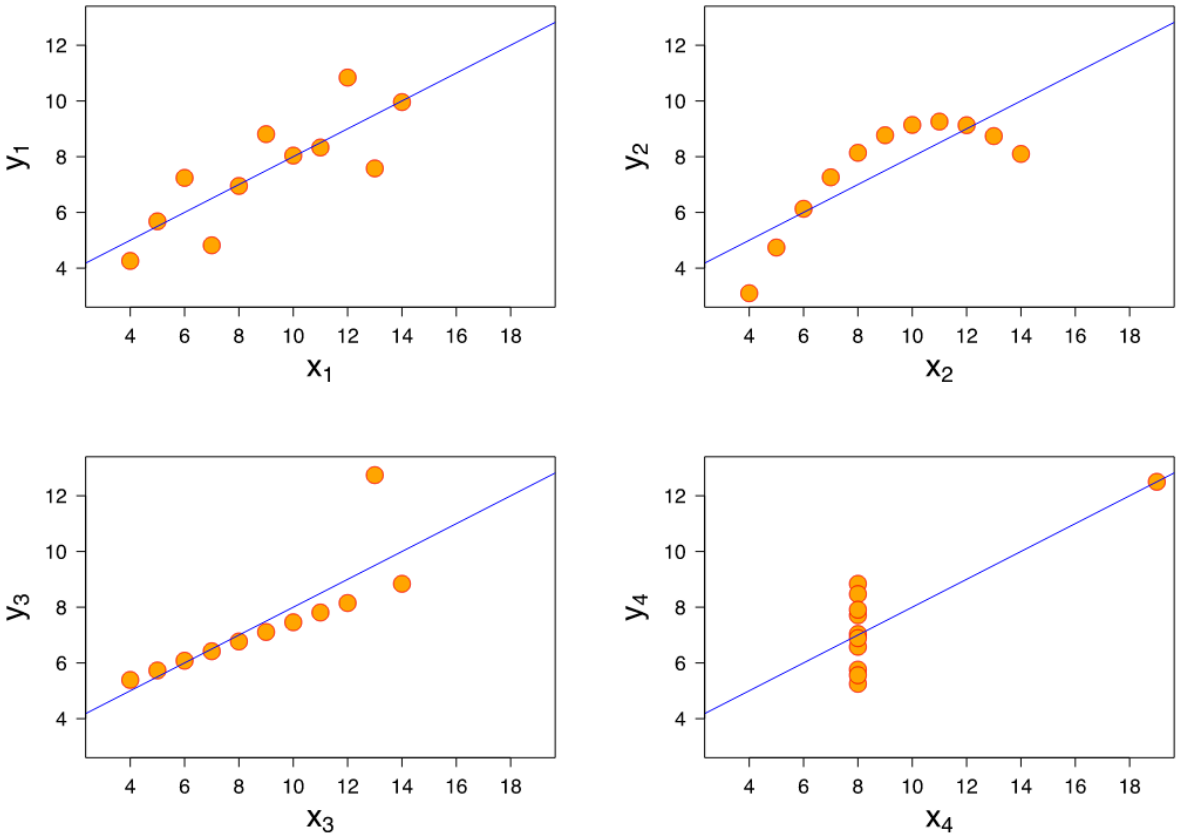
Perhaps the most elegant demonstration of the dangers of summary statistics is *Anscombe's Quartet*. It's a group of four datasets that appear to be similar when using typical summary statistics, yet tell four different stories when graphed. Each dataset consists of eleven *(x,y)* pairs as follows:

I		**II**		**III**		**IV**	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

All the summary statistics you'd think to compute are close to identical:

- The average x value is 9 for each dataset
- The average y value is 7.50 for each dataset
- The variance for x is 11 and the variance for y is 4.12
- The correlation between x and y is 0.816 for each dataset
- A linear regression (line of best fit) for each dataset follows the equation $y = 0.5x + 3$

So far these four datasets appear to be pretty similar. But when we plot these four data sets on an x/y coordinate plane, we get the following results:



Now we see the real relationships in the datasets start to emerge. Dataset I consist of a set of points that appear to follow a rough linear relationship with some variance. Dataset II fits a neat curve but doesn't follow a linear relationship (maybe it's quadratic?). Dataset III looks like a tight linear relationship between x and y, except for one large outlier. Dataset IV looks like x remains constant, except for one outlier as well. Computing summary statistics or staring at the data wouldn't have told us any of these stories. Instead, it's important to visualize the data to get a clear picture of what's going on.

Question 3. What is Pearson's R?

The degree at which the two variables are related to each other , defines correlation . Pearson Product Moment Correlation (called Pearson's Correlation for short) is the most common measure of correlation.

When measured in population, it is designated by the Greek letter rho (ρ). When computed in a sample, it is designated by the letter "r" and is sometimes called "Pearson's r." Pearson's correlation reflects the degree of linear relationship between two variables.
Range: +1 to -1
+1 : Means that there is a perfect positive linear relationship between variables. It is a positive relationship because high scores on the X-axis are associated with high scores on the Y-axis.
-1 : Indicated perfect negative correlation. It means as the one increases then other will tend to reduce.

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

Scaling: Scaling is the term defined to map one variable in to the range of other defined variable. For example X ranges from 1-1000. And we have to scale it in terms of 'y' which ranges from 10-100. So, conversion of every x term into y term is called scaling.

Why Scaling:

Most of the times, your dataset will contain features highly varying in magnitudes, units and range. The difference in 5kg and 5000gm is of the unit, because of which the magnitude have changed a lot. The analysis algorithm will consider higher magnitude and can make large impact on the result. To supress this effect, we need to bring all features to the same level of magnitudes. This can be achieved by scaling.

How to Scale Features

There are four common methods to perform Feature Scaling.

1. **Standardisation:**

Standardization is another scaling technique where the values are centred around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.

Standardisation replaces the values by their Z scores.

$$x' = \frac{x - \bar{x}}{\sigma}$$

#data standardization with sklearn

```
# data
standardization
with  sklearn
                from sklearn.preprocessing import StandardScaler

                # copy of datasets
                X_train_stand = X_train.copy()
                X_test_stand = X_test.copy()

                # numerical features
                num_cols =
                ['Item_Weight','Item_Visibility','Item_MRP','Outlet_Establishment_Year']

                # apply standardization on numerical features
                for i in num_cols:

                    # fit on training data column
                    scale = StandardScaler().fit(X_train_stand[[i]])

                    # transform the training data column
                    X_train_stand[i] = scale.transform(X_train_stand[[i]])

                    # transform the testing data column
                    X_test_stand[i] = scale.transform(X_test_stand[[i]])
```

2. **Mean Normalisation:**
   Normalization is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling.

$$x' = \frac{x - \text{mean}(x)}{\max(x) - \min(x)}$$

This distribution will have values between **-1 and 1**with **μ=0**.

3. **Min-Max Scaling:**

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

This scaling brings the value between 0 and 1.

4. **Unit Vector:**

$$x' = \frac{x}{||x||}$$

Scaling is done considering the whole feature vector to be of unit length. Dealing features with hard boundaries this is quite useful. For example, when dealing with image data, the colors can range from only 0 to 255.

When to Scale

Some examples of algorithms where feature scaling matters are:

- **k-nearest neighbors** with an Euclidean distance measure is sensitive to magnitudes and hence should be scaled for all features to weigh in equally.
- While performing **Principal Component Analysis(PCA),** scaling is critical. It tries to get the features with maximum variance and the variance is high for high magnitude features. This skews the PCA towards high magnitude features.
- We can speed up **gradient descent** by scaling. This is because θ will descend quickly on small ranges and slowly on large ranges, and so will oscillate inefficiently down to the optimum when the variables are very uneven.
- **Tree based models** are not distance based models and can handle varying ranges of features. Hence, Scaling is not required while modelling trees.

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

VIF is abbreviation of Variation Inflation Factor .VIF is an index that provides a measure of how much the variance of an estimated regression coefficient increases due to collinearity. In order to determine VIF, we fit a regression model between the independent variables. For example, we would fit the following models to estimate the coefficient of determination R1 and use this value to estimate the VIF:

$$X_1=C+ \alpha_2 X_2+\alpha_3 X_3+\cdots$$

$$〖VIF〗_1=1/(1-R_1^2)$$

Next, we fit the model between X2 and the other independent variables to estimate the coefficient of determination R2:

$$X_2=C+ \alpha_1 X_1+\alpha_3 X_3+\cdots$$

$$〖VIF〗_2=1/(1-R_2^2)$$

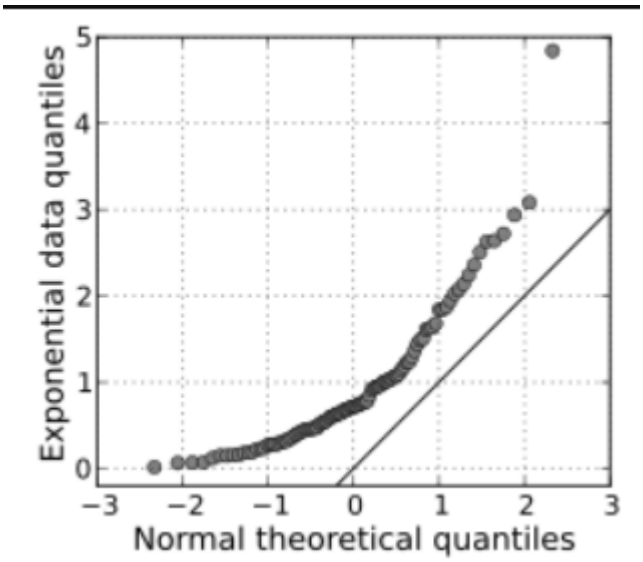
If all the independent variables are orthogonal to each other, then VIF = 1.0. If there is perfect correlation, then VIF = infinity. A large value of VIF indicates that there is a correlation between the variables. If the VIF is 4,

this means that the variance of the model coefficient is inflated by a factor of 4 due to the presence of multicollinearity. This would mean that that standard error of this coefficient is inflated by a factor of 2 (square root of variance is the standard deviation). The standard error of the coefficient determines the confidence interval of the model coefficients. If the standard error is large, then the confidence intervals may be large, and the model coefficient may come out to be non-significant due to the presence of multicollinearity. A general rule of thumb is that if VIF > 10 then there is multicollinearity. Note that this is a rough rule of thumb, in some cases we might choose to live with high VIF values if it does not affect our model results such as when we are fitting a quadratic or cubic model or depending on the sample size a large value of VIF may not necessarily indicate a poor model.

VIF = 1 : No Multicollinearity (Good)
VIF = 4-5 : Moderate collinearity (considerable)
VIF: >=10 : Very High collinearity. (Bad)

Question 5. What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression.

Q Q Plots (Quantile-Quantile plots) are plots of two quantiles against each other. A quantile is a fraction where certain values fall below that quantile. For example, the median is a quantile where 50% of the data fall below that point and 50% lie above it. The purpose of Q Q plots is to find out if two sets of data come from the same distribution. A 45 degree angle is plotted on the Q Q plot; if the two data sets come from a common distribution, the points will fall on that reference line.



The image above shows quantiles from a theoretical normal distribution on the horizontal axis. It's being compared to a set of data on the y-axis. This particular type of Q Q plot is called a **normal quantile-quantile (QQ)**

plot. The points are not clustered on the 45 degree line, and in fact follow a curve, suggesting that the sample data is not normally distributed.