General Subjective Questions

[QUESTION]
What is Pearson's R?
[ANSWER]
Correlation coefficient formulas are used to find how strong a relationship is between data. The formulas return a value between -1 and 1, where:

- 1 indicates a strong positive relationship.
- -1 indicates a strong negative relationship.
- A result of zero indicates no relationship at all

Correlation between sets of data is a measure of how well they are related. The most common measure of correlation in stats is the Pearson Correlation. The full name is the Pearson Product Moment Correlation (PPMC). It shows the linear relationship between two sets of data.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[\, n\sum x^2 - (\sum x)^2 \,][\, n\sum y^2 - (\sum y)^2 \,]}}$$

Fig : Formula of Pearson's R

The correlation coefficient ranges from −1 to 1. An absolute value of exactly 1 implies that a linear equation describes the relationship between X and Y perfectly, with all data points lying on a line. The correlation sign is determined by the regression slope: a value of +1 implies that all data points lie on a line for which Y increases as X increases, and vice versa for -1. A value of 0 implies that there is no linear dependency between the variables.

As a simple example, one would expect the age and height of a sample of teenagers from a high school to have a Pearson correlation coefficient significantly greater than 0, but less than 1 (as 1 would represent an unrealistically perfect correlation). Another example below :

Person	Age (*x*)	Score (*y*)	(*xy*)	(*x*^2)	(*y*^2)
1	43	99	4257	1849	9801
2	21	65	1365	441	4225
3	25	79	1975	625	6241
4	42	75	3150	1764	5625
5	57	87	4959	3249	7569
6	59	81	4779	3481	6561
Total	247	486	20485	11409	40022

Using above Pearson R formula, we get answer as r=**0.5298**

[QUESTION]

Explain the Anscombe's quartet in detail.

[ANSWER]

Anscombe's Quartet can be defined as a group of four data sets which are nearly identical in simple descriptive statistics, but there are some peculiarities in the dataset that fools the regression model if built. They have very different distributions and appear differently when plotted on scatter plots.

It was constructed in 1973 by statistician Francis Anscombe to illustrate the importance of plotting the graphs before analyzing and model building, and the effect of other observations on statistical properties. There are these four data set plots which have nearly same statistical observations, which provides same statistical information that involves variance, and mean of all x,y points in all four datasets.

This tells us about the importance of visualising the data before applying various algorithms to build models out of them, which suggests that the data features must be plotted in order 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. Also, the Linear Regression can be only be considered a fit for the data with linear relationships and is incapable of handling any other kind of datasets.

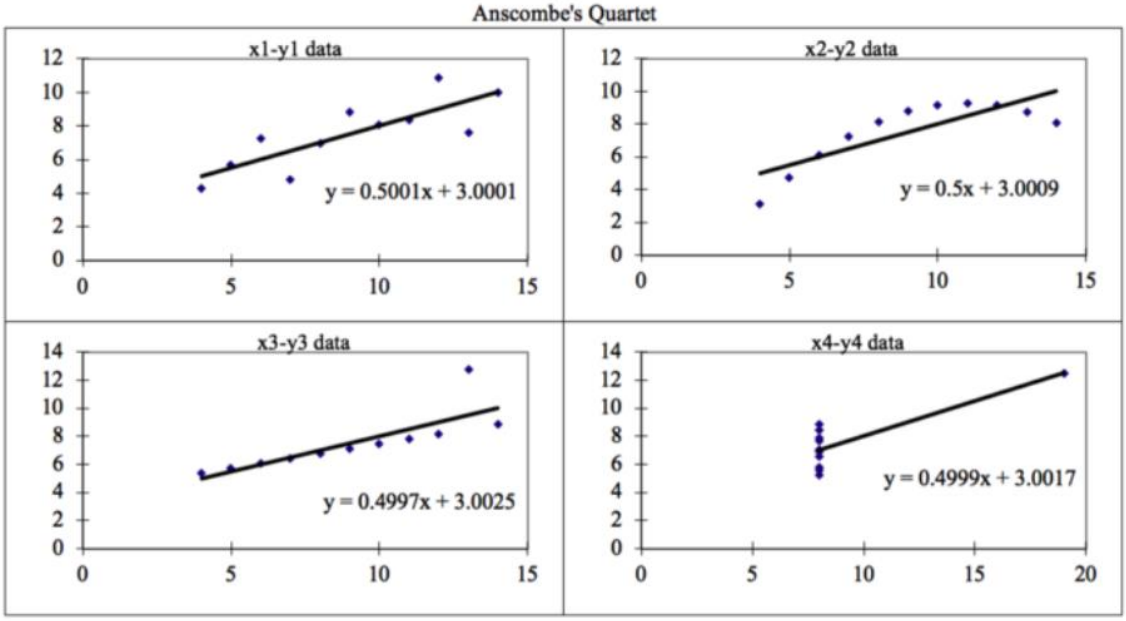


Fig :Anscombe's quartet

[QUESTION]

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

[ANSWER]

Q plots are also known as Quantile-Quantile plots. As the name suggests, they plot the quantiles of a sample distribution against quantiles of a theoretical distribution. Doing this helps us determine if a dataset follows any particular type of probability distribution like normal, uniform, exponential.

The power of Q-Q plots lies in their ability to summarize any distribution visually.

QQ plots is very useful to determine

- If two populations are of the same distribution.
- If residuals follow a normal distribution. Having a normal error term is an assumption in regression and we can verify if it's met using this.
- Skewness of distribution.

In Q-Q plots, we plot the theoretical Quantile values with the sample Quantile values. Quantiles are obtained by sorting the data. It determines how many values in a distribution are above or below a certain limit.

This helps in a scenario of linear regression when we have training and test data set received separately and then we can confirm using Q-Q plot that both the data sets are from populations with same distributions.

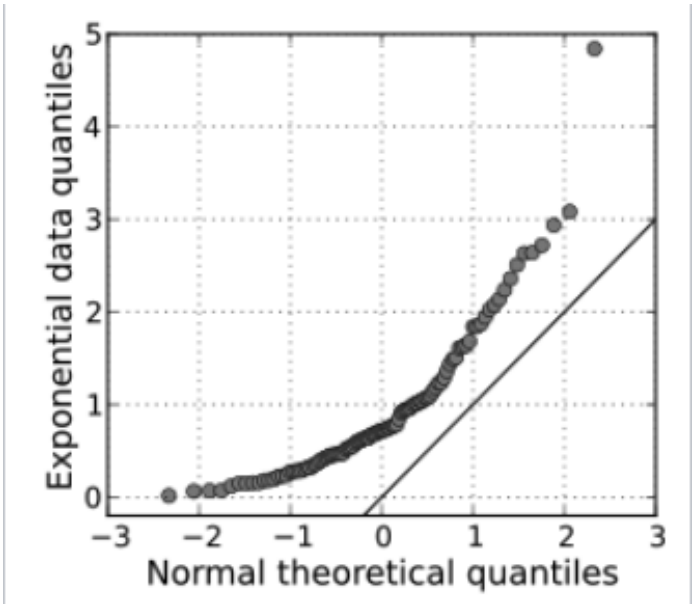


Fig : Example of a QQ plot

[QUESTION]
You might have observed that sometimes the value of VIF is infinite. Why does this happen?
[ANSWER]

If there is perfect correlation, then VIF = infinity. This shows a perfect correlation between two independent variables. In the case of perfect correlation, we get R2 =1, which lead to 1/(1-R2) infinity. To solve this problem we need to drop one of the variables from the dataset which is causing this perfect multicollinearity.

An infinite VIF value indicates that the corresponding variable may be expressed exactly by a linear combination of other variables (which show an infinite VIF as well).

[QUESTION]
What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling?
[ANSWER]

It is a step of data Pre-Processing which is applied to independent variables to normalize the data within a particular range. It also helps in speeding up the calculations in an algorithm.

Most of the times, collected data set contains features highly varying in magnitudes, units and range. If scaling is not done then algorithm only takes magnitude in account and not units hence incorrect modelling. To solve this issue, we have to do scaling to bring all the variables to the same level of magnitude.

It is important to note that scaling just affects the coefficients and none of the other parameters like t-statistic, F-statistic, p-values, R-squared, etc.

Normalization/Min-Max Scaling:

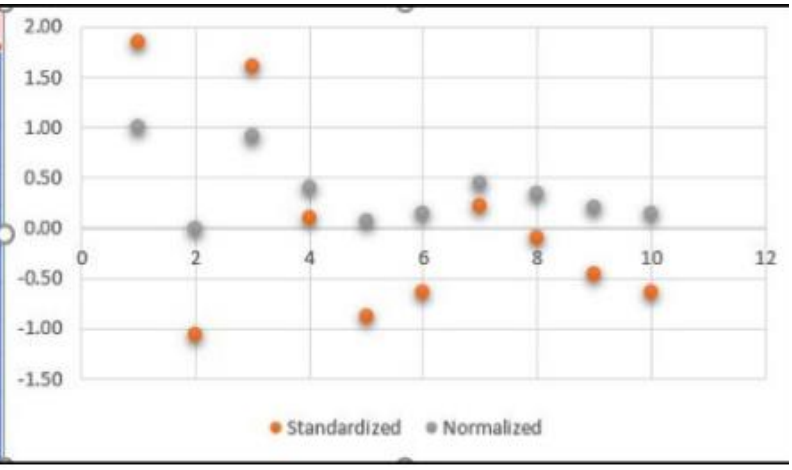
- It brings all of the data in the range of 0 and 1. sklearn.preprocessing.MinMaxScaler helps to implement normalization in python.

Standardization Scaling:

- Standardization replaces the values by their Z scores. It brings all of the data into a standard normal distribution which has mean (μ) zero and standard deviation one (σ).

Below shows example of Standardized and Normalized scaling on original values.

Original	Standardized	Normalized
59	1.85	1.000
11	-1.06	0.000
55	1.61	0.917
30	0.09	0.396
14	-0.88	0.063
18	-0.64	0.146
32	0.21	0.438
27	-0.09	0.333
21	-0.46	0.208
18	-0.64	0.146



[QUESTION]
Explain the linear regression algorithm in detail.
[ANSWER]
Machine learning models can be classified into the following three types based on the task performed and the nature of the output:
1. **Regression**: The output variable to be predicted is a **continuous variable**, e.g. scores of a student

2. **Classification**: The output variable to be predicted is a **categorical variable**, e.g. incoming emails as spam or ham

3. **Clustering**: **No predefined notion of label** allocated to groups/clusters formed, e.g. customer segmentation for generating discounts

Regression falls under supervised learning methods. There are two types of linear regression

● **Simple Linear Regression**

The most elementary type of regression model is the simple linear regression which explains the relationship between a dependent variable and one independent variable using a straight line. The straight line is plotted on the scatter plot of these two points

The best-fit line is found by minimising the expression of RSS (Residual Sum of Squares) which is equal to the sum of squares of the residual for each data point in the plot. Residuals for any data point is found by subtracting predicted value of dependent variable from actual value of dependent variable.

The strength of the linear regression model can be assessed using 2 metrics:
1. R2 or Coefficient of Determination
2. Residual Standard Error (RSE)

Equation : $y = B0 + B1(X)$
Where y = dependent variable value, X = Independent variable value, B0 = y-intercept and B1=Beta coefficient for X

● **Multiple Linear regression**

Multiple linear regression is a statistical technique to understand the relationship between one dependent variable and several independent variables (explanatory variables).
The objective of multiple regression is to find a linear equation that can best determine the value of dependent variable Y for different values independent variables in X.
Equation : $y = B0 + B1(X) + B2(X2) + \ldots\ldots + Bn(Xn)$

1) In statistical modelling, linear regression is a process of estimating the relationship among variables. The focus here is to establish the relationship between a dependent variable and one or more independent variable(s). Independent variables are also called as 'predictors'.
2)Regression helps you understand how the values of dependent variable changes as you change the values of 1 predictor, holding the other predictors static (or same). This means that simple linear regression in its most basic form doesn't allow you to change all the predictors at a time and measure the impact on the dependent variable. You can only change 1 at a time.
3) Regression only shows relationship, i.e. correlation and NOT causality. In a very restrictive environment, regression may show causality. However, if you blindly interpret regression results as causation, it may lead to false insights. Correlation does not imply causation.
4) Regression analysis is widely used for 2 purposes: a) Forecasting and b) Prediction. The uses of forecasting and prediction have substantial overlap. However, they are different and it's important to understand why, to be able to use regression effectively for each purpose. Regression guarantees 'interpolation' but not necessarily 'extrapolation'.
5) Linear regression is a form of parametric regression.

Assignment-based Subjective Questions

[QUESTION]

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

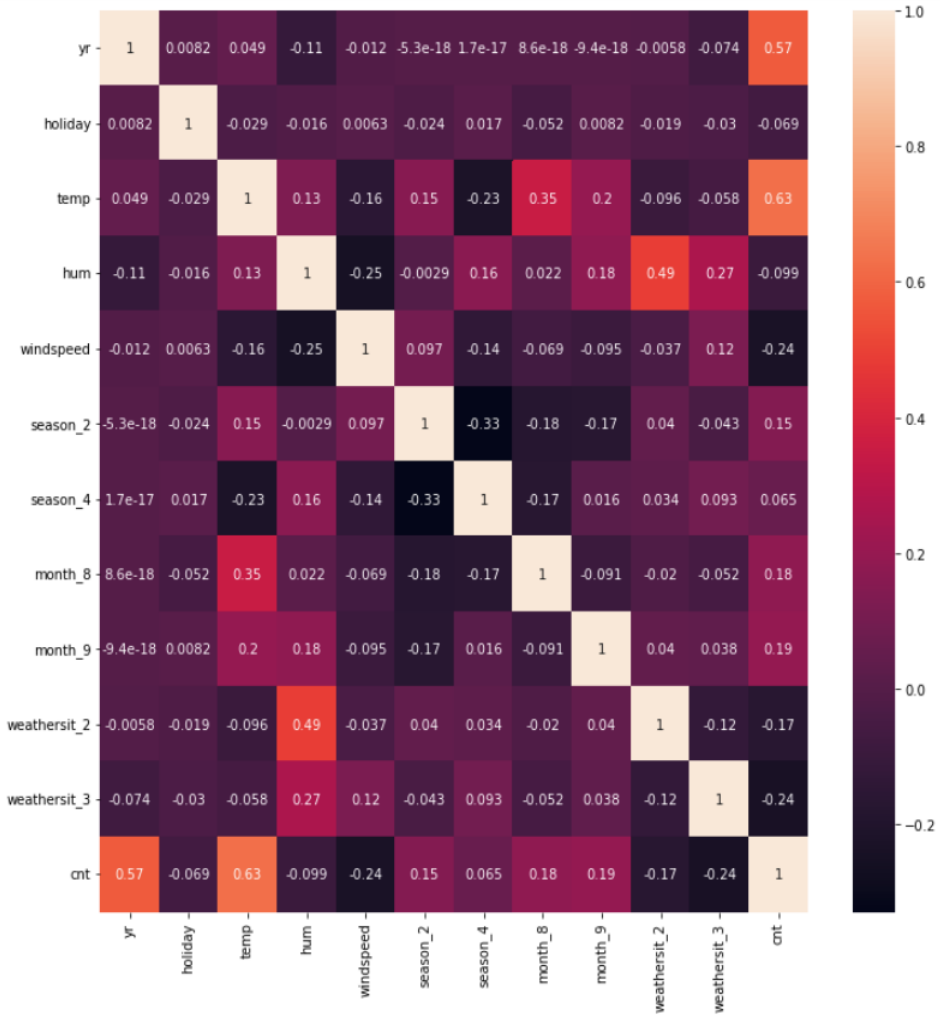
[ANSWER]

It is observed that "temp" is a significant variable as it is having high beta coefficient = 4690.8361 and also evident by the fact that it has 0.63 correlation with "cnt" target variable. Thus, people usually prefer to use bikes when temperature is not cold i.e. temperature is conducive for bike riding.

It is observed that "yr" is a significant variable as it is having high beta coefficient = +1986.6570 and also evident by the fact that it has 0.57 correlation with "cnt" target variable. As "yr" takes two values (0 for 2018 and 1 for 2019), we can conclude that the demand for bike-sharing system is increasing on yearly basis.

It is observed that "windspeed" is a significant variable as it is having high negative beta coefficient = -1665.6437 and also evident by the fact that it has -0.24 correlation with "cnt" target variable. Thus, when windspeed increases, the count of the people using bikes goes down as people avoid bike riding in high windy conditions.

It is observed that "weathersit3" (Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds) is a significant variable as it is having high negative beta coefficient = -2031.7881 and also evident by the fact that it has -0.24 correlation with "cnt" target variable. This means people do not prefer such weather conditions (Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds)


yr 1 0.0082 0.049 -0.11 -0.012 -5.3e-18 1.7e-17 8.6e-18 -9.4e-18 -0.0058 -0.074 0.57
holiday 0.0082 1 -0.029 -0.016 0.0063 -0.024 0.017 -0.052 0.0082 -0.019 -0.03 -0.069
temp 0.049 -0.029 1 0.13 -0.16 0.15 -0.23 0.35 0.2 -0.096 -0.058 0.63
hum -0.11 -0.016 0.13 1 -0.25 -0.0029 0.16 0.022 0.18 0.49 0.27 -0.099
windspeed -0.012 0.0063 -0.16 -0.25 1 0.097 -0.14 -0.069 -0.095 -0.037 0.12 -0.24
season_2 -5.3e-18 -0.024 0.15 -0.0029 0.097 1 -0.33 -0.18 -0.17 0.04 -0.043 0.15
season_4 1.7e-17 0.017 -0.23 0.16 -0.14 -0.33 1 -0.17 0.016 0.034 0.093 0.065
month_8 8.6e-18 -0.052 0.35 0.022 -0.069 -0.18 -0.17 1 -0.091 -0.02 -0.052 0.18
month_9 -9.4e-18 0.0082 0.2 0.18 -0.095 -0.17 0.016 -0.091 1 0.04 0.038 0.19
weathersit_2 -0.0058 -0.019 -0.096 0.49 -0.037 0.04 0.034 -0.02 0.04 1 -0.12 -0.17
weathersit_3 -0.074 -0.03 -0.058 0.27 0.12 -0.043 0.093 -0.052 0.038 -0.12 1 -0.24
cnt 0.57 -0.069 0.63 -0.099 -0.24 0.15 0.065 0.18 0.19 -0.17 -0.24 1
yr holiday temp hum windspeed season_2 season_4 month_8 month_9 weathersit_2 weathersit_3 cnt
1.0 0.8 0.6 0.4 0.2 0.0 −0.2

OLS Regression Results

Dep. Variable:	cnt	**R-squared:**	0.841
Model:	OLS	**Adj. R-squared:**	0.838
Method:	Least Squares	**F-statistic:**	239.5
Date:	Tue, 14 Dec 2021	**Prob (F-statistic):**	7.50e-191
Time:	12:31:27	**Log-Likelihood:**	-4117.9
No. Observations:	510	**AIC:**	8260.
o scroll output; double click to hide	498	**BIC:**	8311.
Df Model:	11		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2108.3028	239.671	8.797	0.000	1637.412	2579.193
yr	1986.6570	70.594	28.142	0.000	1847.958	2125.356
holiday	-828.1994	222.664	-3.720	0.000	-1265.676	-390.723
temp	4690.8361	191.430	24.504	0.000	4314.726	5066.946
hum	-1546.7261	326.335	-4.740	0.000	-2187.888	-905.564
windspeed	-1665.6437	232.599	-7.161	0.000	-2122.639	-1208.648
season_2	904.9010	94.646	9.561	0.000	718.947	1090.855
season_4	1278.9715	92.699	13.797	0.000	1096.842	1461.102
month_8	481.0181	141.820	3.392	0.001	202.379	759.657
month_9	1063.9061	140.911	7.550	0.000	787.054	1340.759
weathersit_2	-459.1888	91.265	-5.031	0.000	-638.500	-279.878
weathersit_3	-2031.7881	228.828	-8.879	0.000	-2481.376	-1582.200

[QUESTION]

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

[ANSWER]

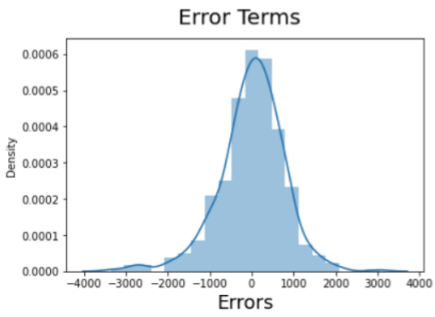
We have plotted a distribution plot which highlights the distribution of error terms. Below distribution plot clearly indicates that the assumptions of Linear regression are satisfied - As we can see a normal distribution (bell shaped curve for error terms) with mean close to 0. Same has been answered with help of a distribution plot in - Step 6: Residual Analysis of the train data.

Step 6: Residual Analysis of the train data

```
In [312]: y_train_pred = lr.predict(X_train_rfe)
```

```
In [313]: # Plot the histogram of the error terms
          fig = plt.figure()
          sns.distplot((y_train - y_train_pred), bins = 20)
          fig.suptitle('Error Terms', fontsize = 20)                  # Plot heading
          plt.xlabel('Errors', fontsize = 18)
```

```
Out[313]: Text(0.5, 0, 'Errors')
```

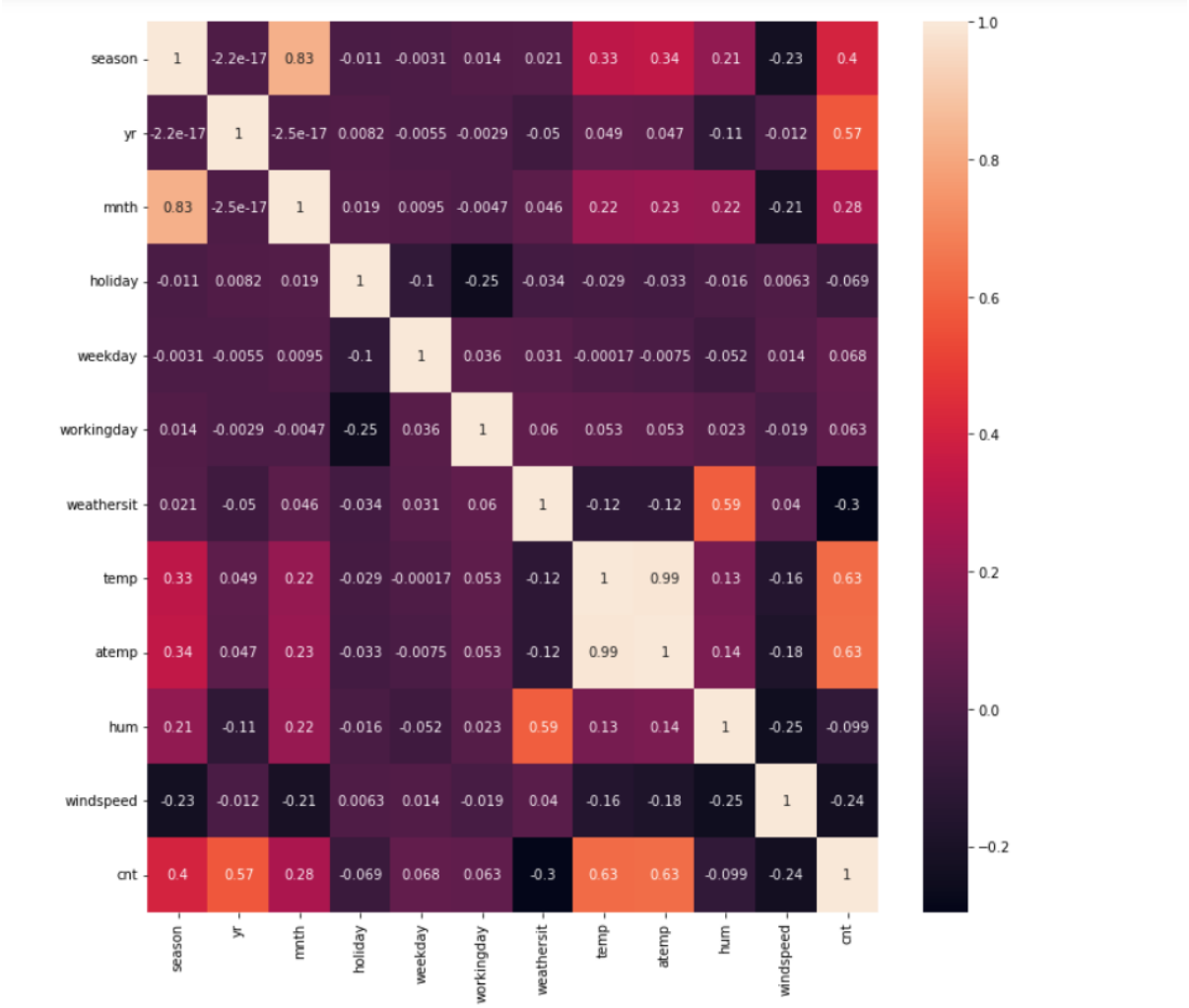


[QUESTION]

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

[ANSWER]

Looking at the pair-plot among the numerical variables, 'temp' variable has the highest correlation of 0.63 with the 'cnt' target variable, followed by 'yr' with correlation of 0.57 with 'cnt' target variable. Same has been answered with help of a heatmap in - Step 2 : Visualising the data



[QUESTION]

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

[ANSWER]

drop_first=True is important to use, as it helps in reducing the extra column created during dummy variable creation. Hence it reduces the correlations created among dummy variables i.e. it reduces multi-collinearity.

Dummy variables

In [52]:
```
#Creating dummy variable for season
season_dummy_var = pd.get_dummies(bike['season'], drop_first = True, prefix='season')
season_dummy_var.value_counts()
```

Out[52]:
```
season_2  season_3  season_4
0         1         0           188
1         0         0           184
0         0         0           180
                    1           178
dtype: int64
```

In [53]:
```
#Creating dummy variable for month
month_dummy_var = pd.get_dummies(bike['mnth'], drop_first = True, prefix='month')
month_dummy_var.value_counts()
```

Out[53]:
```
month_2  month_3  month_4  month_5  month_6  month_7  month_8  month_9  month_10  month_11  month_12
0        1        0        0        0        0        0        0        0         0         0           62
         0        0        1        0        0        0        0        0         0         0           62
                           0        0        1        0        0        0         0         0           62
                                    0        0        1        0        0         0         0           62
                                             0        0        1        0         0         0           62
                                                      0        0        1         0         0           62
                                                               0        0         0         0           62
                  1        0        0        0        0        0        0         0         0           60
                  0        0        1        0        0        0        0         0         0           60
                           0        0        0        1        0        0         0         0           60
                                                      0        0        0         1         0           60
1        0        0        0        0        0        0        0        0         0         0           56
dtype: int64
```

In [54]:
```
#Creating dummy variable for weekday
weekday_dummy_var = pd.get_dummies(bike['weekday'], drop_first = True, prefix='weekday')
weekday_dummy_var.value_counts()
```

Out[54]:
```
weekday_1  weekday_2  weekday_3  weekday_4  weekday_5  weekday_6
1          0          0          0          0          0            105
0          0          0          0          0          1            105
                                             0          0            105
           1          0          0          0          0            104
           0          0          1          0          0            104
                                 0          1          0            104
                      1          0          0          0            103
dtype: int64
```

In [55]:
```
#Creating dummy variable for weathersit
weathersit_dummy_var = pd.get_dummies(bike['weathersit'], drop_first = True, prefix='weathersit')
weathersit_dummy_var.value_counts()
```

Out[55]:
```
weathersit_2  weathersit_3
0             0              463
1             0              246
0             1               21
dtype: int64
```

In [56]:
```
#Concatenate all created dummy variables to roiginal 'bike' dataframe
bike = pd.concat([bike,season_dummy_var],axis = 1)
bike = pd.concat([bike,month_dummy_var],axis = 1)
bike = pd.concat([bike,weekday_dummy_var],axis = 1)
bike = pd.concat([bike,weathersit_dummy_var],axis = 1)
```

[QUESTION]

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

[ANSWER]

Below are the categorical variables in the dataset identified in the dataset and their respective correlation value with 'cnt' target variable (refer the heatmap from - Step 2 : Visualising the data:

1. Season : 0.4
2. Month : 0.28
3. Weekday : 0.063
4. weather_sit : -0.3

But this may not paint the best picture as model may interpreted them as ordinal variables (thus implying, for example that month_1 is less important than month_12 or season_4 is better than season_1).

Hence, we converted this categorical variables into dummy variables (please refer - # Step 3: Data Preparation) for more details.

After training the model using training data using an automated approach of RFE + manual method of variable selection, we got 11 variables/features from the model that contribute in prediction powers & out of those 11, season_2, season_4, month_8, month_9, weather_sit2 and weather_sit3 are 6 of the dummy variables.

Below are their respective correlation value with 'cnt' target variable

1. weather_sit3 : -0.24
2. weather_sit2 : -0.17
3. month_8 : 0.18
4. month_9 : 0.19
5. season_2 - 0.15
6. season_4 - 0.0065

The above variable's respective p-values and beta coefficients can be referred from the # Step 5

Thus we can conclude that among the above 6 variables, people avoid weather_sit2(Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist) and weather_sit3 (Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds).

Atleast for month_8(August) and month_9(September) and for Season_2(summer), good number of people are expected to use the bike-sharing system.

OLS Regression Results

Dep. Variable:	cnt	**R-squared:**	0.841
Model:	OLS	**Adj. R-squared:**	0.838
Method:	Least Squares	**F-statistic:**	239.5
Date:	Tue, 14 Dec 2021	**Prob (F-statistic):**	7.50e-191
Time:	12:31:27	**Log-Likelihood:**	-4117.9
No. Observations:	510	**AIC:**	8260.
o scroll output; double click to hide	498	**BIC:**	8311.
Df Model:	11		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2108.3028	239.671	8.797	0.000	1637.412	2579.193
yr	1986.6570	70.594	28.142	0.000	1847.958	2125.356
holiday	-828.1994	222.664	-3.720	0.000	-1265.676	-390.723
temp	4690.8361	191.430	24.504	0.000	4314.726	5066.946
hum	-1546.7261	326.335	-4.740	0.000	-2187.888	-905.564
windspeed	-1665.6437	232.599	-7.161	0.000	-2122.639	-1208.648
season_2	904.9010	94.646	9.561	0.000	718.947	1090.855
season_4	1278.9715	92.699	13.797	0.000	1096.842	1461.102
month_8	481.0181	141.820	3.392	0.001	202.379	759.657
month_9	1063.9061	140.911	7.550	0.000	787.054	1340.759
weathersit_2	-459.1888	91.265	-5.031	0.000	-638.500	-279.878
weathersit_3	-2031.7881	228.828	-8.879	0.000	-2481.376	-1582.200