Assignment-based Subjective Questions

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

Answer:

Firstly, it's good to reference the equation of the MLR model –

$$cnt = 0.1807 + 0.2302*yr - 0.0785*holiday + 0.0479*weekday + 0.5947*temp - 0.1393*hum - 0.1934*windspeed - 0.2392*Light\_rain\_snow - 0.0542*Mist + 0.1367*Fall\_Winter + 0.0769*Summer$$

From the analysis, following is the summary of categorical variables which were used to train the model –

Categorical Variable	Dummy Variables (if any)	Description	Significance to ***cnt***
Year (yr)	*NA*	year (0: 2018, 1:2019)	**23.02%** ↑
Holiday (holiday)		weather day is a holiday or not	**7.8%** ↓
Weather Situation	Mist		**5.42%** ↓
	Light_rain_snow		**24.0%** ↓
Season	Summer		**7.7%** ↑
	Fall_Winter		**13.7%** ↑

Conclusive Points:

- The total number of bike rents shot up in 2019 by 23% as compared to 2018.
- The rentals tend to decrease by around 8% on holidays.
- The company saw less rentals when it was raining than it was less of a rain. Numbers tell the story; on a rainy day the sale tends to drop acutely by 24%.
- Rentals don't see much demand when it's hot outside. Whereas during fall/winter season, the sale tends to shoot up by 14%.

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

Answer:

It is always important to use as a smaller number of independent variables as possible. And when one identifies certain categorical variables in the dataset, one must carefully select them based on significance towards analysis. For instance, the "season" variable in the dataset "day.csv" (see Python notebook) has four states mapped numerical values 1, 2, 3, and 4. Its dictionary representation is as follows:

season = {1: "Spring", 2: "Summer", 3: "Fall", 4: "Winter"}

For such variables one must create dummy variables to be able quantify their contribution to the model. One can assign binary values to them i.e. 0 or 1.

E.g. To tell if it's a spring we may define the following conditions –

Spring	Summer	Fall	Winter
1	0	0	0

However, it's not required to assign "1" explicitly to tell if it's Spring. It is sufficient to assign 0's to Summer, Fall, and Winter to indicate that it's Spring. Thus, for that one can remove the first column "Spring" altogether, which can be done by providing the argument *drop_first=True* during dummy variable creation.

Indicating it's a Spring

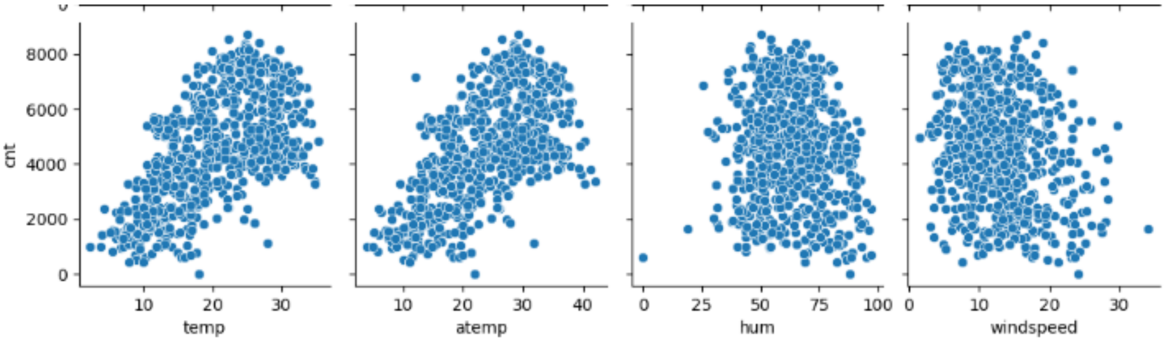
Summer	Fall	Winter
0	0	0

season_status = pd.get_dummies(df['season'], drop_first=True)

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

Answer:

From the pairplot (please see below), the dependent variable cnt has the highest correlation with **atemp**.



Also, from the correlation matrix, we find that corr(cnt, temp) and corr(cnt, atemp) are very close by viz. 62% and 63%.

However, as per the model, the RFE rejected atemp – hence it can be said that cnt is highly correlated with temp.

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

Answer:

Following are the tests that must be carried out to validate the assumption of Linear Regression:

a. Multicollinearity test

Computing Variance Inflation Factor (VIF) of each independent variable used for model. VIF should be < 5. Following is the table of VIF –

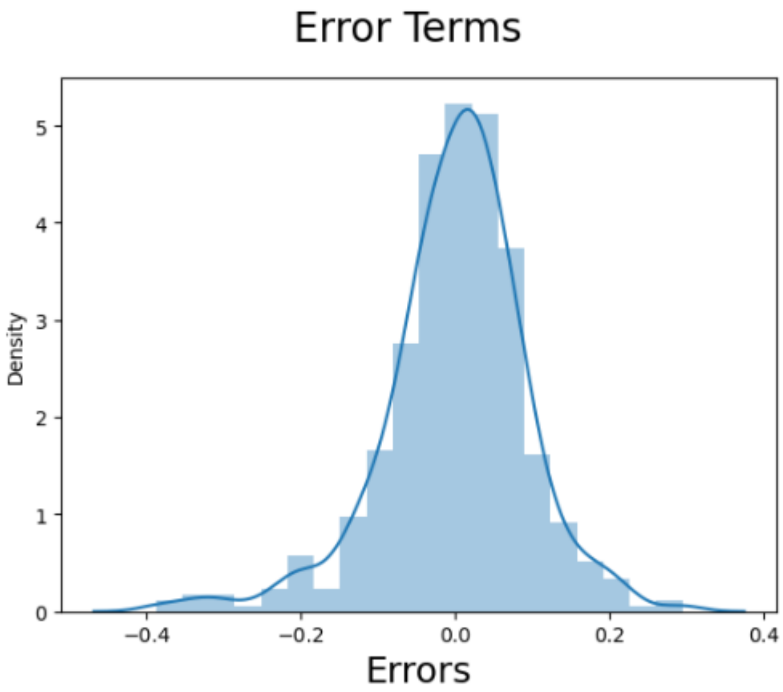
Features	**VIF**	**Features**	**VIF**
const	49.01	windspeed	1.18
hum	1.87	Summer	1.14
Mist	1.56	yr	1.03
Fall_Winter	1.25	weekday	1.03
Light_Rain_Snow	1.24	holiday	1.02
temp	1.23		

It must be noted that all the VIF values in the table above are less than 2.0 except const which is expected since it's a constant.

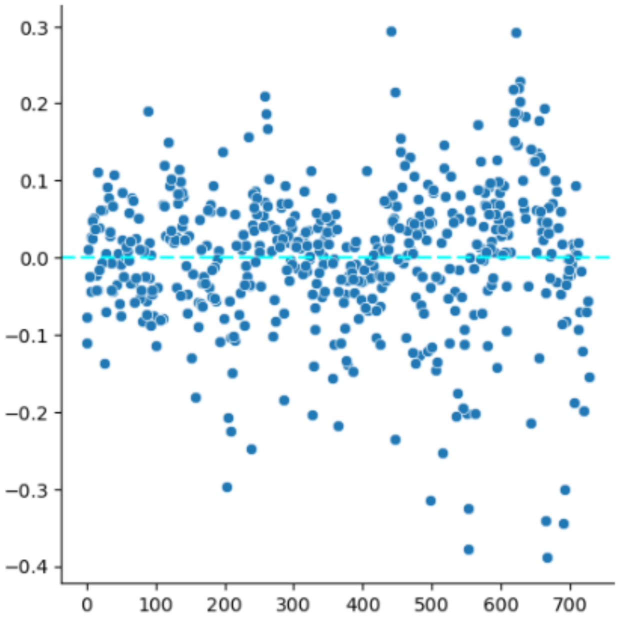
b. Error terms

e = y_train - y_train_cnt; where y_train is the actual training data and y_train_cnt is the data predicted by the model.

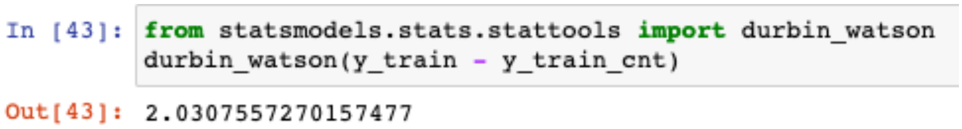
i. Error terms must follow a normal distribution centered at zero. Following distribution plot confirms the same –



ii. Constant variance condition – the error term must have, overall, the standard deviations within a narrow band. Following is the scatter plot of the error term confirming the same –



iii. Durbin-Watson-Test – the test statistic is equal to 2(1-r) where r is the sample autocorrelation of the residuals. Generally, it lies between 0 and 4. Closer to 0 means more positive serial correlation, closer to 4 means more negative serial correlation. Normal range is 1.5 to 2.5.
Performing Durbin-Watson-Test over the error terms of the model predictions, we get the test statistic equal to 2.03 → normal range → no autocorrelation.



```
In [43]: from statsmodels.stats.stattools import durbin_watson
         durbin_watson(y_train - y_train_cnt)

Out[43]: 2.0307557270157477
```

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

Answer:

Following is the trained model equation –

*cnt = 0.1807 + 0.2302*yr - 0.0785*holiday + 0.0479*weekday + 0.5947*temp - 0.1393*hum - 0.1934*windspeed - 0.2392*Light_rain_snow - 0.0542*Mist + 0.1367*Fall_Winter + 0.0769*Summer*

Top 3 features that contribute significantly towards explaining the demand of the shared bikes are –

- Temperature – With other variables held constant, demand of the shared bikes increases by a factor of 0.5947 (almost 60%) with temperature.
- Year – With other variables held constant, demand of the shared bikes increases by a factor of 0.2303 (almost 23%). This means the demand increased specifically in the year 2019.

- Humidity – With other variables held constant, demand of the shared bikes increases by a factor of 0.1392 (almost 14%) with humidity.

General Subjective Questions

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

Answer:

Linear regression is a type of supervised learning which involves predictive analysis over a given dataset of continuous numerical variables. The dataset in general may or may not contain categorical data as well which must be converted to dummy variables (mapped to 0 or 1) for adding up a value in regards to the quantifying their contribution to the analysis.

Following is a generic representation of a multilinear regressive model –

$$y = \beta_0 + \sum_{i=1}^{N} \beta_i X_i \ ; \ \beta_i \ are \ the \ coefficients \ of \ features \ X_i$$

A linear regression model statistically tries to best fit a linear curve.

In general, linear regression have the following assumptions over the dataset over which a regressive model gets trained upon –

- Linear relationship between the independent variables (feature list) and the target (dependent variable) variable.
- Minimal multicollinearity – predictor variables (feature list) should be independent of each other – this can be checked by evaluating correlation and VIF.
- Error terms should be normally distributed, have constant variance, and should have no autocorrelation – perform Durbin-Watson testing for instance.

Following are the steps to perform linear regression –

- Reading and understanding the data – data preparation involving data cleaning (remove NULLs, redundant variables, manipulation), creating dummy variables (convert categorical variables to dummy variables).
- Training the model - this is the most important step. Following are some of the steps on training the model –
  a. With the available feature list (dependent variables), split the data into training and test set. Usual practice is to use 70% - 30% split.
  b. Perform scaling over the numeric features. Some of the options are – MinMax scaling and Standardization. This allows the model to train without biasing a specific variable since already removes outlier values by normalizing the variables to a certain range.

c. One needs to select appropriate variables for training i.e. significant enough for the needed application. One method is to use Recursive Feature Elimination (RFE) that eases the process of eliminating redundant variables. It helps in this case by assigning ranking and support indicators.
d. Train the model. One can use SKlearn's fit() method that essentially utilizes OLS (ordinary least square) technique to optimize the coefficients of the dependent variables and return statistical parameters like R-squared, adjusted R-squared, p-value, AIC, BIC, etc. Out of these the most important ones are R-squared and p-value. p-value of the variables must be < 0.05 as a thumb of rule and R-squared as close to 1 as possible. Apart from these, one must also look for Variance Inflation Factor (VIF) to check for the multicollinearity. VIF should be < 5.
e. Repeat a through d until p-values and VIF of all the independent variables are optimal.

- Residual Analysis – one must confirm that the residual terms (error terms i.e. y_predicted – y_train) follow a normal distribution centred at zero.
- Predicting and evaluating on the test set.

2. Explain the Anscombe's quartet in detail.

Answer:

Anscombe's quartet is a scatter plot visualization of 4 datasets placed in 4 subplot boxes. These 4 datasets are mostly seen to have similar distribution but follow a completely different trend which can be exposed by scatter plots.

It can save a lot of our time building the models by providing a pre-model-development analysis on crucial datasets that have ability to fool the regression models in hand.

It can help us identify outliers in the dataset which the model cannot remove it for us. These can be removed early on during the data preparation step.

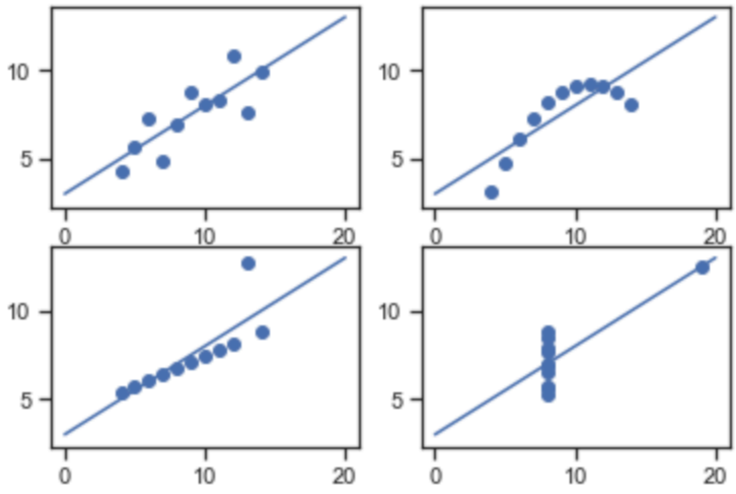


Image courtesy: https://medium.com/analytics-vidhya/anscombes-quartet-an-importance-of-data-visualization-856b3d1bd403

In the above figure, the distribution of the 4 datasets might be similar but the scatter plot tells us that they are not as linear as the linear regression would assume. On the dataset 1 and 4 a regression analysis can be performed. However, some portion of 2 might follow a linear behaviour but a model cannot be fit properly. 4 is not at all suitable for a linear regression. Moreover, if at all suitable, the outliers now can be identified and removed beforehand.

3. What is Pearson's R?

Answer:

Pearson's R is the most general way of computing correlation (r) between two independent variables.

Theoretically, the formula for computing r is –

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

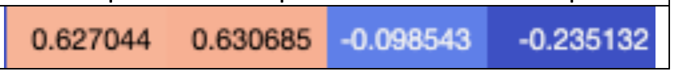
Where x and y are the independent variables and n is the number of samples.

r ranges from -1 to 1; -1 being perfect negative correlation and +1 being perfect positive correlation, 0 means no correlation.

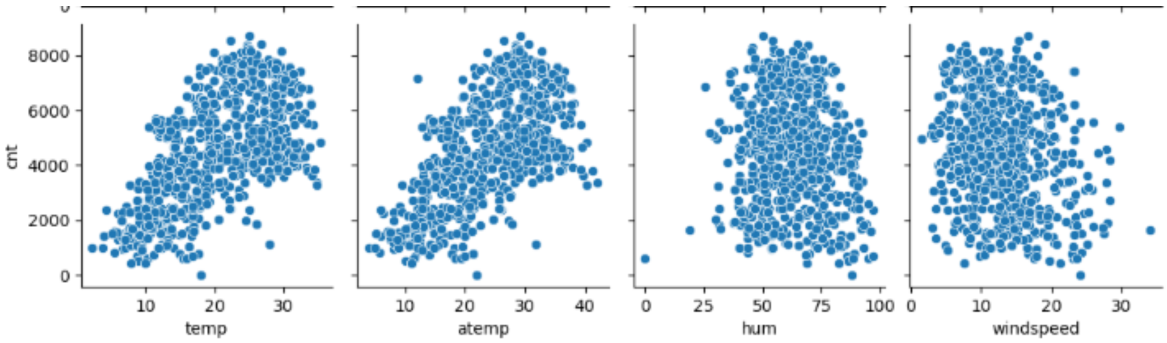
Correlation is one of the preliminary ways of checking correlation between independent and dependent/other independent variables.

Check correlation between dependent variables
corr = df.corr()
corr.style.background_gradient(cmap='coolwarm')

Correlation	temp	atemp	hum	windspeed
cnt	0.627044	0.630685	-0.098543	-0.235132



In the following plots, 1st and 2nd have a positive correlation, hum shows a negative correlation and the windspeed has a poor correlation of -0.23



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

   Answer:

   Scaling is a way by which one can normalize the feature values so that model training can be carried out in an unbiased manner. For instance, consider the model training in the Bike assignment, let's take the following variables into account – "temp" and "mnth". Temperature values generally vary up to 33-40 whereas the month max value is 12. Thus, if the model is trained without scaling, it will be more biased towards temperature in comparison to the month. It must be noted that month can also be an important feature which one must consider. The model must first normalize the data in such a way that the bias is no longer present.

   There are two most important types of scaling used in linear regression –
   a. MinMax

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

   This type of scaling normalizes the feature values between 0 and 1.

   b. Standardization

$$x = \frac{x - mean(x)}{std(x)}$$

   With this scaling, the overall mean and standard deviation of the feature values become 0 and 1 respectively.

   One must use MinMax scaling specifically removing outliers. Values however big they might be, are eventually mapped to the range [0, 1] thereby removing the notion of outliers.

   In practice, scaling is applied only to the numeric values since they are continuous in property. When applied to categorical variables i.e., converted to dummy, will have no impact since mapping will be simply either 0 or 1.

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

Answer:

VIF is calculated as –

$$VIF = \frac{1}{1 - {R^2}_i}$$

When R-squared value becomes 1 i.e., in case of perfect correlation, VIF value is infinite. Perfect correlation is a result of the regression line passing through all the data points. On the other hand, such condition might be rare, usual suspects might be some variables in consideration in a way directly sum up to the target variable. E.g., in the “day.csv” dataset, “causal” and “registered” add up to “cnt”. Thus, if these variables are included in the analysis, the resulting R-squared value will be 1. These tend to cause perfect collinearity.

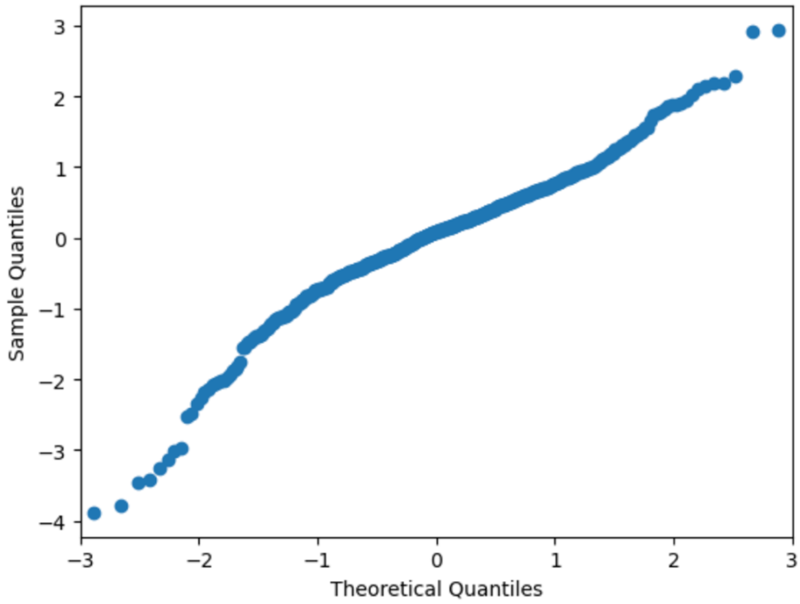
6. *What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression.*

Answer:

Q-Q plot or “Quantile-Quantile” plot is used to check if a given dataset follows a particular distribution. Generally, we check for normal distribution in linear regression models during the residual analysis.

For the given dataset “day.csv”, the q-q plot for the error terms between y_pred and y_train is shown below –

```
fig = sm.qqplot(10*(y_train - y_train_cnt))
plt.show()
```



A linear plot, almost following a 45 degrees line i.e., y = mx with m = 1, strongly indicates that the error terms follow a normal distribution, thus confirming good fit of the model with linearity assumptions holding good.

On the x-axis are the theoretical quantiles whereas y-axis indicate the actual data quantiles. It basically is a 1-1 mapping between theoretical normal distribution and distribution of the sample quantiles i.e., error terms in our case. It can also give us a measure of skewness i.e. deviation from the normal distribution.

If the Q-Q plot doesn't show a linearity, it can be said that the regression model is not tuned properly. The process must be repeated considering significant features.