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? (3 marks)**

   The dependent variable shows a general trend on

   - Year: The number of users has increased year-over-year.
   - Season: The number of users is lower during Spring season and highest during fall due to 72.3% contribution of clear sky.
   - Weathersit: As expected, low number of users seen when we have light rains and a greater number of users when there is clear weather.
   - The trend in month is similar to the trend in season with months of Dec, Jan, Feb having low number of users as these are the months of Spring.

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

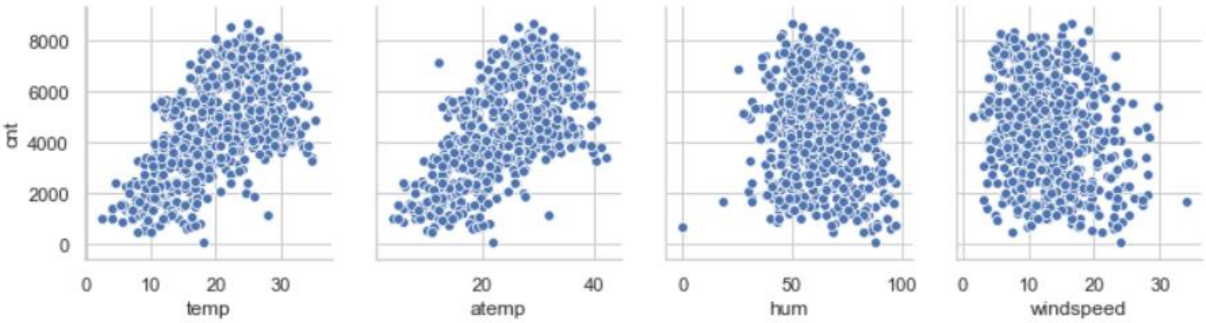
   When a dummy variable is created, it creates number of columns equal to the number of values in the original column, i.e., if we create dummy columns for the column season then we will end up having 4 columns with each column representing a 0 or 1.

Spring	Summer	Fall	Winter
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

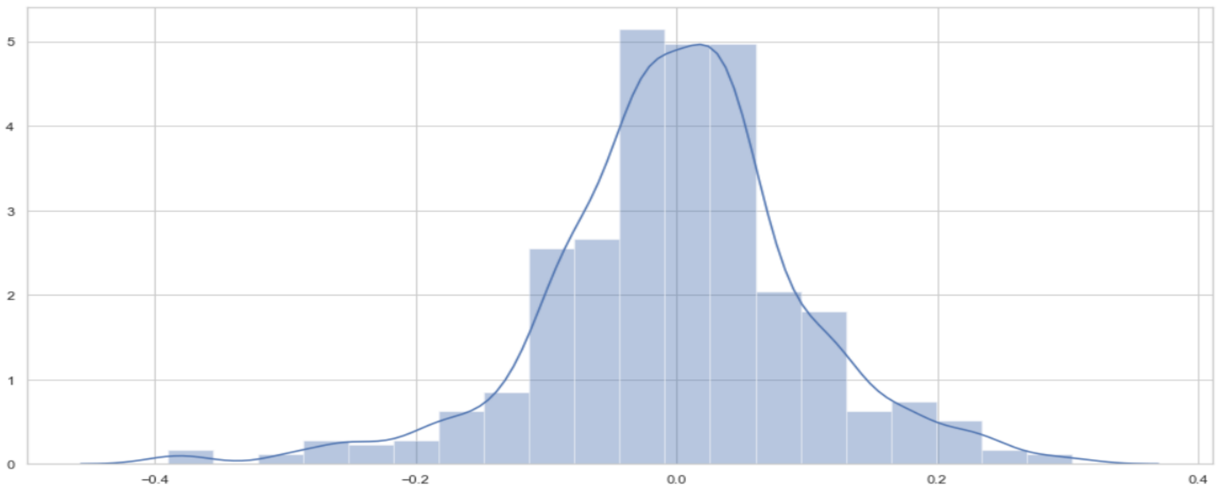
   The above table represent the values that the columns would have. In this, we can safely drop a single column since a 0 in the other 3 columns represent a 1 in the 4th column. Adding drop_first-=True while creating dummy columns does exactly this.

3. **Looking at the pair-plot among the numerical variables, which one has the highest correlation with the target variable? (1 mark)**

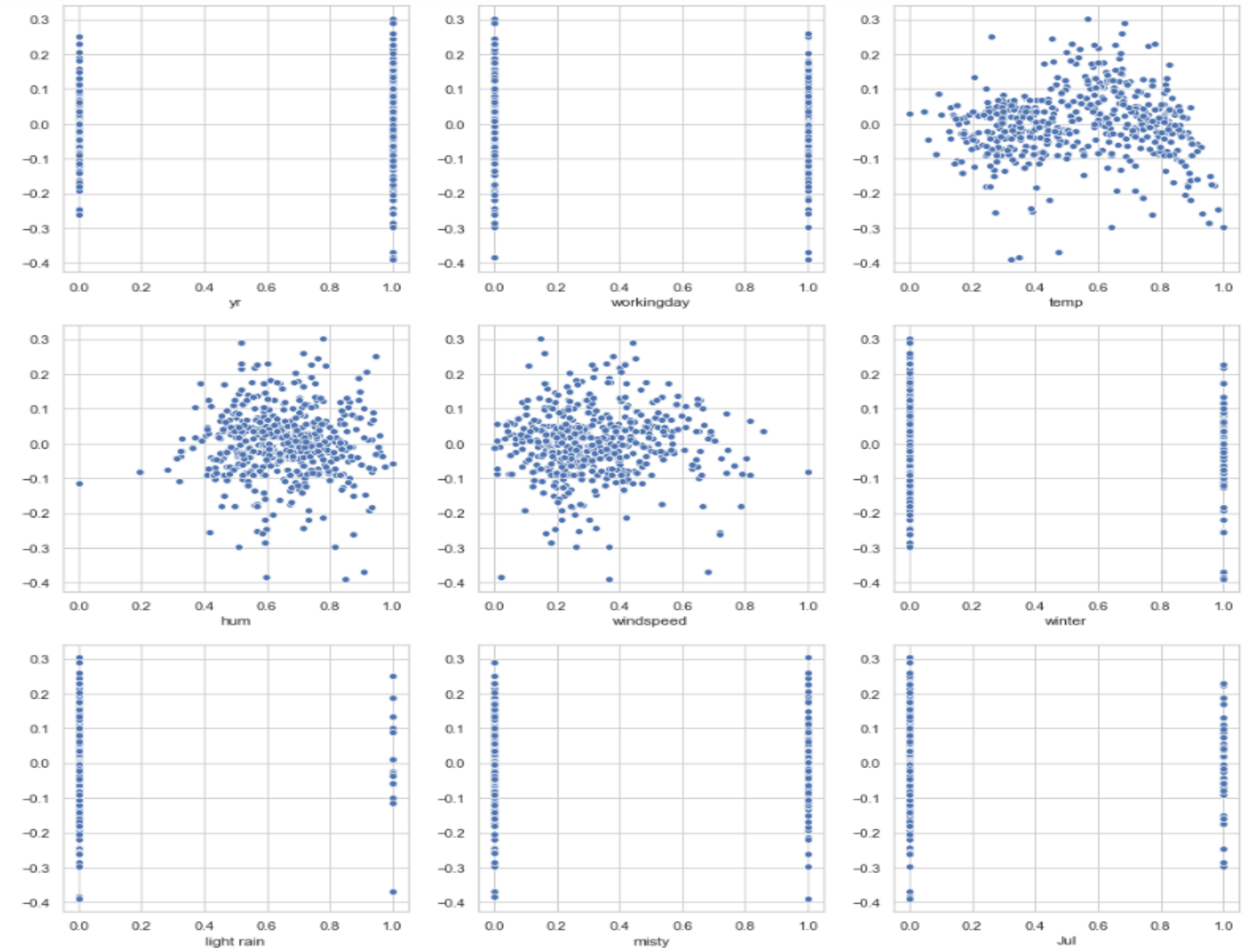
   Looking at the pair plot of numerical variables, we see that the spread of temp and atemp are distributed linearly whereas humidity and windspeed look clustered. Since temp and atemp are highly correlated, we can drop one of the 2 for our modelling.



4. **How did you validate the assumptions of Linear Regression after building the model on the training set? (3 marks)**
After the model was built, Residual analysis of the model was performed. The distribution plot of the residuals appears to be normally distributed with no sharp peak values and centered towards 0.



The residuals were plotted for each independent variable and we do not see any strong pattern for any of the variables, thus we can say that they are homoscedastic in nature.



5. **Based on the final model, which are the top 3 features contributing significantly towards explaining the demand of the shared bikes? (2 marks)**

Based on the model, temp, yr and light rain are the 3 main features explaining the demand of shared bikes.

$$y = 0.6578 * temp + 0.2266 * yr + 0.1075 * winter + 0.0231 * workingday - 0.0465 * misty - 0.1016 * Jul - 0.1695 * hum - 0.1795 * windspeed - 0.2274 * light\ rain + 0.2011$$

Based on the equation of the hyper plane:

- 1 point **increase** in temp while keeping all the other factors constant, causes a ~0.66 points **increase** in the bike demand.
- 1 point **increase** in yr while keeping all the other factors constant, causes a ~0.23 points **increase** in the bike demand.
- 1 point **decrease** in light rain while keeping all the other factors constant, causes a ~0.23 points **increase** in the bike demand.

General Subjective Questions

1. **Explain the linear regression algorithm in detail. (4 marks)**

Linear Regression is a machine learning algorithm based on **supervised learning** and is used to predict a dependent variable (y) based on a set of given independent variable (x).
A simple linear regression model draws a best fit straight line though the data points and this is represented using the formula:

$$y = mx + c$$

Where,
y = dependent variable (target variable)
x = independent variable (features)
m = slope of the line (co-efficient of x)
c = Intercept (constant)

A multiple regression model draws a hyper plane through the data points and is represented by

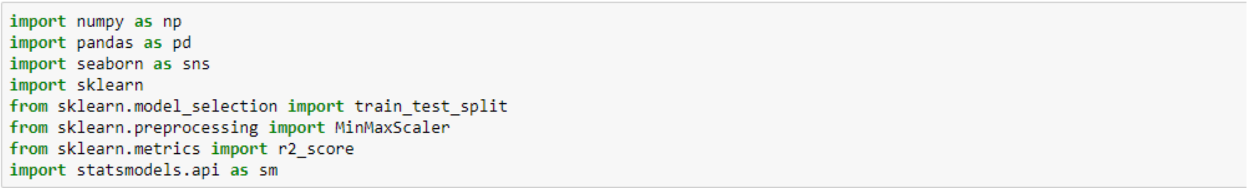
$$y = m_1x_1 + m_2x_2 + .. + m_nx_n + c$$

where, x_1, x_2, .. x_n are the independent variable with m_1, m_2, .. m_n being their respective slopes.

In Linear Regression, **Mean Squared Error (MSE)** cost function is used, which is the average of squared error that occurred between the predicted values and actual values. This cost function helps to figure out the best possible values of co-efficient and slope, which provides the best fit line.

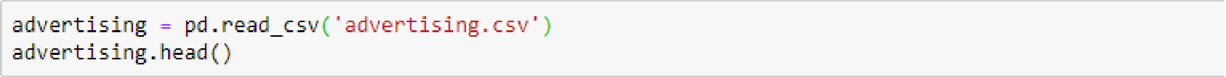
To calculate linear regression expressing using Python, we need to do the following:

1. Declare all the libraries

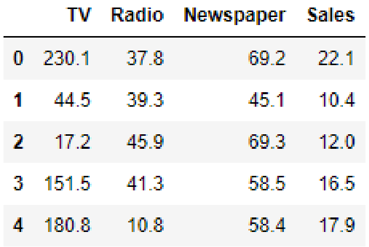


```
import numpy as np
import pandas as pd
import seaborn as sns
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import r2_score
import statsmodels.api as sm
```

2. Read the data

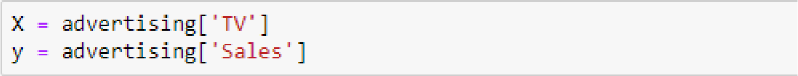


```
advertising = pd.read_csv('advertising.csv')
advertising.head()
```



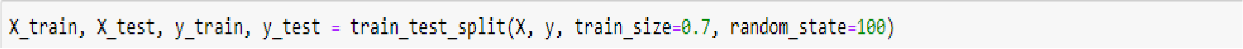
	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

3. Here we are considering TV as our independent variable and would be predicting sales which is our target variable



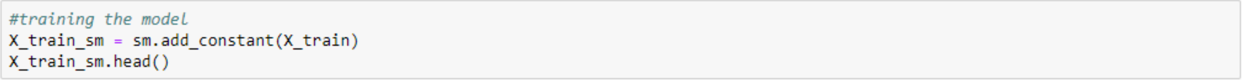
```
X = advertising['TV']
y = advertising['Sales']
```

4. Split the data in train and test using the train_test_split method. We usually split 70:30 ratio where 70% would be our train data and the tests would be performed on the remaining 30% data.

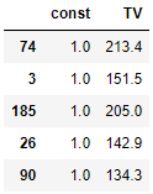


```
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.7, random_state=100)
```

5. We would be using stats model for our example. When using the statsmodel library, we need to add constant to the dataframe since this is not added by default.

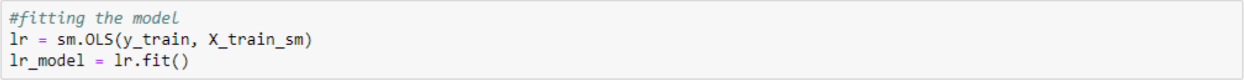


```
#training the model
X_train_sm = sm.add_constant(X_train)
X_train_sm.head()
```



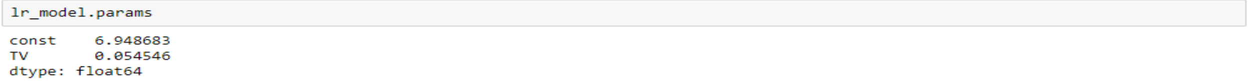
	const	TV
74	1.0	213.4
3	1.0	151.5
185	1.0	205.0
26	1.0	142.9
90	1.0	134.3

6. The data then needs to be fitted in the model and this is done by using the fit method in the OLS class.



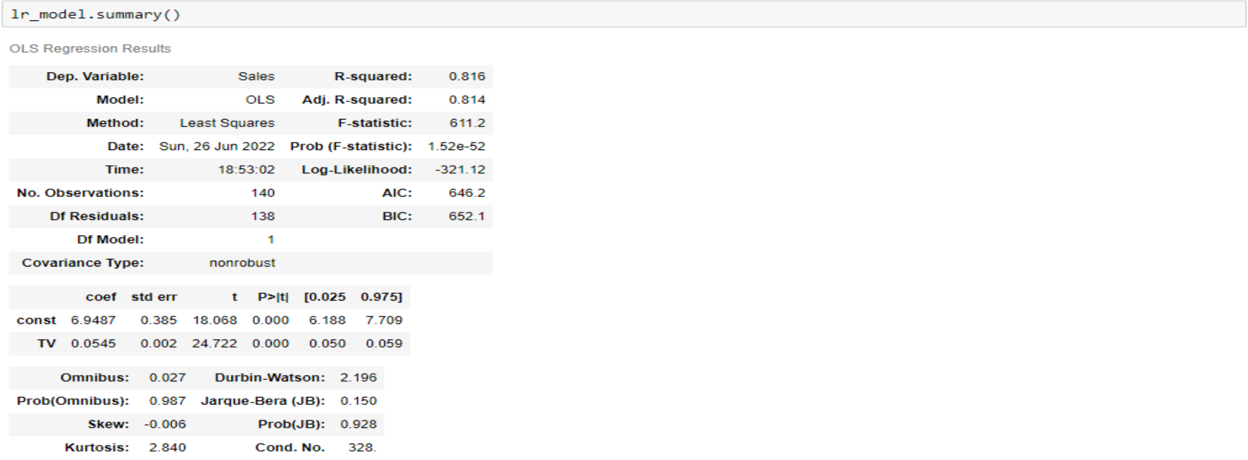
```
#fitting the model
lr = sm.OLS(y_train, X_train_sm)
lr_model = lr.fit()
```

7. Now that our model has been fitted in, we can check some summary metrics in order to see how well does this independent variable predict the target variable.



```
lr_model.params
```

```
const    6.948683
TV       0.054546
dtype: float64
```



```
lr_model.summary()
```

OLS Regression Results

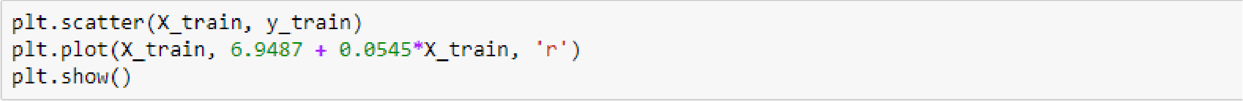
Dep. Variable:	Sales	R-squared:	0.816
Model:	OLS	Adj. R-squared:	0.814
Method:	Least Squares	F-statistic:	611.2
Date:	Sun, 26 Jun 2022	Prob (F-statistic):	1.52e-52
Time:	18:53:02	Log-Likelihood:	-321.12
No. Observations:	140	AIC:	646.2
Df Residuals:	138	BIC:	652.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	6.9487	0.385	18.068	0.000	6.188	7.709
TV	0.0545	0.002	24.722	0.000	0.050	0.059

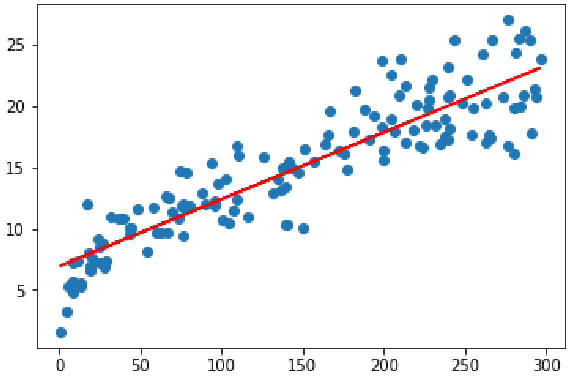
Omnibus:	0.027	Durbin-Watson:	2.196
Prob(Omnibus):	0.987	Jarque-Bera (JB):	0.150
Skew:	-0.006	Prob(JB):	0.928
Kurtosis:	2.840	Cond. No.	328.

Here we see that we have an R-squared of 81.6 with a very low p-value. This means that this model can predict 81.6% of the total variance in sales.

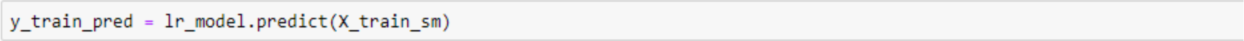
8. Plotting a line using the equation y=mx+c along with all the data points for sales and TV, we get the below chart



```
plt.scatter(X_train, y_train)
plt.plot(X_train, 6.9487 + 0.0545*X_train, 'r')
plt.show()
```

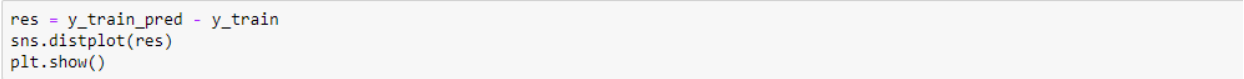


9. Predicting the train data

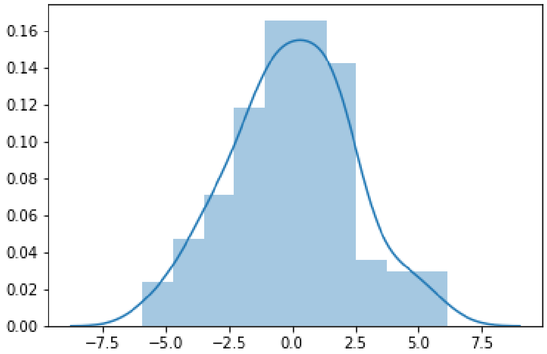


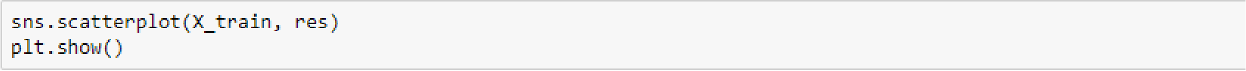
```
y_train_pred = lr_model.predict(X_train_sm)
```

10. Analyzing the residuals (cost functions), we see that the distribution plot looks normally distributed with the no sharp peaks and centered towards 0. The scatter plot for the residual is used to check if there are any patterns for the residual. In this case we do not see any strong patter and the data looks to me homoscedastic in nature.

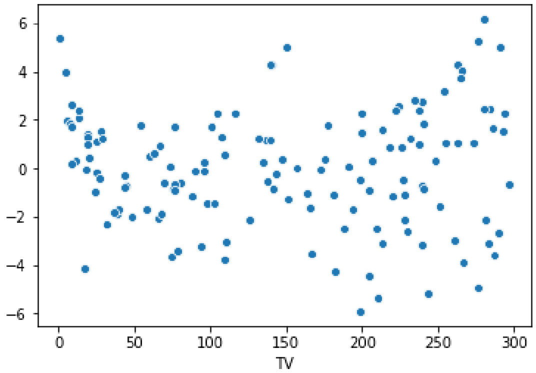


```
res = y_train_pred - y_train
sns.distplot(res)
plt.show()
```

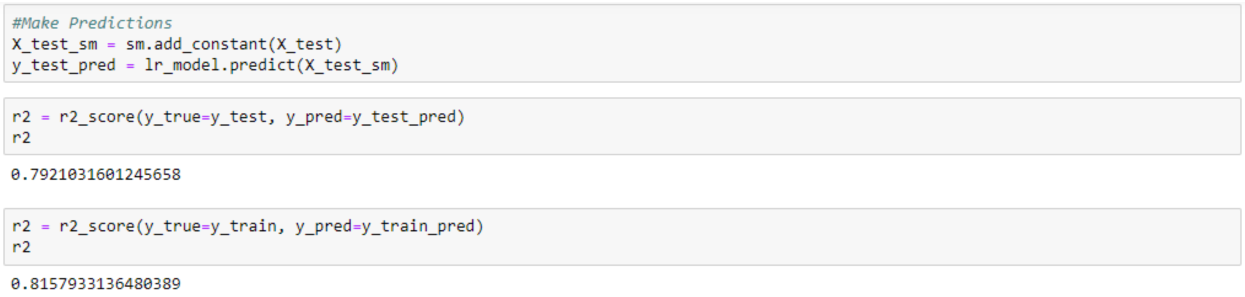




```
sns.scatterplot(X_train, res)
plt.show()
```



11. Finally, we predict the test data and check the R-squared on the test data. Since the R-squared on the test data (79.2) is within 5% of the R-squared (81.5) of the train data, we can say that the model works correctly on the test data as well.



```
#Make Predictions
X_test_sm = sm.add_constant(X_test)
y_test_pred = lr_model.predict(X_test_sm)
```

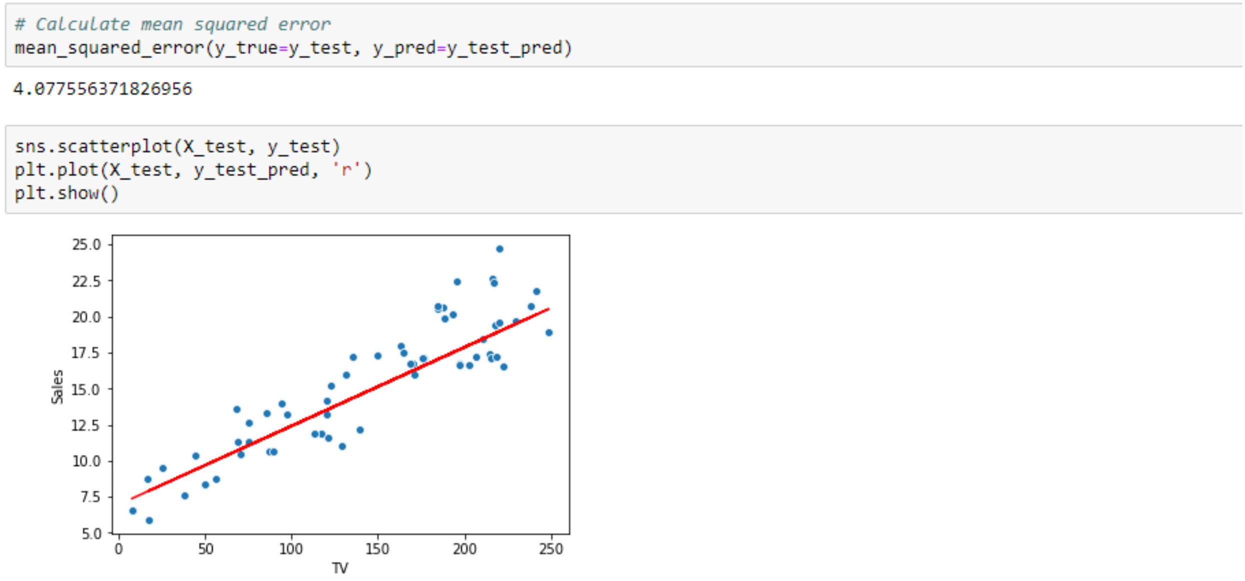
```
r2 = r2_score(y_true=y_test, y_pred=y_test_pred)
r2
```

```
0.7921031601245658
```

```
r2 = r2_score(y_true=y_train, y_pred=y_train_pred)
r2
```

```
0.8157933136480389
```

12. We then plot the predicted value with the actual value to get an understanding of what our predictions look like



```
# Calculate mean squared error
mean_squared_error(y_true=y_test, y_pred=y_test_pred)
```

```
4.077556371826956
```

```
sns.scatterplot(X_test, y_test)
plt.plot(X_test, y_test_pred, 'r')
plt.show()
```

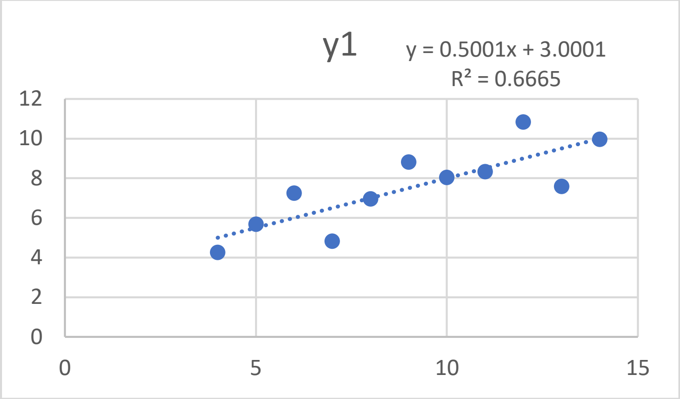
2. **Explain the Anscombe's quartet in detail. (3 marks)**

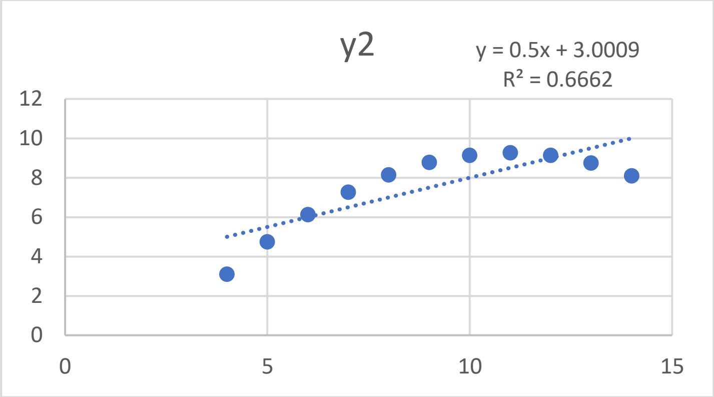
Anscombe's quartet comprises four datasets that have nearly identical simple statistical properties, yet appear very different when graphed. They were constructed by the statistician Francis Anscombe to demonstrate both the importance of graphing data before analyzing it and the effect of outliers on statistical properties.

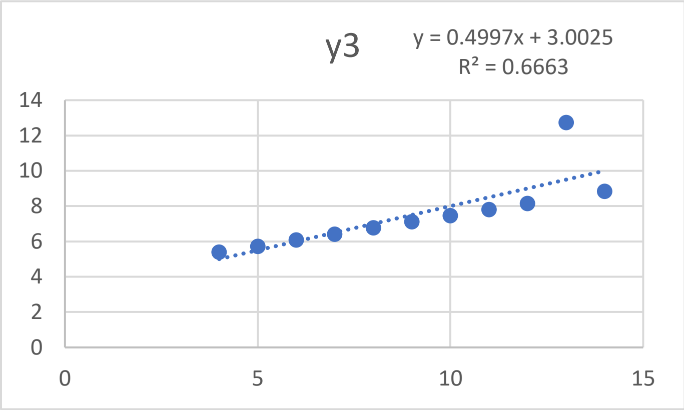
By simply looking at the statistics of the columns, we sometimes might conclude that the data is identical in nature since the Mean, standard deviation and Correlation are very similar if not the exact same.

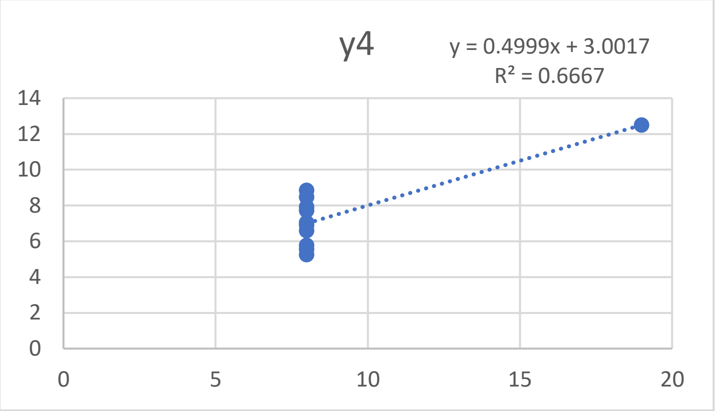
	x1	y1	x2	y2	x3	y3	x4	y4
	10	8.04	10	9.14	10	7.46	8	6.58
	8	6.95	8	8.14	8	6.77	8	5.76
	13	7.58	13	8.74	13	12.74	8	7.71
	9	8.81	9	8.77	9	7.11	8	8.84
	11	8.33	11	9.26	11	7.81	8	8.47
	14	9.96	14	8.1	14	8.84	8	7.04
	6	7.24	6	6.13	6	6.08	8	5.25
	4	4.26	4	3.1	4	5.39	19	12.5
	12	10.84	12	9.13	12	8.15	8	5.56
	7	4.82	7	7.26	7	6.42	8	7.91
	5	5.68	5	4.74	5	5.73	8	6.89
Mean	9	7.50	9	7.50	9	7.50	9	7.50
Standard Deviation	3.32	2.03	3.32	2.03	3.32	2.03	3.32	2.03
Correlation	0.816		0.816		0.816		0.817	

However, if we plot these points on a scatter plot in the excel, we get the below









Here, looking at the trends we see that the numbers when plotted have different trends and they are not the same.

Explanation of this output:

- Y1 we see that there seems to be a linear relationship between x and y.
- Y2 there is a non-linear relationship between x and y.
- Y3 has a perfect linear relationship for all the data points except one which seems to be an outlier
- Y4 has 1 extreme outlier which produce a high correlation coefficient.

Conclusion: This illustrates the importance of looking at the data graphically before starting to analyze than solely relying on statistic properties.

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

The **Pearson correlation coefficient (*r*)** is the most common way of measuring a linear correlation. It is a number between –1 and 1 that measures the strength and direction of the relationship between two variables.
The below table explains how to interpret the different values of R.

Pearson correlation coefficient (*r*)	Correlation type	Interpretation
Between 0 and 1	Positive correlation	When one variable changes, the other variable changes in the **same direction**.
0	No correlation	There is **no relationship** between the variables.
Between 0 and -1	Negative correlation	When one variable changes, the other variable changes in the **opposite direction**.

Pearson's R can be calculated using the below formula

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

In order to demonstrate the formula, I have used x and y values, calculated their respective squares and product of x and y. Later the sum of all columns has been calculated. Using these values in the above formula we can calculate the Pearson's R value.

	x	y	x^2	y^2	x*y
	3.63	53.1	13.1769	2819.61	192.753
	3.02	49.7	9.1204	2470.09	150.094
	3.82	48.4	14.5924	2342.56	184.888
	3.42	54.2	11.6964	2937.64	185.364
	3.59	54.9	12.8881	3014.01	197.091
	2.87	43.7	8.2369	1909.69	125.419
	3.03	47.2	9.1809	2227.84	143.016
	3.46	45.2	11.9716	2043.04	156.392
	3.36	54.4	11.2896	2959.36	182.784
	3.3	50.4	10.89	2540.16	166.32
Sum	33.5	501.2	113.0432	25264	1684.121

Pearson's R	0.47017723

4. **What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling? (3 marks)**

Scaling is a step of data Pre-Processing which is applied to independent variables to normalize the data within a particular range which 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 resulting in 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.

Normalized scaling: it is a technique in which the values are rescaled so that they end up between 0 and 1. It is also know and min-max scaling. This technique is used when the distribution of that data does not follow Gaussian distribution. This can be useful in algorithms that do not assume any distribution of the data like K-Nearest Neighbors and Neural Networks.

$$X = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Where X_{max} and X_{min} are the maximum and minimum values of the features.

Standardized scaling: It is technique where the values are centered 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. Unlike normalization, standardization does not have a bounding range. So, even if you have outliers in your data, they will not be affected by standardization.

$$X = \frac{X - \mu}{\sigma}$$

Where μ is the mean and σ is the standard deviation of the feature values.

5. You might have observed that sometimes the value of VIF is infinite. Why does this happen? (3 marks)

Variance inflation factor (VIF) is a measure of the amount of multicollinearity in a set of multiple regression variables. VIF calculate how well one independent variable is explained by all the other variables (except target variable) combined.

VIF is given by:

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

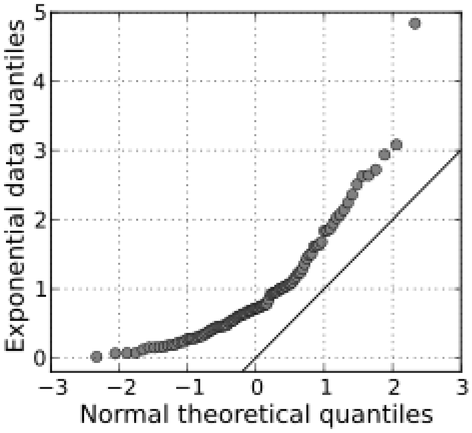
Where 'i' refers to the i^{th} variable which is being represented as a linear combination of rest of the independent variables.

The value is infinite if the denominator of the equation becomes 0, i.e., R-squared become 1. This happens when the variable is highly correlated with the other variables. Since these variables cause a multicollinearity effect, we drop such variables.

5. **What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression. (3 marks)**

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° 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. A Q-Q plot showing the 45° reference line:



If the two distributions being compared are similar, the points in the Q–Q plot will approximately lie on the line y = x. If the distributions are linearly related, the points in the Q–Q plot will approximately lie on a line, but not necessarily on the line y = x. Q–Q plots can also be used as a graphical means of estimating parameters in a location-scale family of distributions.

A Q–Q plot is used to compare the shapes of distributions, providing a graphical view of how properties such as location, scale, and skewness are similar or different in the two distributions.