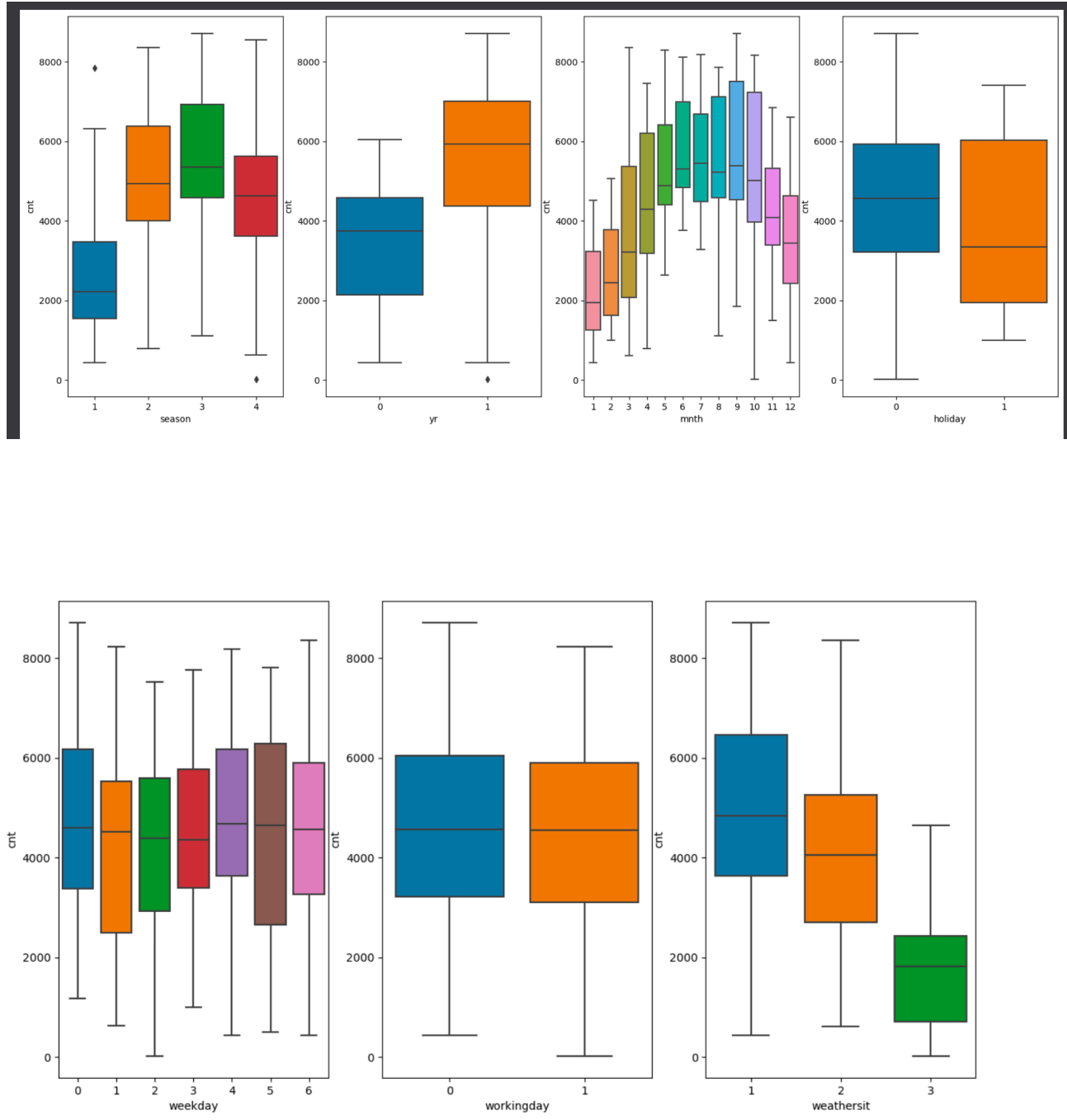


Assignment-based Subjective Questions

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



As we can notice above , we have identified season , yr , month , holiday , weekday , working day , weatherise as categorical data .

We could make below observations

1.For yr : from year to year the demand for the bikes has increased.

2.From holiday : non-holiday the demand for bikes is more this we could tell from the 25% quartile region for non holiday day is more compared to the holiday day.

3.From seasons : among 'seasons' , fall has the highest demand , and spring has the least demand for the bikes

4.From weathersit : we could infer that on a rainy day and hail storm the demand for bikes is very less , compared to the cloudy days or partial cloudy days

The final equation derived also supports our narrative

```
cnt = 0.2315 * yr - 0.0629 * holiday + 0.0252 * weekday +  
0.5001 * atemp - 0.2388 * slight_to_heavy_rain_or_snow - 0.1085 *  
spring + 0.0613 * winter
```

In the above equation we could see that slight_to_heavy_rain_or_snow , spring , holiday has negative slope . Yr , weekday , winter has positive slope.

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

When we extract dummy variables , it would generate dummy variables for all the categories , for us to represent a variable with n categories we would need n-1 dummy variables , hence we drop the first dummy value as it could be represented by the n-1 dummy variables

As we can see above there for a variable with 4 categories it created 4 dummy variables , as we need only 3 of these variables we are dropping the first dummy variable

```
[245] seasons = pd.get_dummies(daily_data['seasons'])
✓ 0.2s Python
```

```
[246] seasons
✓ 0.3s Python
```

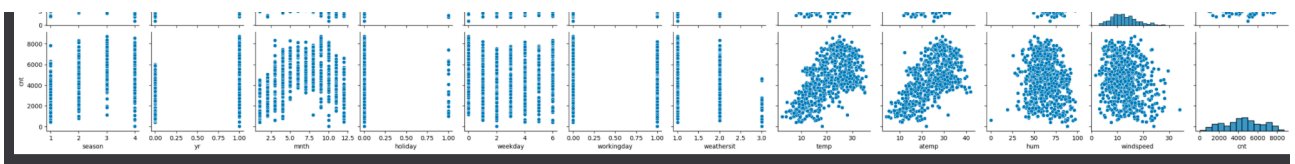
	fall	spring	summer	winter
0	0	1	0	0
1	0	1	0	0
2	0	1	0	0
3	0	1	0	0
4	0	1	0	0
...
725	0	1	0	0
726	0	1	0	0
727	0	1	0	0
728	0	1	0	0
729	0	1	0	0

730 rows x 4 columns

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

From the pair plot we could see that temp and temp have high correlation with our target variable cnt

Below image cements our understanding as well



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

We validate the assumptions on the Linear Regression by summarising the model which would provide us the R-squared and Adjusted R-squared values , if we have these values almost equal we could see that the model is good . On top of this we also see the P-value for the features and could see which features have high P-values and we drop those features , once the feature is dropped we will build the model and take the stats again . Then we take the VIF for the features and to understand the correlation with other features and drop the ones with high VIF one after the other and rebuild the module to until we attain the desired VIF values between features , and better R-squared and adj-R-squared , and P-values of features.

Sample outputs:

Model summary

```
x_extra_test = x_extra_test.drop('workingday',axis=1)
linear_regression_model7 = build_model(X_features=x_extra_test , y=y_train_daily)
linear_regression_model7.summary()
```

✓ 0.8s Python

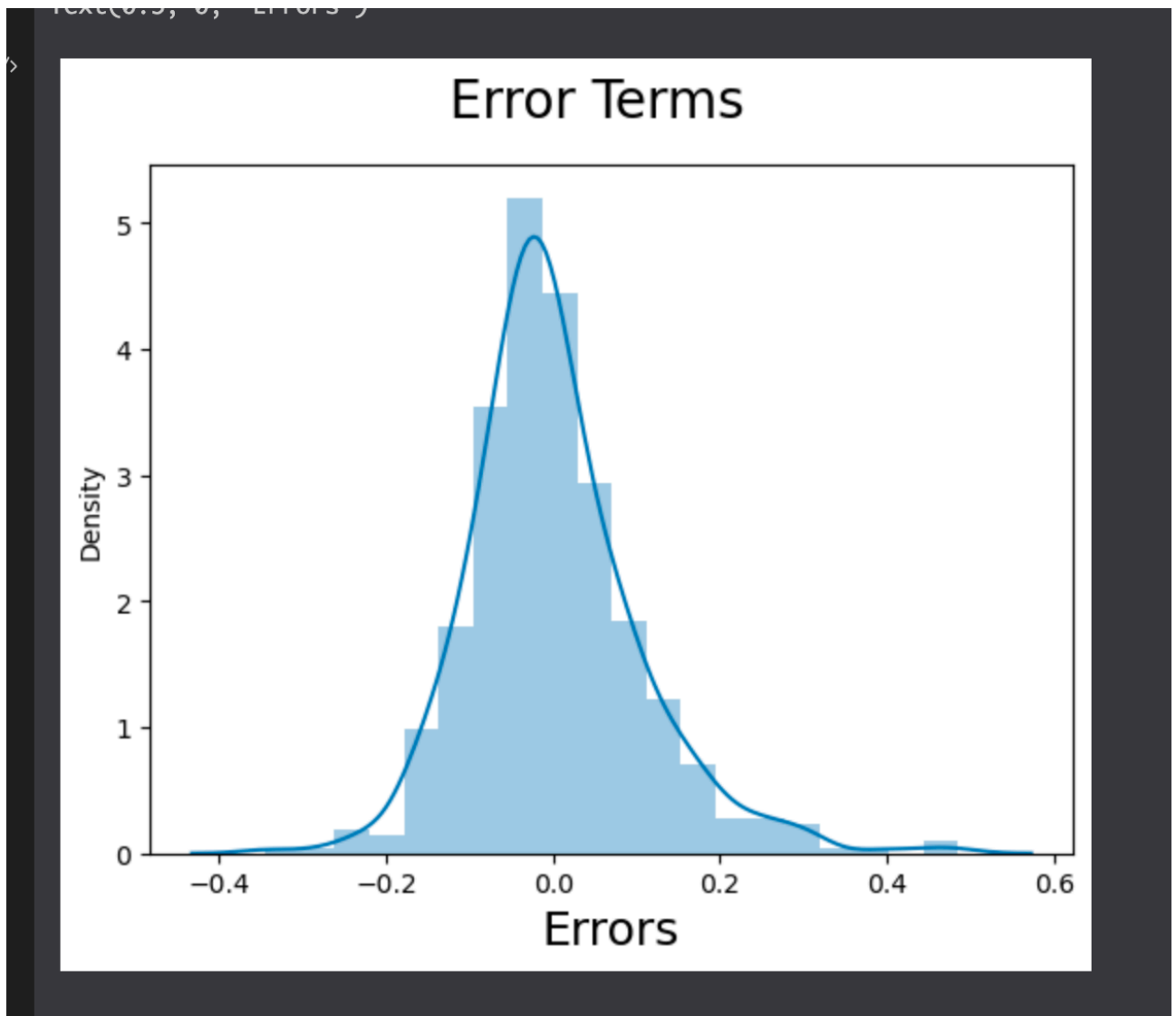
OLS Regression Results							
Dep. Variable:	cnt			R-squared:		0.793	
Model:	OLS			Adj. R-squared:		0.791	
Method:	Least Squares			F-statistic:		275.4	
Date:	Tue, 02 May 2023			Prob (F-statistic):		2.10e-167	
Time:	00:15:28			Log-Likelihood:		441.05	
No. Observations:	510			AIC:		-866.1	
Df Residuals:	502			BIC:		-832.2	
Df Model:	7						
Covariance Type:	nonrobust						
		coef	std err	t	P> t	[0.025	0.975]
const		0.1515	0.024	6.383	0.000	0.105	0.198
yr		0.2315	0.009	25.229	0.000	0.214	0.250
holiday		-0.0629	0.025	-2.480	0.013	-0.113	-0.013
weekday		0.0252	0.014	1.802	0.072	-0.002	0.053
atemp		0.5001	0.033	15.302	0.000	0.436	0.564
slight_to_heavy_rain_or_snow		-0.2388	0.026	-9.050	0.000	-0.291	-0.187
spring		-0.1085	0.017	-6.315	0.000	-0.142	-0.075
winter		0.0613	0.013	4.615	0.000	0.035	0.087
Omnibus:	80.515			Durbin-Watson:		1.834	
Prob(Omnibus):	0.000			Jarque-Bera (JB):		177.446	
Skew:	-0.847			Prob(JB):		2.94e-39	
Kurtosis:	5.341			Cond. No.		13.2	

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

VIF comparisons

	Features	VIF
3	atemp	3.29
2	weekday	3.23
0	yr	2.06
5	spring	1.33
6	winter	1.32
4	slight_to_heavy_rain_or_snow	1.05
1	holiday	1.04

Once we have a better model we predict on the test data and prepare a histogram to see the distribution of difference in actual data and predicted data



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

The top 3 features contributing significantly towards the demand of shared bikes are as below.

1. atemp
2. yr
3. winter
4. weekday

The below equation built by our model also helps us identify the top feature that could help increase the demand .

```
cnt = 0.2315 * yr - 0.0629 * holiday + 0.0252 * weekday +  
0.5001 * atemp - 0.2388 * slight_to_heavy_rain_or_snow - 0.1085 *  
spring + 0.0613 * winter
```

General Subjective Questions

1. Explain the linear regression algorithm in detail.

Linear regression is used to predict the value of a feature based on the value of a feature . There are two types of linear regressions

- 1.Simple Linear Regression
- 2.Multiple linear regression

Simple Linear Regression :

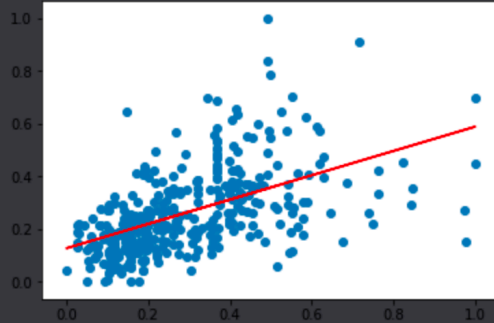
A simple linear regression could be simply put as $y = mx + c$, where m is the slope which shows the level of dependency , for a positive slope the dependency would be high and for a negative slope the dependency would be constant . The c represents a constant . This feature is built using a single variable . By using libraries from stats model library (`sm.OLS(y,X_feature_lm).fit()`) we can build the simple regression module

Below shows the linear regression model built for 1 feature , this shows a red line which is the linear line which fits the scatter plot , this line has a slope m which shows how inclined the line is and an intercept - which helps to find where this line cuts on the y axis .

Multiple linear Regression : Similar to simple linear regression , multiple linear regression has dependency on multiple variables , this can be represented using $y = m_1x_1 + m_2x_2 + m_3x_3 + \dots + c$, this shows that y value either increases or decreases keeping the other values 0 . A positive slope shows a better dependency and negative slope shows that the y value is inversely dependent on the feature . This model can be built using the stats model library (`sm.OLS(y,X_feature_lm).fit()`) we can build the simple regression module

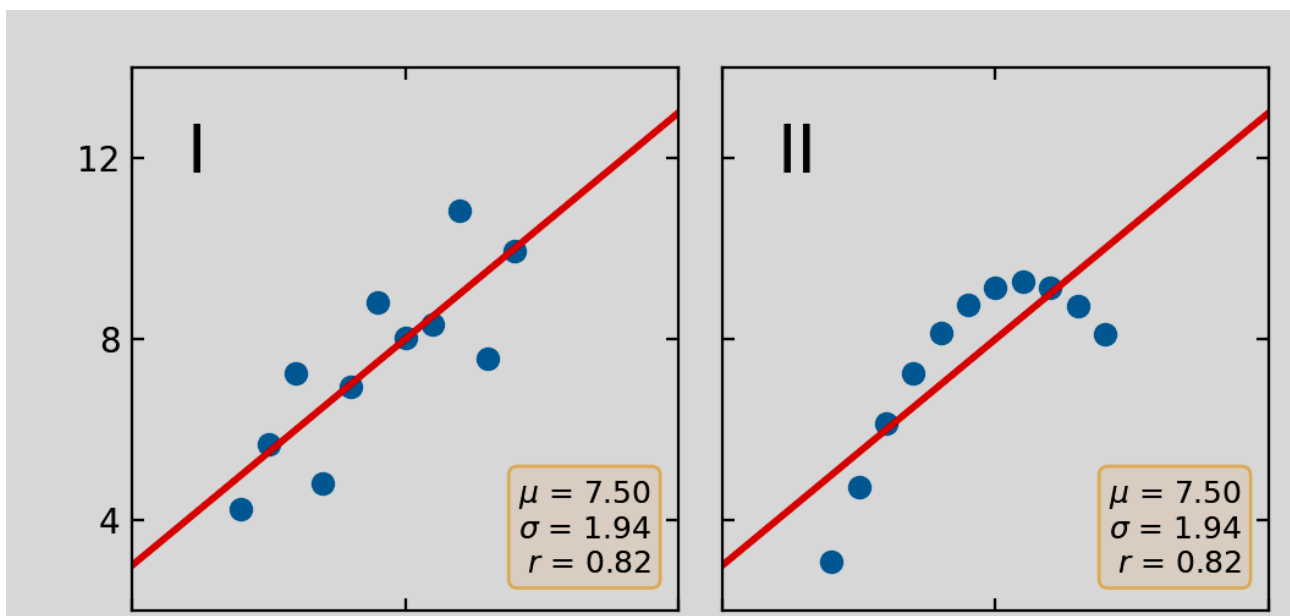
```
# Let's visualise the data with a scatter plot and the fitted regression line
plt.scatter(X_train_lm.iloc[:, 1], y_train)
plt.plot(X_train_lm.iloc[:, 1], 0.127 + 0.462*X_train_lm.iloc[:, 1], 'r')
plt.show()
```

</>



2. Explain the Anscombe's quartet in detail.

Anscombe's quartet is a group of data sets that have the same mean, standard deviation, regression line, but qualitatively they are different. Below figure shows the Anscombe quartet.



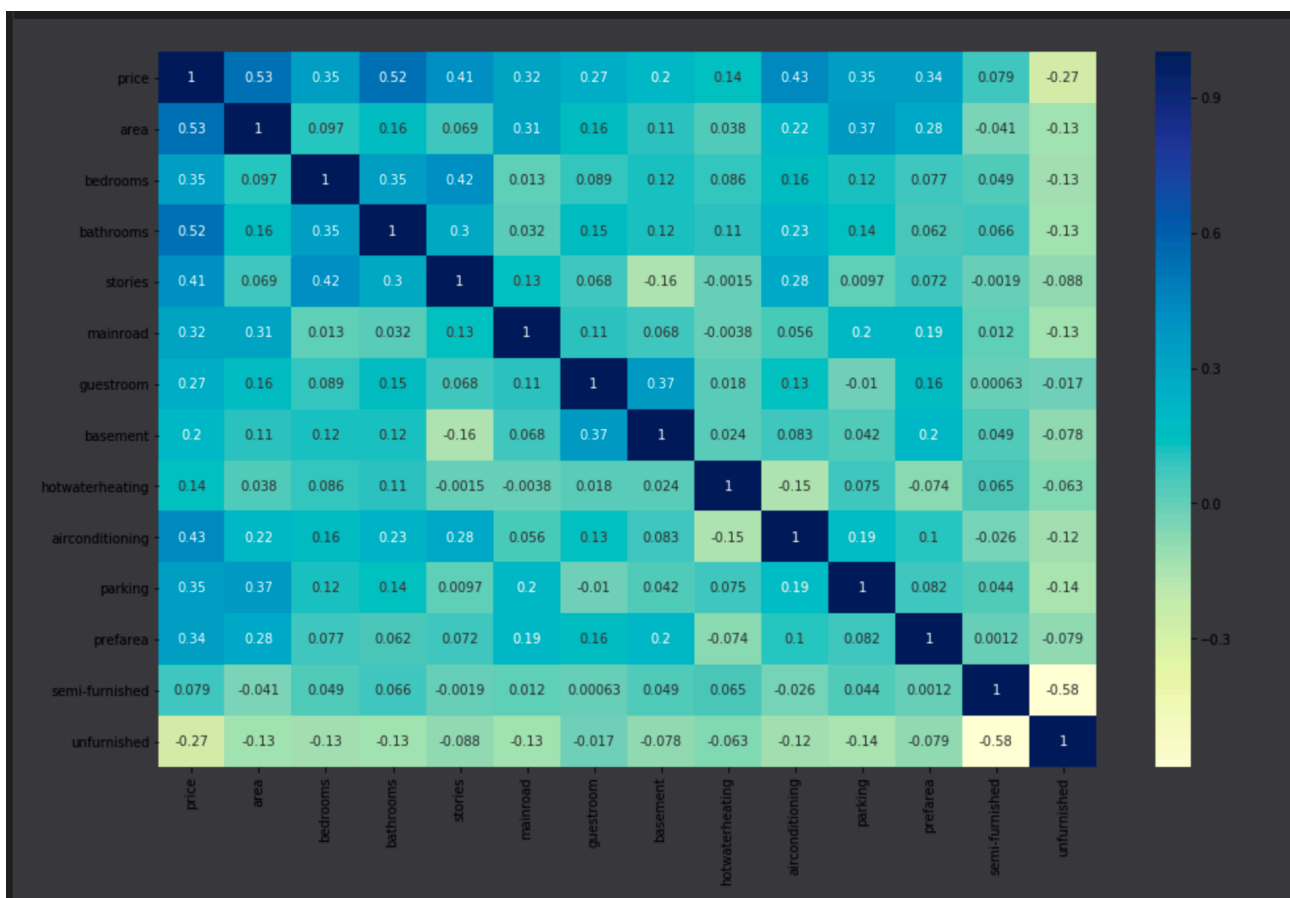
3. What is Pearson's R?

Pearsons R is a correlation coefficient , this helps to identify the correlation between features . The method corr() in pandas helps to derive the persons coefficient between different features
The below diagram shows how we could extract persons corr and how heat map help to quickly identify the correlation between features

```
plt.figure(figsize = (16, 10))
sns.heatmap(df_train.corr(), annot = True, cmap="YlGnBu")
plt.show()
```

The df_train.corr() in the above code helps to extract persons R value between the features .and below is the heat map which helps us better understand the correlation

The values lie between -1 to 1 where -1 shows negative correlation between features , +1 shows positive correlation between features and 0 shows les correlation.



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

Scaling is a transformation performed on the continuous data which doesn't lie between 0 -1 so that they can be bought in the same range as the categorical features , so that on building a model we won't end up with high slope values .

There are different ways of scaling

Normalised scaling : Normalised scaling is also called Min-Max scaling , where values are normalised so that they fall between 0 - 1. The mathematical min max scaling is as below

$$X_{sc} = (X - X_{min}) / (X_{max} - X_{min})$$

Standardised scaling :

The scaling is done so that the values are around the mean , the mathematical expression is

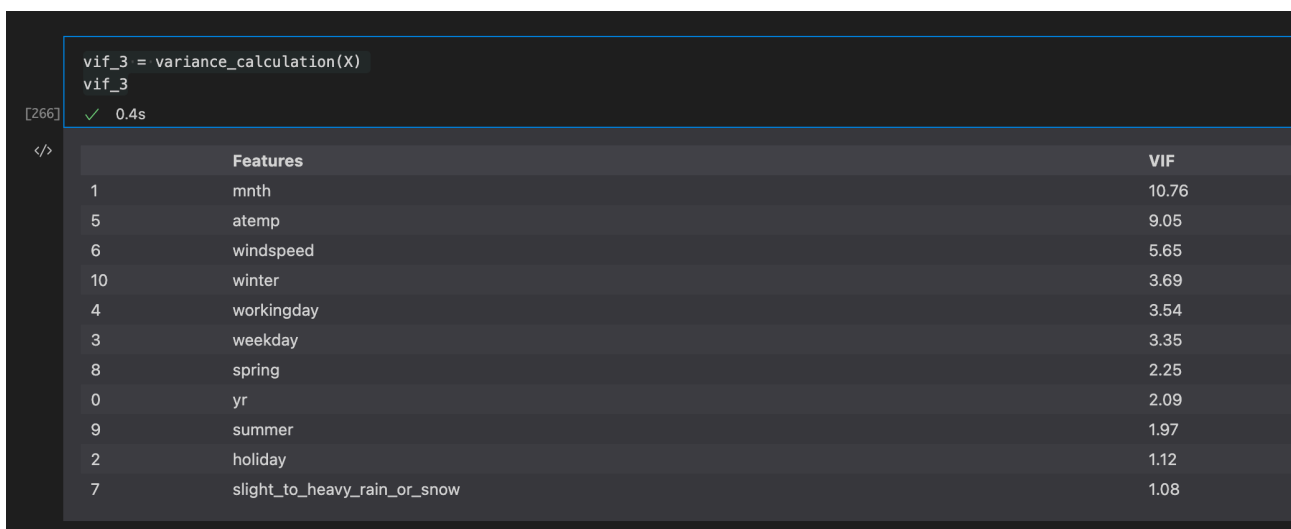
$$X' = \frac{X - \mu}{\sigma}$$

All the values are scaled around the mean with a unit standard deviation .

In short scaling helps to less slope values for the features .

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

VIF stands for Variance Inflation Factor , When building a machine learning module it helps us to find the correlation between features . A high VIF shows high correlation and low VIF shows low correlation . Any feature showing $VIF > 5$ is suggested to be dropped when building the model . In the image below we can see features with different VIF values for features .



```
vif_3 = variance_calculation(X)
vif_3
```

[266] ✓ 0.4s

	Features	VIF
1	mnth	10.76
5	atemp	9.05
6	windspeed	5.65
10	winter	3.69
4	workingday	3.54
3	weekday	3.35
8	spring	2.25
0	yr	2.09
9	summer	1.97
2	holiday	1.12
7	slight_to_heavy_rain_or_snow	1.08

A infinite VIF shows high correlation between the feature and causes multi collinearity , this results in building a unstable model , which poorly explains the dependency variable dependency on the feature variables .

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

Q-Q plots are also known as Quantile-Quantile plots. 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.

QQ plots is very useful to determine

- If two populations are of the same distribution
- If residuals follow a normal distribution or not . Having a normal error term is an assumption in regression and we can verify if it's met using this.
- Skewness of distribution