Assignment-based Subjective Questions

Q1. From your analysis of the categorical variables from the dataset, what could you infer about their effect on the dependent variable?

Answer: -

Categorical variables such as '*season*' and '*weathersit'* have a significant impact on bike demand (*cnt*).

i. Analysis on '*season*' shows that demand is highest during the fall and summer seasons, likely due to favourable weather and holidays, and lowest in winter.

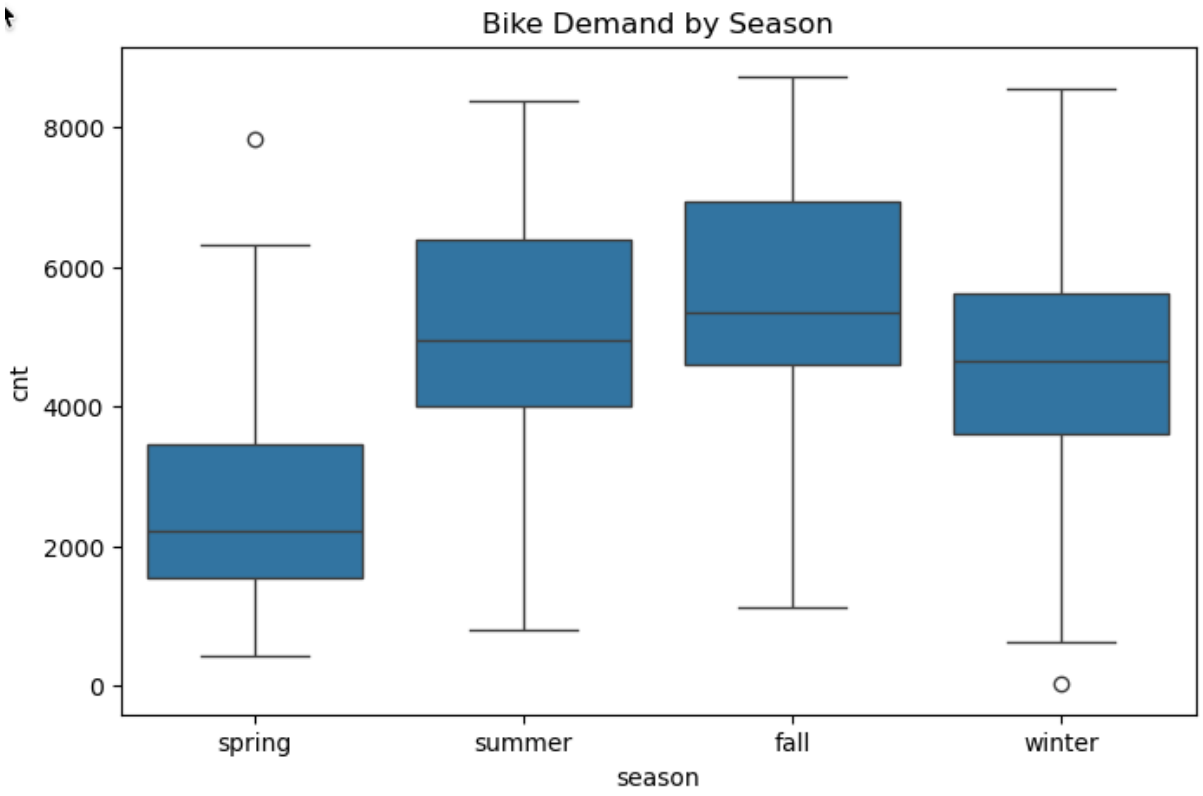
ii. The '*weathersit*' variable reveals that clear weather days see much higher rentals, while mist, light snow/rain, and especially heavy rain/snow drastically reduce demand. These effects are visible in boxplots and group means, confirming that both seasonality and weather conditions are key drivers of bike usage

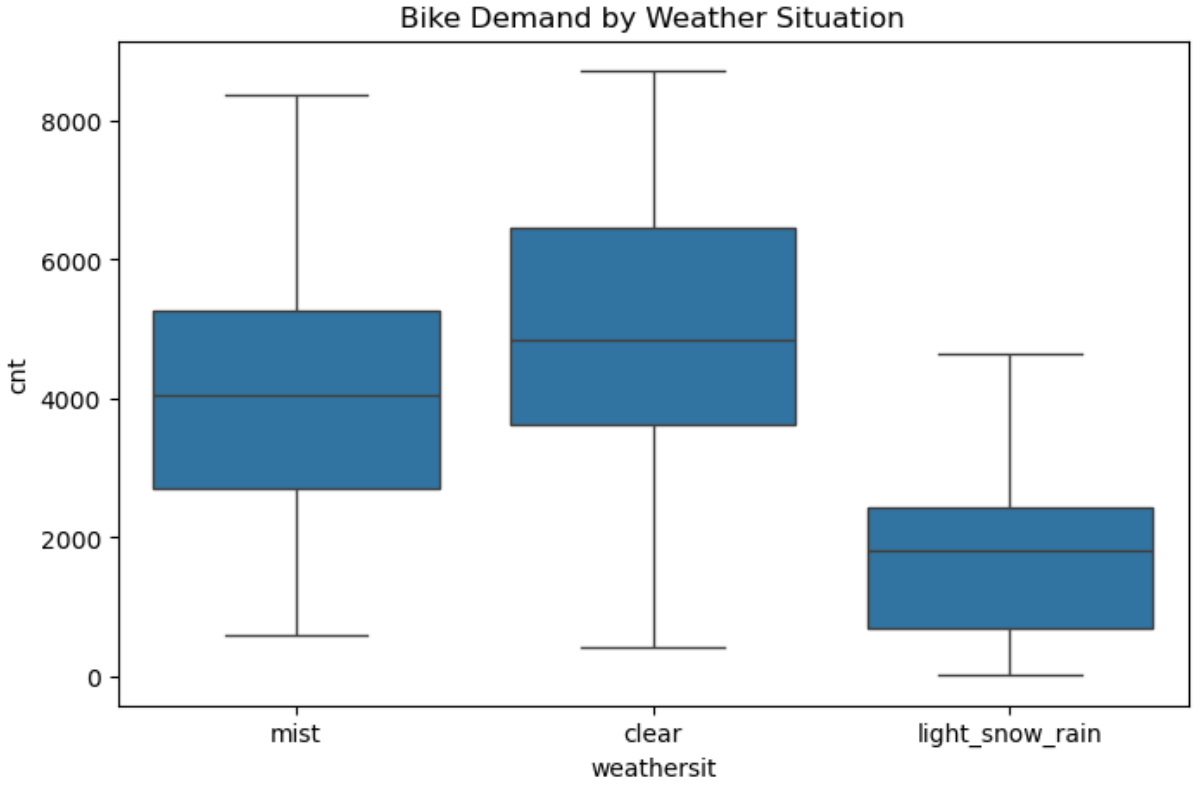
```
# Grouping by season and weathersit to see their effect on demand
season_effect = bike_df.groupby('season')['cnt'].mean()
weather_effect = bike_df.groupby('weathersit')['cnt'].mean()
print("Average demand by season:\n", season_effect)
print("\nAverage demand by weather situation:\n", weather_effect)
```

✓ 0.0s

```
Average demand by season:
 season
fall      5644.303191
spring    2608.411111
summer    4992.331522
winter    4728.162921
Name: cnt, dtype: float64

Average demand by weather situation:
 weathersit
clear              4876.786177
light_snow_rain    1803.285714
mist               4044.813008
Name: cnt, dtype: float64
```


Bike Demand by Season
cnt
8000
6000
4000
2000
0
spring
summer
fall
winter
season


Bike Demand by Weather Situation
cnt
8000
6000
4000
2000
0
mist
clear
light_snow_rain
weathersit

Q2. Why is it important to use *drop_first=True* during dummy variable creation?

Answer:-

When we make dummy variables, we turn categories into columns of 0s and 1s. If we keep all the columns, the model can get mixed up because one column can be figured out from the others. This can mess up the results.

By using *drop_first=True*, we leave out one category. The model then compares the other categories to the one we left out. This makes things easier for the model and helps us see how each category changes the result compared to the one we dropped.

```
# Create dummy variables for 'season' without dropping any category
season_dummies_all = pd.get_dummies(bike_df['season'], drop_first=False)
print('Dummy variables (all categories):')
print(season_dummies_all.head())

# Create dummy variables for 'season' with drop_first=True
season_dummies_drop = pd.get_dummies(bike_df['season'], drop_first=True)
print('\nDummy variables (drop_first=True):')
print(season_dummies_drop.head())

# Simple explanation:
# If we keep all columns, one column can be predicted from the others, which can confuse the model.
# By dropping the first column, the model compares other seasons to the one we left out.
```

✓ 0.0s

```
Dummy variables (all categories):
    fall  spring  summer  winter
0  False    True   False   False
1  False    True   False   False
2  False    True   False   False
3  False    True   False   False
4  False    True   False   False

Dummy variables (drop_first=True):
   spring  summer  winter
0    True   False   False
1    True   False   False
2    True   False   False
3    True   False   False
4    True   False   False
```

Q3. Looking at the pair-plot among the numerical variables, which one has the highest correlation with the target variable?

Answer:-

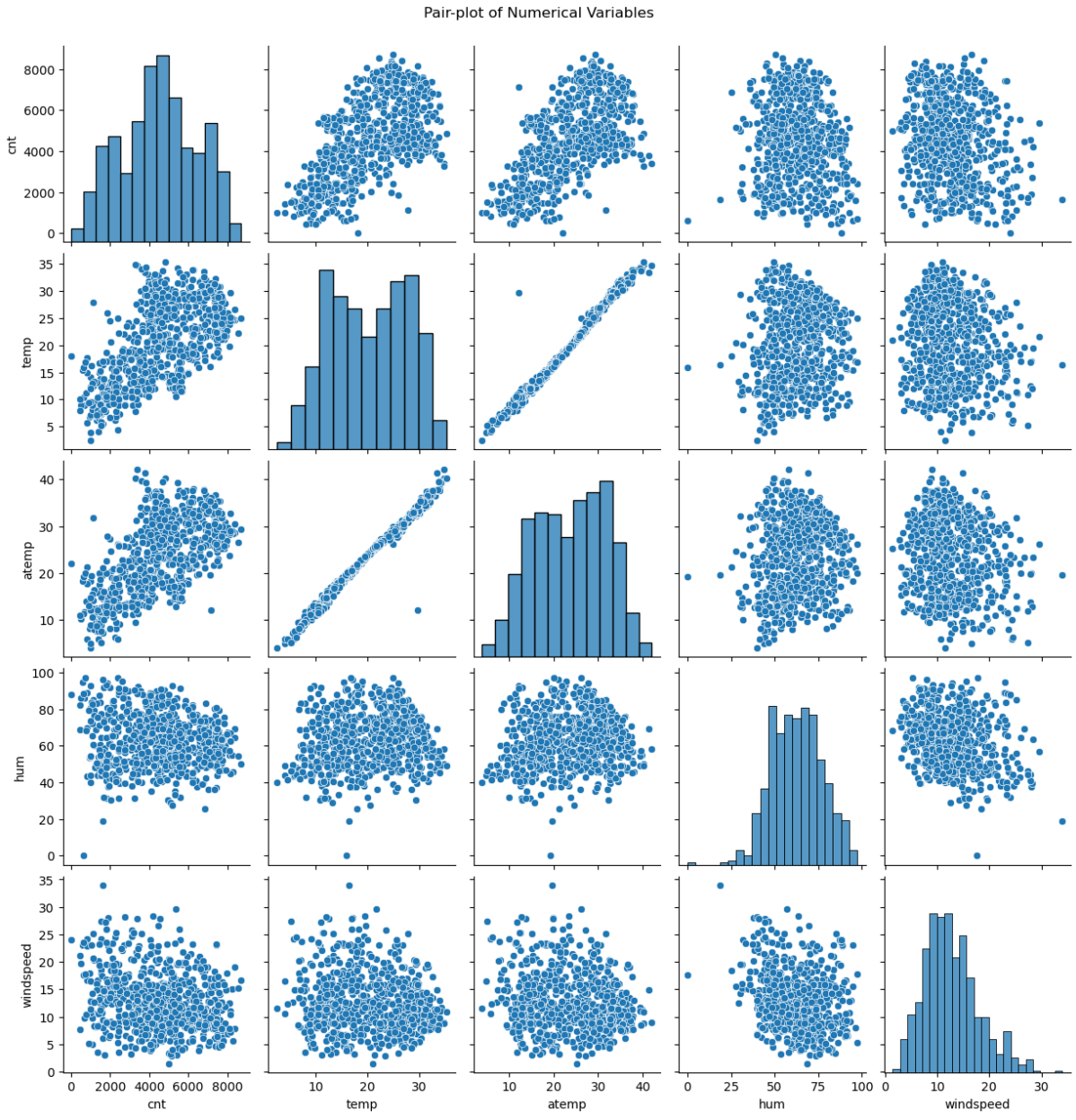
The pair-plot and correlation matrix show that the *'registered'* variable has the highest correlation with the target variable *'cnt'* (total rentals). This makes sense because *'cnt'* is the sum of *'casual'* and *'registered'* users. However, since *'registered'* is part of the target, it's not used as a predictor in the model.

Among the external predictors, *'temp'* (temperature) has the strongest positive correlation with *'cnt'* . This means that on warmer days, more people tend to rent bikes. The relationship is clear both visually in the pair-plot (where the points for *'temp'* and *'cnt'* show a strong upward trend) and numerically in the correlation matrix, where temp has the highest correlation coefficient with *'cnt'* among the independent variables.

In summary:

i. *'registered'* has the highest correlation with *'cnt'* (expected, since it's part of the target).

ii. Among predictors, *'temp'* is most strongly correlated with bike demand, showing that temperature is a key driver for rentals.

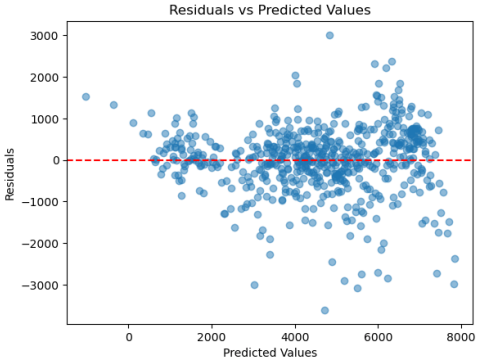


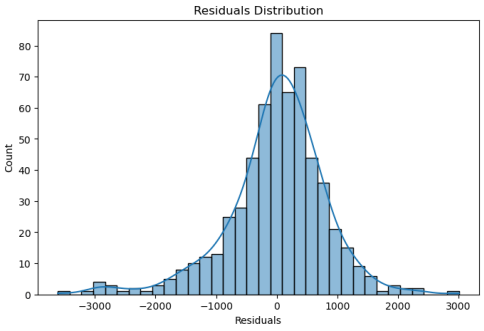
Q4. How did you validate the assumptions of Linear Regression after building the model on the training set?

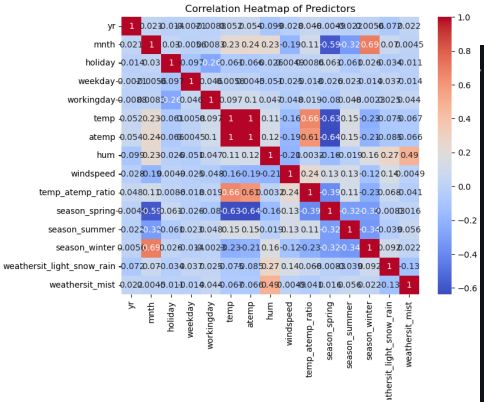
Answer:-

After building the model, I checked if the basic rules for linear regression were followed:

i. I made scatter plots to see if predictions and errors looked random (*linearity and independence*).
ii. I plotted the errors (*residuals*) to see if they looked like a normal curve (*normality*).
iii. I checked if the spread of errors was even across all predictions (*homoscedasticity*).
iv. I looked at how the input variables relate to each other to make sure none were too similar (*multicollinearity*).







	feature	VIF
0	yr	2.128198
1	mnth	17.824350
2	holiday	1.134541
3	weekday	3.205132
4	workingday	3.334794
5	temp	1620.495758
6	atemp	1713.995299
7	hum	40.275920
8	windspeed	9.795323
9	temp_atemp_ratio	129.340432
10	season_spring	7.611147
11	season_summer	3.512933
12	season_winter	5.000954
13	weathersit_light_snow_rain	1.391754
14	weathersit_mist	2.443165

Q5. Based on the final model, which are the top 3 features contributing significantly towards explaining the demand of the shared bikes?

Answer:-

The top 3 things that most affect bike demand are:

i. *'season_fall'*: Indicates if the day is in the fall season.
ii. *'weathersit_clear'*: Indicates if the weather is clear.
iii. *'temp'*: The normalized temperature for the day.

These were the most important features in the model and when looking at the data.

Top 3 features that affect bike demand the most:

	Feature	Coefficient
9	temp_atemp_ratio	8040.440874
13	weathersit_light_snow_rain	-2099.081243
0	yr	1961.401296

General Subjective Questions

Q1. Explain the linear regression algorithm in detail.

Answer:-

Linear regression is a supervised learning algorithm used to predict a continuous outcome (target variable) based on one or more input features. It finds the best-fitting straight that describes the relationship between the input variables and the target.

Key Concepts:

1. **Equation:**
   For simple linear regression (one feature):
   $$y = \beta_0 + \beta_1 x + \epsilon$$
   For multiple linear regression (many features):
   $$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$
   y is the predicted value.
   $x_1, x_2, \ldots, x_n$ are input features.
   β_0 is the intercept (where the line crosses the y-axis).
   $\beta_1, \ldots, \beta_n$ are coefficients (slopes for each feature).
   ϵ is the error term (difference between actual and predicted).
2. **Goal:**
   Find the values of $\beta_0, \beta_1, \ldots, \beta_n$ that minimize the sum of squared errors (differences between actual and predicted values).
3. **Fitting the Model:**
   The algorithm uses a method called "least squares" to find the best coefficients.
   It calculates the line (or plane) that minimizes the total squared distance between the actual data points and the predicted values.
4. **Assumptions:**
   Linearity: The relationship between features and target is linear.
   Independence: Observations are independent.
   Homoscedasticity: Constant variance of errors.
   Normality: Errors are normally distributed.
   No multicollinearity: Features are not too similar to each other.
5. **Interpretation:**
   Each coefficient shows how much the target variable changes when that feature increases by one unit, keeping other features constant.
   The intercept is the predicted value when all features are zero.
6. **Evaluation:**
   Common metrics: R-squared (explained variance), Mean Squared Error (MSE).
   Residual analysis helps check if assumptions are met.

Q2. Explain the Anscombe's quartet in detail.

Answer:-

Anscombe's quartet is a famous example in statistics that consists of four different datasets. Each dataset has nearly identical summary statistics (mean, variance, correlation, regression line), but when graphed, they look very different. This demonstrates the importance of visualizing data, not just relying on summary statistics.

Key Points:

1. **What is Anscombe's quartet?**
    - Four datasets (I, II, III, IV) with the same mean, variance, correlation, and regression line.
    - Created by statistician Francis Anscombe in 1973.
2. **Why is it important?**
    - Shows that statistical properties alone can be misleading.
    - Highlights the need for data visualization to understand the true nature of data.
3. **Details of the datasets:**

i. All four have:
- Mean of x ≈ 9
- Mean of y ≈ 7.5
- Variance of x and y are similar
- Correlation between x and y ≈ 0.82
- Linear regression line: $y = 3 + 0.5x$

ii. But their scatter plots are very different:
- Dataset I: Linear relationship, fits regression well.
- Dataset II: Nonlinear (curved) relationship.
- Dataset III: Linear except for one outlier.
- Dataset IV: Most x values are the same except one outlier, which drives the correlation.

4. **Lesson:**
    - Always plot your data before analysing or modelling.
    - Outliers, nonlinearity, and data structure can't be detected by summary statistics alone.

Q3. What is Pearson's R?

Answer:-

Pearson's R (also called the Pearson correlation coefficient) is a measure of the linear relationship between two continuous variables. It tells you how strongly and in what direction the variables are related.

Key Points:

- Values range from -1 to +1.
  +1 means a perfect positive linear relationship.
  -1 means a perfect negative linear relationship.
  0 means no linear relationship.
- It is calculated using the covariance of the variables divided by the product of their standard deviations.
- Pearson's R only measures linear relationships, not nonlinear ones.

Formula:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

Pearson's R is a simple way to quantify how two variables move together. A value close to +1 or -1 means a strong relationship, while a value near 0 means little or no linear relationship.

Q4. What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling?

Answer:-

What is scaling?

Scaling is the process of transforming numerical features so they are on a similar range or scale. This helps algorithms treat all features equally, especially when they use distance or gradient calculations.

Why is scaling performed?

- Many machine learning algorithms (like linear regression, k-nearest neighbors, and neural networks) work better when features are on similar scales.
- Features with larger ranges can dominate the model, making it harder to learn from other features.
- Scaling improves convergence speed and model accuracy.

Difference between normalized scaling and standardized scaling:

- **Normalized scaling (Min-Max Scaling):**
  - Rescales values to a fixed range, usually [0, 1].
  - Formula: $x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$
  - Keeps the shape of the original distribution but compresses the range.
  - Sensitive to outliers.
- **Standardized scaling (Z-score Standardization):**
  - Rescales values so they have mean 0 and standard deviation 1.
  - Formula: $x_{std} = \frac{x - \mu}{\sigma}$
  - Centres the data and adjusts for spread.
  - Less sensitive to outliers than normalization.

Q5. You might have observed that sometimes the value of VIF is infinite. Why does this happen?

Answer:-

A VIF (Variance Inflation Factor) value becomes infinite when one feature is a perfect linear combination of one or more other features in your dataset. This means there is perfect multicollinearity—one column can be exactly predicted from others.

Why does this happen?

- If two or more columns are duplicates or have a perfect linear relationship (e.g., one column is always twice another), the denominator in the VIF formula becomes zero, causing the VIF to be infinite.
- This usually happens if you include all dummy variables for a categorical feature (without dropping one), or if you accidentally include redundant columns.

Summary:
Infinite VIF means perfect multicollinearity. To avoid this, always check for duplicate or redundant features and use drop_first=True when creating dummy variables.

Q6. What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression.

Answer:-

A Q-Q plot (Quantile-Quantile plot) is a graphical tool used to compare the distribution of a dataset to a theoretical distribution, usually the normal distribution. It plots the quantiles of your data against the quantiles of the reference distribution.

Use in Linear Regression:

- In linear regression, a Q-Q plot is commonly used to check if the residuals (errors) are normally distributed.
- Normality of residuals is an important assumption for valid hypothesis tests and confidence intervals in regression.

How to interpret:

- If the points in the Q-Q plot fall roughly along a straight line, the data is approximately normally distributed.
- Deviations from the line (especially at the ends) indicate non-normality, such as skewness or heavy tails.

Importance:

- Helps diagnose problems with model assumptions.
- If residuals are not normal, it may affect the reliability of p-values, confidence intervals, and predictions.
- Guides you to consider data transformations or alternative models if normality is violated.

A Q-Q plot is a simple but powerful way to visually check the normality of residuals in linear regression, ensuring your model's statistical inferences are trustworthy.