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?

Ans:- Effect of each categorcal column on dependent variable cnt :

1. **season**

Season exerts a strong influence.

In the bike hiring, the highest is the summer and autumn, the spring is moderate and the winter is the lowest.

2. **yr**

Year has an effect on rentals.

The year has an impact on rents.

Bike renting is higher in 2019 (yr = 1) than in 2018 (yr = 0), because the bike-sharing system has increasingly gained popularity.

3. **mnth**

Month influences bicycle use as well.

High consumption of rentals were recorded in the warm months, from May to October, followed by a rapid decrease among colder months, from December to February.

4. **holiday**

Rentals count grinds to a halt on holidays because not as many people want to work or go to school.

5. **weekday**

There’s not much of a variation over the course of the week.

6. **working day**

Because of work day commuters, more people rent bikes on working days.

But the number of casual users during weekend is also relatively high, the difference for all users is not very big.

7. **weathersit**

Weather condition has the strongest impact.

Clear / partly cloudy weather → highest rentals

Mist → moderate rentals

Rain / snow → strong decline in rentals

---

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

Ans:- Using **drop_first=True** during dummy variable creation is important because it helps avoid the problem of **dummy variable trap**.

What is Dummy Variable Trap?

When you create dummy variables for a categorical column, you generate a separate column for each category. If all dummy columns are included, one column can be predicted from the others, which creates perfect multicollinearity.

Example: Suppose season has 4 categories → Spring, Summer, Fall, Winter.
Dummy variables become:

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

If we know the values of **three** dummy columns, the fourth can be **easily determined**.
This creates multicollinearity in regression models → coefficients become unstable and model interpretation becomes difficult.

Why drop_first=True solves this?

drop_first=True removes **one dummy column** automatically and keeps the remaining **(n − 1)** dummies.

So instead of **4 dummy columns**, you get **3**.
The dropped category becomes the **reference category**.

Regression can now compare other categories with the reference category without multicollinearity.

Example

Before using drop_first:

- season_Spring
- season_Summer
- season_Fall
- season_Winter

After using drop_first=True:

- season_Summer
- season_Fall
- season_Winter

(Spring becomes the reference category)

Now, if all three are 0 → it automatically means Spring.

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

Ans :- From the pair-plot of the numerical variables, the variable that shows the strongest positive correlation with the target variable cnt is:

atemp (Feels-like temperature in Celsius)

You can observe in the bottom row / last column of the pair-plot that:

- As **atemp** increases, **cnt** increases sharply and the scatter points form a clear upward linear trend.
- **temp** also has a strong positive relationship, but **atemp** shows a slightly clearer / tighter correlation with **cnt**.

---

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

1. **Linearity**

- We plot **predicted values vs residuals**.
- Residuals should be **randomly scattered around zero**.
- If a pattern appears (curve or funnel shape), linearity is violated.

2. **Normality of Residuals**

- We plot a **histogram / KDE plot of residuals** or a **Q-Q plot**.
- Residuals should follow a **normal (bell-shaped) distribution**.
- This confirms valid p-values and confidence intervals.

3. **Homoscedasticity (constant variance of residuals)**

- We again check **residuals vs fitted values plot**.
- The spread of residuals should be **constant** across all fitted values.
- No funnel shape → means variance is constant.

4. **No Multicollinearity**

- We calculate **VIF (Variance Inflation Factor)** for each predictor.
- VIF > 5 (or 10) indicates multicollinearity and requires feature removal.
- Ensures stable coefficient estimates.

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

Top 3 Significant Features

1. **atemp (feeling temperature)**

- Highest coefficient: 4546.85

- Very high t-value: 23.157

  As the “feeling temperature” increases, bike rentals increase significantly.

2. **yr (year – 2019 vs 2018)**

- Coefficient: 1995.32
- Very high t-value: 26.161

  Bike rentals in 2019 are much higher than in 2018, indicating growth in the service.

3. **season**

- Coefficient: 538.25
- High t-value: 7.994

  Season affects rentals, with summer and fall showing higher demand compared to winter and spring.

General Subjective Questions

1. Explain the linear regression algorithm in detail.

Linear Regression is a supervised machine learning algorithm used to predict a continuous numeric value (e.g., house price, salary, temperature).
It finds a straight-line relationship between the input variable(s) (X) and the target variable (Y).

How does it work?

The model fits a line to the data using the equation:

Y = β_0 + β_1 X

Where:

- Y → predicted value
- X → input feature
- β_0 (intercept) → Y value when X = 0
- β_1 (slope) → amount Y changes when X increases by 1 unit

The model adjusts β_0 and β_1 to reduce prediction errors.

Cost Function and Learning

To measure how well the model fits the data, we calculate error using Mean Squared Error (MSE):

MSE = 1 / n sum(Y_actual - Y_pred) ^ 2

The goal is to minimize MSE.
Optimization method like Gradient Descent updates β_0 and β_1 step-by-step until the smallest error is reached.

Evaluation Metrics

To check model performance, we use:

- R^2 score — how much variation is explained
- MAE / MSE / RMSE — error between actual & predicted values

Where is it used?

Area	Example
Real estate	Predict house prices
Finance	Predict stock trends
Business	Sales forecasting
Healthcare	Predict medical costs

2. **Explain the Anscombe's quartet in detail.**

Anscombe's Quartet is a set of four different datasets created by statistician Francis Anscombe in 1973.
Each dataset has almost identical summary statistics, including:

- Same mean of X and Y
- Same variance of X and Y
- Same correlation between X and Y
- Same linear regression line

BUT — when you visualize the datasets, they look completely different.

Why was Anscombe's Quartet created?

Anscombe designed these datasets to show an important lesson:

Summary statistics alone can be misleading — data visualization is essential.

Two datasets can look very different but still produce the same correlation and regression values, leading to incorrect conclusions if visual inspection is ignored.

The Four Datasets

All four datasets share these statistical properties:

Property	Value
Mean of X	9
Mean of Y	7.50
Correlation (X, Y)	≈ 0.816
Regression equation	(y = 3 + 0.5x)
Variance of X & Y	Almost identical

However:

Dataset	Visual Pattern
I	Follows a straight-line trend
II	Curved/non-linear relationship
III	Linear trend but one extreme outlier pulls the line
IV	X values are nearly constant except one extreme; looks vertical

Key Learning from Each Set

Dataset	Lesson
I	A typical linear relationship → regression works well
II	Relationship is non-linear, but linear regression incorrectly fits a straight line
III	Data looks linear except one outlier drastically affects regression
IV	Almost all points are identical → one outlier completely defines the regression

Main Insight

Even though all four datasets have the same statistical calculations,
their true patterns are different.

So, don't trust only numbers like mean, correlation, or R² score.
Always plot the data.

---

3. What is Pearson's R?

Pearson's R, also called the Pearson correlation coefficient, is a statistical measure that tells us how strongly two continuous variables are related to each other and in which direction.

It measures linear relationship only.

Range of Pearson's R

$$-1 <= R <= 1$$

Value of R	Meaning
+1	Perfect positive linear relationship
0	No linear relationship
–1	Perfect negative linear relationship
+0.5	Moderate positive correlation
–0.7	Strong negative correlation

Formula

$$R = Cov(X, Y) / \sigma X * \sigma Y$$

Where:

- Cov(X, Y) = covariance between X and Y
- σX, σY = standard deviation of X and Y

Interpretation

- If R is positive → when X increases, Y also increases
- If R is negative → when X increases, Y decreases
- If R is near 0 → no linear relation, but a non-linear relation may still exist

Important Notes

- Pearson's R works only for continuous numerical data
- Sensitive to outliers (one extreme value can change R a lot)
- Checks linear connection only, not non-linear

---

4. What is scaling? Why is scaling performed?

Scaling is a data preprocessing technique used to change the range of numerical features so that all values lie on a similar scale.
It ensures that no feature dominates others simply because of its larger numeric range.

Example without scaling:

Feature	Range
Age	18 – 60
Salary	20,000 – 2,00,000

Salary has a much larger range than Age, so models may treat it as more important — even when it isn't.

Why is Scaling Performed?

1. **To improve model performance**

Algorithms like KNN, SVM, Logistic Regression, and Neural Networks use distance calculations or gradient updates.
If features are not scaled, the model becomes biased toward features with larger values.

2. **To speed up model training**

Scaled data allows gradient descent to converge faster, reducing training time.

3. **To avoid instability**

Without scaling, weights may oscillate or fail to update properly.

When scaling is important

Scaling is required for:

- Linear Regression
- Logistic Regression
- SVM
- K-Means & KNN
- PCA
- Neural Networks

Scaling is not mandatory for:

- Decision Trees
- Random Forests
- XGBoost
  (Because tree-based models split on thresholds, not distance)

Types of Scaling

Method	Output
Standardization (Z-score)	Mean = 0, Std dev = 1
Normalization (Min–Max)	Values between 0 and 1
Robust Scaling	Reduces effect of outliers

---

5. What is the difference between normalized scaling and standardized scaling?

Normalization transforms the data to a **fixed range, usually 0 to 1**.

$$X' = X - X\_min / X\_max - X\_min$$

Key Points

- Output range: **0 to 1**
- Sensitive to **outliers** (because it depends on min and max)
- Used when **distribution is not Gaussian (not normal)**
- Useful for **image processing, deep learning, KNN, distance-based models**

Standardized Scaling (Z-Score Standardization)

Standardization transforms the data to have **mean = 0 and standard deviation = 1**.

$$X' = x - \mu / \sigma$$

Where μ = mean, σ = standard deviation.

Key Points

- No fixed range; values can be **negative or positive**
- **Less affected by outliers** than normalization
- Suitable when **data is normally distributed**
- Commonly used in **Linear Regression, Logistic Regression, SVM, PCA, Neural Networks**

Quick Difference Table

Feature	Normalization	Standardization
Goal	Scale values to a fixed range	Center data around mean
Range	0 to 1	No fixed range
Formula	Min–Max	Z-Score
Outliers	Highly sensitive	Less sensitive
When to use	Non-Gaussian / distance models	Gaussian / ML models using gradient

---

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

VIF (Variance Inflation Factor) measures how strongly one predictor variable is correlated with other predictor variables in a regression model.
Normally,

$$\mathbf{VIF = 1 / (1 - R^2)}$$

Where (R^2) is the coefficient of determination from regressing one feature against all other features.

When does VIF become infinite?

VIF becomes infinite when:

$$\mathbf{R^2 = 1}$$

This happens when one feature is perfectly (or almost perfectly) predicted from other features — meaning there is perfect multicollinearity.

Example:

- **X2 = 2 × X1**
  or
- **X3 = X1 + X2** (exact linear combination)

In such cases:

- The model cannot separate the individual contribution of correlated features
- Their variances inflate infinitely → resulting in infinite VIF

Practical Meaning

Infinite VIF indicates:

- Two or more variables are linearly dependent
- Regression model cannot be computed properly
- Coefficients become unstable and unreliable

How to fix it?

- Remove one of the highly correlated variables
- Use PCA to combine correlated variables
- Apply regularization (Lasso / Ridge) if removing is not possible

---

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

A Q–Q plot (Quantile–Quantile plot) is a statistical graph used to check whether a dataset follows a particular theoretical distribution, most commonly the normal (Gaussian) distribution.

It compares:

- Quantiles of the sample data
  vs.
- Quantiles of a theoretical distribution

If the data is normally distributed, the points in the Q–Q plot will lie approximately on a straight diagonal line.

How to interpret a Q–Q Plot?

Pattern in Q–Q Plot	Interpretation
Points follow the straight line	Distribution ~ normal
Points curve upward/downward	Data has heavy/light tails
Points form an S-shape	Skewed distribution
Points show strong deviation	Far from normal

Why is a Q–Q Plot important in Linear Regression?

Linear Regression has an assumption that:
👉 Residuals (errors) must follow a normal distribution

This ensures:

- Valid p-values
- Valid confidence intervals
- Reliable hypothesis testing
- Stable coefficient estimation

A Q–Q plot applied to residuals of the model checks this assumption.

When is it useful?

A Q–Q plot helps detect:

- Skewness
- Heavy tails (kurtosis)
- Outliers
- Non-normal error distribution

If the residuals deviate significantly from the normal line, it indicates the regression model:

- might be missing variables
- might need transformation (e.g., log scaling)
- might be non-linear