Assignment-based Subjective Questions

Question 1. From your analysis of the categorical variables from the dataset, what could you infer about their effect on the dependent variable? (Do not edit)
Total Marks: 3 marks (Do not edit)
Answer: <Your answer for Question 1 goes below this line> (Do not edit)

1. Year (yr): Coefficient: 0.2355 - This positive coefficient indicates that 2019 (yr=1) has a significantly higher bike rental demand than 2018 (yr=0). This supports the idea that bike-sharing is gaining popularity.

2. Holiday: Coefficient: -0.0965 - This negative coefficient suggests that holidays tend to have a lower bike rental demand. This might be because people are less likely to commute or more likely to travel by other means.

3. Season:

season_spring: -0.1177 - This negative coefficient indicates that spring has a lower bike rental demand compared to the base season (which is likely fall, as it's not explicitly listed).

season_winter: 0.0455 - This positive coefficient indicates that winter has a slightly higher demand than spring, but still lower than the base season (fall). It is important to note that this is still lower than the average, as spring was significantly negative. This is somewhat counter intuitive, but can be explained by the fact that the base season is likely fall.

4. Weather Situation (weathersit):

weathersit_Light Snow/Rain: -0.2875 - This strongly negative coefficient shows that light snow or rain significantly reduces bike rental demand.

weathersit_Mist/Cloudy: -0.0771 - This negative coefficient shows that mist or cloudy weather also reduces bike rental demand, but not as severely as light snow/rain.

5. Month (mnth_9 - September):mnth_9: 0.0696 - This positive coefficient indicates that September has a higher bike rental demand compared to the base month. This aligns with the understanding that fall months are popular for biking.

Question 2. Why is it important to use **drop_first=True** during dummy variable creation? (Do not edit)
Total Marks: 2 marks (Do not edit)
Answer: <Your answer for Question 2 goes below this line> (Do not edit)

Using drop_first=True during dummy variable creation is crucial to avoid multicollinearity, which can impact the reliability of our regression models. Multicollinearity occurs when predictor variables in a regression model are highly correlated with each other. When we create dummy variables for a categorical variable with n categories, we typically create n dummy variables.

If we include all n dummy variables in our regression model, we introduce perfect multicollinearity.

Question 3. Looking at the pair-plot among the numerical variables, which one has the highest correlation with the target variable? (Do not edit)
Total Marks: 1 mark (Do not edit)
Answer: <Your answer for Question 3 goes below this line> (Do not edit)

Registered variable has the highest correlation with the target variable followed by casual, temp and atemp. Sum of registered and casual is equal to target variable.

Question 4. How did you validate the assumptions of Linear Regression after building the model on the training set? (Do not edit)
Total Marks: 3 marks (Do not edit)
Answer: <Your answer for Question 4 goes below this line> (Do not edit)

We did Residual analysis, which validates linear regression assumptions (linearity, independence, homoscedasticity, normality), detects model errors (non-linearity, missing variables), identifies outliers, assesses model fit, and improves generalizability. It's crucial for ensuring model reliability and accuracy on unseen data.

Code used performs residual analysis for a linear regression model:

1. y_train_pred = lr.predict(X_train_sm): Predicts the target variable using the trained model (lr) on the training data (X_train_sm).
2. res = y_train - y_train_pred: Calculates the residuals (the difference between actual and predicted values).
3. sns.distplot(res): Creates a histogram of the residuals to check for normality.
4. sm.qqplot(res, line='45'): Creates a Q-Q plot of the residuals to further assess normality.
5. plt.show(): Displays the plots.

In short, it's checking if the model's errors are normally distributed, a key assumption for valid linear regression.

Then we performed prediction and evaluation on the test dataset using a linear regression model built with statsmodels. Here's a breakdown:

1. X_test_sm = sm.add_constant(X_test):
    - sm.add_constant(X_test): This line adds a constant term (intercept) to the test dataset (X_test).
    - statsmodels requires this constant term for its linear regression model to properly calculate the intercept.
    - The result is stored in X_test_sm, which now includes a column of ones representing the intercept.
    - This is done so that the test data aligns with how the model was trained. If the model was trained with a constant, the test data prediction must also include the constant.
2. y_pred = lr.predict(X_test_sm):
    - lr.predict(X_test_sm): This line uses the trained linear regression model (lr) to make predictions on the test dataset (X_test_sm).
    - The model calculates the predicted values for the target variable based on the features in X_test_sm and the learned coefficients.
    - The predicted values are stored in the y_pred variable.

In essence, these two lines prepare the test data for prediction and then generate the predicted target values using the trained model. This is a standard step in evaluating the model's performance on unseen data.

After that we plotted y_test and y_pred to understand the spread. The code creates a scatter plot of y_test (actual target values from the test set) versus y_pred (predicted target values from the model).

This plot is crucial for visualizing the model's performance and understanding how well the predictions align with the actual values. Here's why:

- If the model is a good fit, the points on the scatter plot should be clustered closely around a diagonal line (y = x).
- Deviations from this diagonal line indicate errors in the model's predictions.
- The plot can reveal patterns in the errors, such as:
  - Systematic overestimation or underestimation.
  - Regions where the model performs poorly.
  - Outliers that significantly deviate from the general trend.
- The plot provides a visual representation of how accurately the model predicts the target variable.
- It complements numerical evaluation metrics (like R-squared or MSE) by providing a graphical insight into the model's performance.

Essentially, this scatter plot helps to quickly assess whether our model's predictions are reliable and whether there are any systematic issues that need to be addressed. It is a very important tool for visually validating the model's performance on the test data.

At last we checked R-squared on test data

```
r2 = r2_score(y_test, y_pred)
print("R-squared:", r2)
```

This code calculates and prints the R-squared value, a metric that measures how well the regression model fits the test data. r2_score(y_test, y_pred) compares the actual test values (y_test) to the model's predictions (y_pred), and the result indicates the proportion of variance in y_test explained by the model. A higher R-squared (closer to 1) signifies a better fit.

Question 5. Based on the final model, which are the top 3 features contributing significantly towards explaining the demand of the shared bikes? (Do not edit)

Total Marks: 2 marks (Do not edit)

Answer: <Your answer for Question 5 goes below this line> (Do not edit)

Temperature (temp): Coefficient: 0.4060 - Temperature has the largest positive coefficient, indicating that it has the strongest positive impact on bike rental demand. As temperature increases, the demand for shared bikes significantly increases.

Year (yr): Coefficient: 0.2355 - The year variable also has a significant positive coefficient, showing that the demand for shared bikes in 2019 (yr=1) is considerably higher than in 2018 (yr=0). This reflects the growing popularity of bike-sharing systems.

Weather Situation - Light Snow/Rain (weathersit_Light Snow/Rain): Coefficient: -0.2875 - While this is a negative coefficient, its large magnitude signifies a substantial impact. Light snow or rain significantly reduces bike rental demand. This shows that bad weather has a very large affect on the dependent variable.

General Subjective Questions

Question 6. Explain the linear regression algorithm in detail. (Do not edit)
Total Marks: 4 marks (Do not edit)
Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 6 goes here>
Linear regression is a widely used statistical algorithm for modeling the relationship between a dependent variable (also called the target variable) and one or more independent variables (also called predictors or features). It aims to find the best-fitting linear relationship that describes how the independent variables affect the dependent variable. ⍰

- Linear regression assumes that there's a linear relationship between the input features and the output target.
- It seeks to find the line (or hyperplane in higher dimensions) that best fits the data points.
- "Best-fitting" is typically defined as the line that minimizes the sum of squared differences between the actual target values and the predicted target values.

Simple Linear Regression (One Independent Variable):

- When there's only one independent variable (x) and one dependent variable (y), the relationship is represented by a straight line:
  - $y = mx + b$
  - y: Dependent variable
  - x: Independent variable
  - m: Slope of the line (how much y changes for a unit change in x)
  - b: Y-intercept (the value of y when x is 0)
  - The algorithm's goal is to find the optimal values for 'm' and 'b' that minimize the error.

Multiple Linear Regression (Multiple Independent Variables):

  - When there are multiple independent variables (x1, x2, ..., xn), the relationship is represented by a hyperplane:
  - $y = b0 + b1x1 + b2x2 + ... + bnxn$
  - y: Dependent variable
  - x1, x2, ..., xn: Independent variables
  - b0: Y-intercept
  - b1, b2, ..., bn: Coefficients (slopes) for each independent variable
  - The algorithm aims to find the optimal values for b0, b1, b2, ..., bn that minimize the error.

Cost Function (Loss Function):

- The most common cost function used in linear regression is the Mean Squared Error (MSE).
- MSE calculates the average of the squared differences between the predicted values and the actual values:
  - $MSE = (1/n) * \Sigma(yi - \hat{y}i)^2$
  - n: Number of data points
  - yi: Actual target value for the i-th data point
  - ŷi: Predicted target value for the i-th data point
- The goal of the algorithm is to minimize this cost function.

Optimization Methods:

- Ordinary Least Squares (OLS):
  - OLS is a direct method that analytically calculates the coefficients that minimize the MSE.
  - It involves solving a system of linear equations.
  - It's computationally efficient for smaller datasets.
- **Gradient Descent:**
  - Gradient descent is an iterative optimization algorithm that finds the minimum of the cost function by repeatedly adjusting the coefficients in the direction of the negative gradient.
  - It's particularly useful for large datasets where OLS might be computationally expensive.
  - It starts with initial coefficient values and iteratively updates them until convergence.
  - There are variations of Gradient Descent, such as Stochastic Gradient Descent(SGD), and mini-batch Gradient Descent.
- **Normal Equation:**
  - A closed form solution that directly computes the coefficients that minimize the cost function.
  - It is computationally expensive for large datasets.

Model Evaluation:

- **R-squared (Coefficient of Determination):**
  - Measures the proportion of the variance in the dependent variable that is explained by the independent variables.
  - Values range from 0 to 1, where 1 indicates a perfect fit.
- **Adjusted R-squared:**
  - Similar to R-squared, but it penalizes the model for including unnecessary independent variables.
- **Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE):**
  - These metrics measure the average error of the model's predictions.
- **Residual Analysis:**
  - Examining the residuals (the differences between actual and predicted values) is crucial for validating the assumptions of linear regression.

Assumptions of Linear Regression:

- **Linearity:** The relationship between the independent and dependent variables is linear.
- **Independence:** The residuals are independent of each other.
- **Homoscedasticity:** The variance of the residuals is constant.
- **Normality:** The residuals are normally distributed.
- **No Multicollinearity:** The independent variables are not highly correlated.

Question 7. Explain the Anscombe's quartet in detail. (Do not edit)
Total Marks: 3 marks (Do not edit)

Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 7 goes here>

Anscombe's quartet is a classic example in statistics that highlights the importance of visualizing data before relying solely on summary statistics. In essence, it demonstrates that datasets with nearly identical statistical properties can have drastically different visual representations.

Here's a breakdown of Anscombe's quartet:

What it is:

- Anscombe's quartet is a set of four datasets created by statistician Francis Anscombe in 1973.
- Each dataset consists of eleven (x, y) points.
- These four datasets have nearly identical summary statistics, including:
  - Mean of x values
  - Mean of y values
  - Variance of x values
  - Variance of y values
  - Correlation between x and y
  - Linear regression line

The Purpose:

- Anscombe created this quartet to emphasize the importance of visualizing data before analyzing it.
- It serves as a cautionary tale against relying solely on numerical summaries, which can be misleading.
- It underscores the need for exploratory data analysis, which includes graphical examination of data.

The Four Datasets:

- Dataset 1:
  - This dataset appears to have a simple linear relationship between x and y.
  - A linear regression model fits this data well.
- Dataset 2:
  - This dataset shows a clear non-linear relationship between x and y.
  - A linear regression model is not appropriate for this data.
- Dataset 3:
  - This dataset shows a linear relationship with one significant outlier.
  - The outlier heavily influences the regression line.
- Dataset 4:
  - This dataset has most x values the same, with one outlying x value.
  - This shows how one high leverage point can heavily influence a regression line.

Why it matters:

- Anscombe's quartet demonstrates that summary statistics alone can mask important patterns and relationships in data.
- Visualizing data through scatter plots and other graphical representations can reveal insights that would otherwise be missed.
- It reinforces the importance of exploratory data analysis in the statistical process.
- It is a very good tool for teaching people about the importance of looking at data visually.

Question 8. What is Pearson's R? (Do not edit)

Total Marks: 3 marks (Do not edit)
Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 8 goes here>

Pearson's R, more formally known as the Pearson correlation coefficient, is a statistical measure that quantifies the strength and direction of a linear relationship between two quantitative variables. Here's a breakdown of its key aspects:

Key Characteristics:

- Linear Relationship:
    - Pearson's R specifically measures the degree to which two variables are linearly related. This means it assesses how well the relationship can be represented by a straight line.
- Range:
    - The Pearson correlation coefficient (r) takes on values between -1 and +1:
        - +1 indicates a perfect positive linear relationship.
        - -1 indicates a perfect negative linear relationship.
        - 0 indicates no linear relationship.
- Strength and Direction:
    - The absolute value of r indicates the strength of the linear relationship:
        - Values closer to +1 or -1 indicate a strong relationship.
        - Values closer to 0 indicate a weak relationship.
    - The sign of r indicates the direction of the relationship:
        - A positive r means that as one variable increases, the other tends to increase as well.
        - A negative r means that as one variable increases, the other tends to decrease.

When to Use Pearson's R:

- Quantitative Variables:
    - Both variables being analyzed must be quantitative (numerical).
- Linear Relationship:
    - The relationship between the variables should be approximately linear.
- Normally Distributed Data:
    - Ideally, the variables should be approximately normally distributed.

Important Considerations:

- Correlation vs. Causation:
    - Pearson's R measures correlation, not causation. A strong correlation does not necessarily imply that one variable causes the other.
- Outliers:
    - Pearson's R can be sensitive to outliers, which can significantly affect the correlation coefficient.

Question 9. What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling? (Do not edit)
Total Marks: 3 marks (Do not edit)

Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 9 goes here>

Scaling refers to the process of transforming numerical features to a similar scale. This means adjusting the range of values for each feature so that they fall within a specific interval or have a particular distribution.

Scaling is performed for several important reasons:

1. Feature Dominance:
    - Features with larger ranges can dominate features with smaller ranges, especially in algorithms that rely on distance calculations (e.g., k-nearest neighbors, support vector machines, k-means clustering).
    - Scaling prevents features with larger values from having a disproportionate influence on the model.
2. Improved Model Performance:
    - Many machine learning algorithms, like gradient descent-based algorithms (used in linear regression, logistic regression, neural networks), converge faster when features are on a similar scale.
    - Scaling can improve the stability and accuracy of these algorithms.
3. Regularization:
    - Regularization techniques (e.g., L1, L2 regularization) are sensitive to the scale of features. Scaling ensures that regularization is applied fairly to all features.
4. Distance-Based Algorithms:
    - Algorithms that rely on distance metrics (e.g., Euclidean distance) are highly influenced by the scale of features. Scaling ensures that distances are calculated meaningfully.

Normalized Scaling vs. Standardized Scaling:

Both normalized and standardized scaling are common techniques, but they differ in their approach:

1. Normalized Scaling (Min-Max Scaling):

- Formula: X_scaled = (X - X_min) / (X_max - X_min)
- Range: Transforms features to a specific range, typically or [-1, 1].
- Use Cases:
    - Useful when you need to bound your data within a specific range.
    - Suitable when you know the exact upper and lower bounds of your data.
    - Sensitive to outliers, as they can significantly affect the X_min and X_max values.
- Example: If your data ranged from 20 to 100, and you wanted to scale it to between 0 and 1, a value of 60 would be transformed to 0.5.

2. Standardized Scaling (Z-score Normalization):

- Formula: X_scaled = (X - mean) / standard_deviation
- Distribution: Transforms features to have a mean of 0 and a standard deviation of 1 (standard normal distribution).
- Use Cases:
    - Widely used in many machine learning algorithms, especially those that assume a normal distribution.
    - Less sensitive to outliers compared to Min-Max scaling.
    - Effective when you don't know the exact bounds of your data.
- Example: Values that are above the mean will have a positive z-score, and values below the mean will have a negative z-score. The magnitude of the z-score indicates how many standard deviations away a value is from the mean.

Key Differences Summary:

- Range:
  - Normalization: Fixed range (e.g., [0, 1]).
  - Standardization: No fixed range.
- Distribution:
  - Normalization: Doesn't change the distribution shape.
  - Standardization: Transforms data to a standard normal distribution.
- Outlier Sensitivity:
  - Normalization: Sensitive to outliers.
  - Standardization: Less sensitive to outliers.

Question 10. You might have observed that sometimes the value of VIF is infinite. Why does this happen? (Do not edit)

Total Marks: 3 marks (Do not edit)

Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 10 goes here>

Encountering an infinite Variance Inflation Factor (VIF) is a clear indication of a severe problem in data. It happens due to perfect multicollinearity.

- VIF measures how much the variance of a regression coefficient is inflated due to multicollinearity (correlation between predictor variables).
- It's calculated as: VIF = 1 / (1 - R^2), where R^2 is the coefficient of determination when one predictor variable is regressed against all other predictor variables.

Why Infinite VIF Occurs:

- Perfect Correlation:
  - An infinite VIF occurs when there's a perfect linear relationship between predictor variables.
  - In other words, one predictor variable can be perfectly predicted from the others.
  - When this happens, the R^2 value in the VIF formula becomes 1, and 1 - R^2 becomes 0.
  - Dividing 1 by 0 results in infinity.
- Dummy Variable Trap (Without drop_first=True):
  - If we create dummy variables for a categorical variable with n categories and include all n dummy variables in your model, you introduce perfect multicollinearity.
  - This is a common cause of infinite VIF.
- Exact Linear Combinations:
  - If one predictor variable is an exact linear combination of other predictor variables, it will lead to perfect multicollinearity and an infinite VIF.
  - Example: If we have variables "x1", "x2", and "x3", and x3 = 2*x1 + x2, then x3 is perfectly predictable from x1 and x2.
- Redundant Variables:
  - If we have the exact same variable entered into the model twice, this will also result in an infinite VIF.

Consequences of Infinite VIF:

- Unstable Coefficients:
  - The coefficients of the affected variables become undefined or extremely unstable.

- Invalid Regression Results:
    - The regression model becomes unreliable, and the results cannot be trusted.
- Computational Issues:
    - Many statistical software packages will have difficulty handling infinite VIF values.

How to Resolve Infinite VIF:

- Remove Redundant Variables:
    - Identify and remove any variables that are perfectly correlated or exact linear combinations of others.
- Use drop_first=True:
    - When creating dummy variables, always use drop_first=True to avoid the dummy variable trap.
- Examine Your Data:
    - Carefully inspect your data to identify any variables that might be perfectly correlated.
- Combine Variables:
    - If two variables are perfectly correlated, and it makes sense to do so, combine them into one variable.
- Principal Component Analysis (PCA):
    - PCA can be used to reduce the dimensionality of your data and eliminate multicollinearity.

In summary, an infinite VIF is a red flag indicating perfect multicollinearity, which requires immediate attention to ensure the validity of your regression model.

Question 11. What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression. (Do not edit)

Total Marks: 3 marks (Do not edit)

Answer: Please write your answer below this line. (Do not edit)

<Your answer for Question 11 goes here>

A Q-Q plot (quantile-quantile plot) is a graphical tool used to assess whether a dataset follows a particular theoretical distribution, most commonly a normal distribution. In the context of linear regression, it is specifically used to check the assumption that the residuals (the differences between observed and predicted values) are normally distributed.

How a Q-Q Plot Works:

1. Quantiles:
    - Quantiles divide a dataset into equal-sized portions. For example, quartiles divide a dataset into four equal parts, and percentiles divide it into 100 equal parts.
    - The Q-Q plot compares the quantiles of your dataset (the residuals) to the quantiles of a theoretical distribution (usually a normal distribution).
2. Plotting:
    - The Q-Q plot plots the quantiles of your dataset against the quantiles of the theoretical distribution.
    - If your dataset follows the theoretical distribution, the points on the Q-Q plot will fall approximately along a straight diagonal line.

Use and Importance in Linear Regression:

1. Checking Normality of Residuals:

    - One of the key assumptions of linear regression is that the residuals are normally distributed.
    - The Q-Q plot is a powerful tool for visually assessing this assumption.
    - If the residuals are normally distributed, the points on the Q-Q plot will closely follow a straight line.
    - Deviations from the straight line indicate departures from normality.
2. Identifying Outliers:
    - Outliers can significantly affect the normality of residuals and the overall validity of the regression model.
    - Q-Q plots can help identify outliers, which appear as points that deviate substantially from the straight line.
3. Assessing Model Validity:
    - If the residuals are not normally distributed, it can affect the reliability of hypothesis tests and confidence intervals associated with the regression model.
    - The Q-Q plot helps determine if the normality assumption is met, which is crucial for ensuring the validity of the model's results.
4. Detecting Skewness and Kurtosis:
    - The shape of the deviations from the straight line can provide insights into the type of non-normality:
        - Curved pattern: Indicates skewness (asymmetry).
        - S-shaped pattern: Indicates heavier or lighter tails (kurtosis).

How to Interpret a Q-Q Plot:

- Straight Line:
    - If the points closely follow a straight line, it suggests that the residuals are approximately normally distributed.
- Deviations from the Line:
    - If the points deviate significantly from the line, it suggests that the residuals are not normally distributed.
    - The pattern of deviations can provide clues about the type of non-normality.