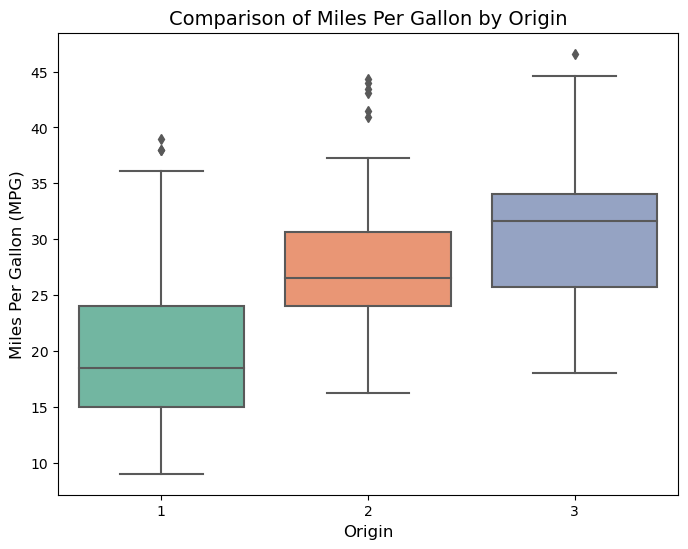
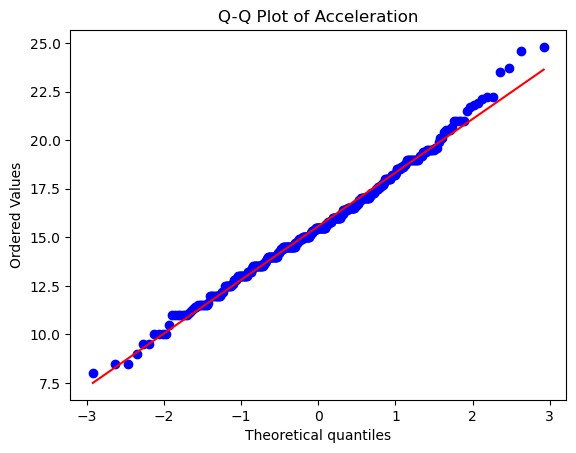
**Question 1**

**Key Insights** *(Figure 1)***:**

Figure 1: Comparing the Miles\_Per\_Gallon of the three groups Origin=1, 2, 3

1. Origin 3 cars are the most fuel efficient, this is because they have the highest median MPG and hyas the least spread among all three categories.
2. Origin 1 have the least fuel efficiency, it shows the lowest median MPG, the wide InterQuartile Range(IQR) shows high variability in fuel efficiency. It also shows a positively skewed distribution.
3. Origin 2 has higher median MPG than origin 1 and shows more compact distribution.

**Acceleration conclusion :**

Figure 2: Q-Q plot of Acceleration

The ‘Acceleration’ variable follows a normal distribution, because in the generated graph(*Figure2),* its evident thatthepoints closely follow diagonal line.

**Kolmogorov-Smirnov (K-S) test:**

**Statistics = 0.0508**

**p-value = 0.2466**

**Using alpha = 0.05 , as the level of significance.**

**From K-S test, results => Fail to reject Ho , ‘Acceleration’ **follows** a *normal distribution.***

****Shapiro-Wilk (S-W) test:****

**Using **alpha = 0.05,** as the significance level.**

**Statistic = 0.9924**

**p-value = 0.0399**

**From S-W test results, **Reject H0** , ‘Acceleration’ is **NOT normally** distributed.**

****K-S and S-W test conclusion**:**

****K-S test**  suggests that Acceleration follows a normal distribution while **S-W test**  suggests that Acceleration is not normally distributed, using additional methods as support (Q-Q plot in *Figure2* ), the conclusion is that it follows a normal distribution.**

**Question 2**

**1.**

1. **Idenitifying the closest Pair – From the matrix the **smallest** **nonzero distance** is **8.** Thatsbetween **Case 2**  and **Case 3** andbetween **Case 4** and **Case 5.****
2. **Forming clusters, merging **(Case 4, Case 5)** as one cluster and the next closest pair **(Case 2, Case 3)** .**
3. **They both have same proximity value, perfoming hierachial clustering, that concludes one cluster contains **Case 4** and Case 5, the other cluster conatins Cases 1,2 and 3.**
4. **Hence, the two cases that are in the same cluster are **Case 4** and **Case 5**, because they form the closest pair (with a squared Euclidean distance of 8) and can be grouped together when forming 2 clusters.**

**2.**

1. **Computing outputs of Hidden Neurons**

**Neuron 1 has Output *h1***

***h1 = f1(w1x1) = x1* = 0.8**

**Neuron 2 has Output h2**

***h2 = f2(w2x2) = x2* = 0.2**

1. **Computing the Input to Neuron 3**

***z3 = w3h1 + w4h2***

***z3 = (0.2×0.8)+(−0.091×0.2)***

***z3 = 0.16−0.0182 = 0.1418***

1. **Computing the output of Neuron 3**

**tanh(x)=(ex+e−x)/(ex−e−x​)**

***y^​ = f3​(z3​) = tanh(0.1418)***

**y^ = 0.1409**

****Question 3****

**Average test adjusted R2 for **Method 1** is 14.71%, its higher in comparison to **Method 2**  at 14.35%, therefore **Method 1** generalizes better to unseen data. Comparing Across Folds; **Method 2**  is more consistent while **Method 1** shows higher variance but outperfoms **Method 2** in 2 out of 3 folds.Therefore **Method 1** would be the choice for modelling, since it achieves a higher average adjusted R2 on test sets.**

**Question 4**

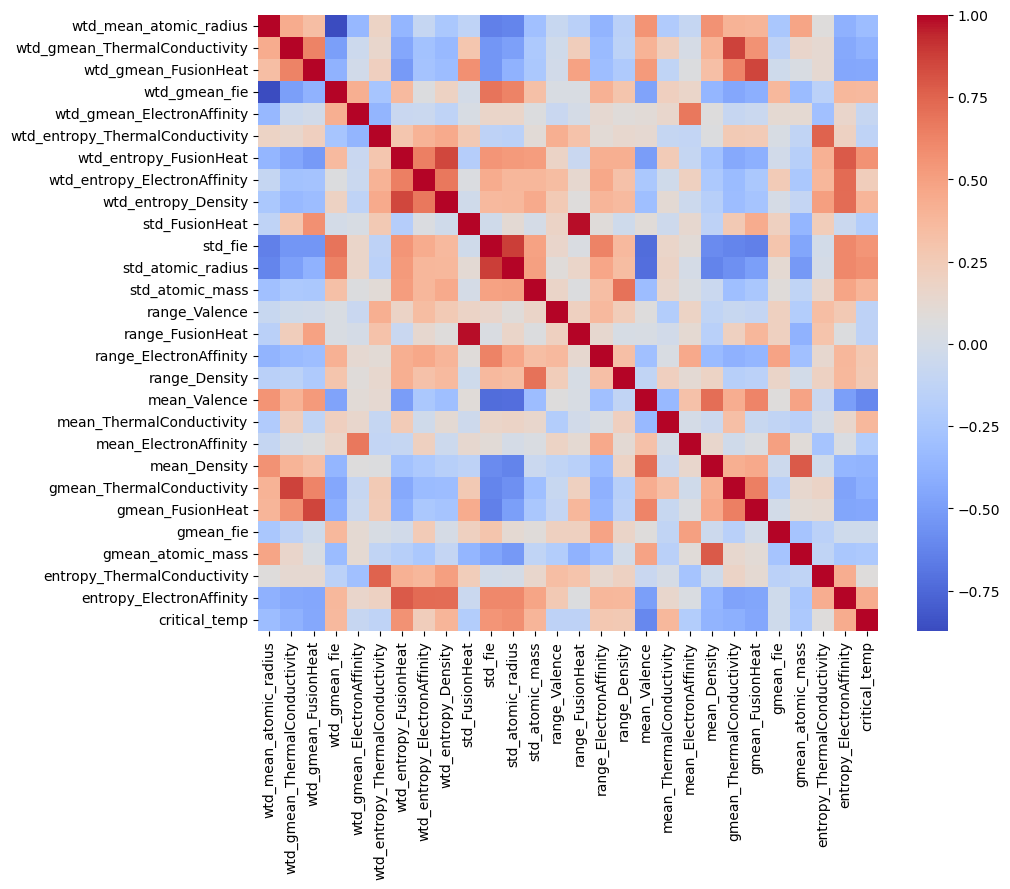
**Implementing Linear Regression in python :**

**Importing Necessary libraries**

**Loading the csv file (2025\_Regression\_Dataset.csv), using pandas library.**

**Perfomed Exploratory Data Analysis (EDA), to check for missing values, get the dataset information and summary statistics.**

**For feature selection, correlation analysis was used to check whioch features are most correlated with critical\_temp (as shown in *figure 3*).**

Figure 3: Correlation heatmap

**Selected features with the most correlation to critical\_temp.**

**Defined independent (X) as the selected features and depedent (y) variable as critical\_temp.**

**Split data into training and testing sets.**

**Trained the Linear Regression Model**

**Got the model coefficients, to show how each feature impacts critical\_temp.**

**Make predictions using X test.**

**Evaluated the model using Mean Squared Error and R-squared Score(to tell how well the model explains variability).**

**Visualized predictions (*Figure 4*).**

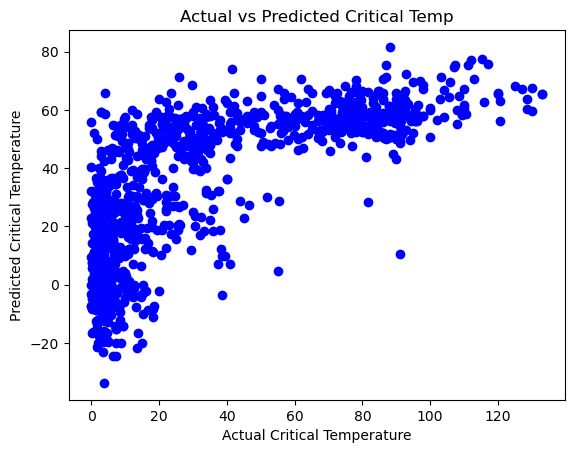
****Perfomance Analysis of the Linear Regression Model****

**The intercept (**9.858758) says that if all independent variables are zero, the model predicts critical\_temp to be 9.86.

The feature coefficients show how much critical\_temp cchanges for a unit increase in each feature,wtd\_entropy\_FusionHeatdisplay the **highest positive** impact (+45.88), wtd\_entropy\_ElectronAffinity had the **highest negative** impact (-29.57).

The perfomance Metrics Mean Squared Error (**MSE)** : 569.68, which is a relatively hgh, shows significant prediction erors.

The R-squared Score, 0.516, interpretes that the model explains about 51.6% of the variance which is mid but not great. This can be conluded that the model did not capture all the relationships in the data.

Figure 4: Actual vs Predicted Critical Temp

From the visual analysis, expected is that more points should align closely to the diagonal (y=x) but the plot (*figure 4*) shows scatter, in the low and high values showiung high variance and prediction errors.

****Question 5.****

1. **Loading the dataset and importing necessary libraries.**

**In data preprocessing , identifying and seperating depedent (variable) and independent variables.**

**Split the dataset to prevent data leakage before encoding categorical variables.**

**Used LabelEncoder on the training the data and transform both training and set tests.**

**Removed constant features with zero variance in the training set.**

**Ensured that the intercept (constant) isn’t removed during feature selection.**

**Standardized features.**

**Fit the model using increased max\_iter and solver**

**Then formed a modelk evaluation test, checking the accuracy score, confusion matrix and classification report on the predicted.**

1. **Model Interpretation**

**Intercept (-1.627) shows the log-odds of the event occuring when all predictor values are 0.**

**Shift (0.246): A one-unit increase in shift, increases the odds of the event occurring by 1.28 times.**

**Gpuls (0.168): A one-unit increase coresponds to 1.18 times increase of the oddds.**

**Nbumps2 (0.161): A one-unit increase in nbumps2 increases the odds by 1.17 times.**

**Nbumps3 (0.172): A one-unit increase in nbump3 increases the odds by 1.19 times.**

**If shifts increases by 1 unit, the odds of Y = 1 occurring multiply by 1.28.**

1. **Improving model perfomance**

**Feature Engineering : creating new variables or transformations (e.g polynomial terms, interaction terms).**

**Hyperparameter Tuning : Using cross-validations to adjust regularization (L1/L2 penalties).**

**Considering alternative Models: considering decision trees, random forestss or boosting algorithms.**

**Improve data quality: To address multicollinearity using VIF (Variance Inflation Factor) analysis and standardize continous variables.**

****Question 6****

1. **Model 1: Support Vector Machine (SVM)**

**Fit the model using SVC.**

**Perfomed Predictions.**

**And Claculated Results of Accuracy, Precision, Recall, F1 Score and ROC-AUC Score.**

**Model 2: CHAID Decision Tree**

**Fit the model using decision\_tree.**

**Perfomed predictions.**

**And Claculated Results of Accuracy, Precision, Recall, F1 Score and ROC-AUC Score.**

**Model 3: K-Nearest Neighbors (KNN)**

**Fit the model using KneighborsClassifier.**

**Perfomed Predictions.**

**And Claculated Results of Accuracy, Precision, Recall, F1 Score and ROC-AUC Score.**

1. **Perfomance Comparison**

Table 1: Perfomance comparison of the three models

|  |  |  |  |
| --- | --- | --- | --- |
| Metric | SVM | CHAID | KNN |
| Accuracy | 0.80 | 0.806 | 0.769 |
| Precision | 0.68 | 0.68 | 0.52 |
| Recall | 0.30 | 0.35 | 0.30 |
| F1 Score | 0.42 | 0.46 | 0.38 |
| ROC-AUC | 0.72 | 0.74 | 0.67 |

**Key Observations:**

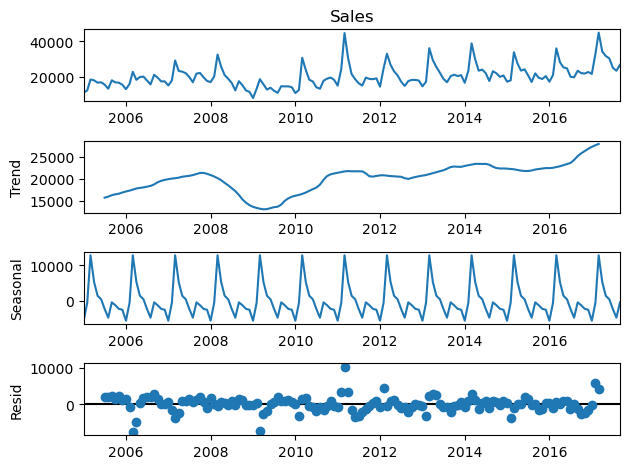
1. **CHAID is better than SVM and KNN in accuracy, recall, F1 and AUC-ROC.**
2. **Higher recall (0.35) indicates CHAID btter identifies true positives.**
3. **Superior AUC-ROC(0.74) suggests stronger discrimination between classes.**

**Future Use of CHAID:**

1. **Use the trained CHAID model to classify new instances by following hierachial splits.**
2. **Track perfomance drift using metrics like **F1** or **AUC-ROC** on new data.**
3. **Using CHAID’s split significancve to priotrize data collection for high-impact variables.**

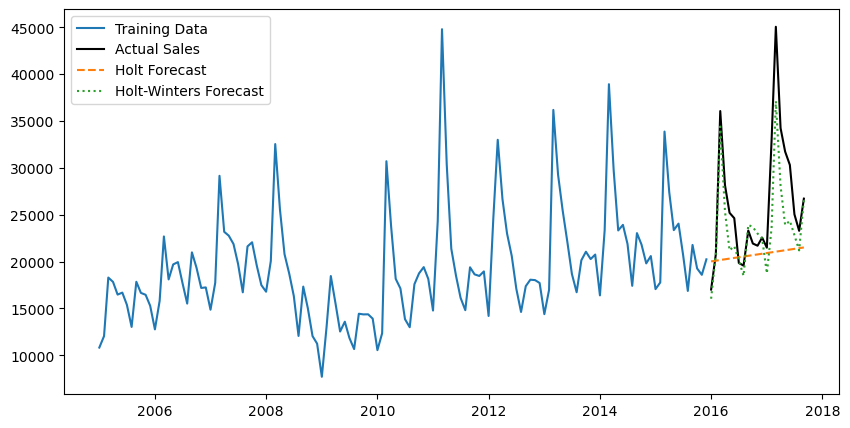
****Question 7****

**I).**

Figure 5: Time series decomposition plot

**A time series decomposition plot(*Figure 6*), breaks down the sales data into key components: Sales Plot – original sales time series, showing flunctuations over time, Trend Component - Suggesting a long-term growth pattern and Seasonal Component – follow repeating pattern, likely influenced by factors such as holidays.**

**II).In Figure 6**

Figure 6: HES model and Holt Winters

**Developed and applied 2 forecasting models: Holt’s Exponential Smoothing (HES) Model and Holt-Winters Model.**

**HES mathematical representation:**

**yt​=ℓt−1​+bt−1​+εt​**

**This model captures level an trend but does not account for seasonability.**

**Holt-Winters Model Mathematical representation:**

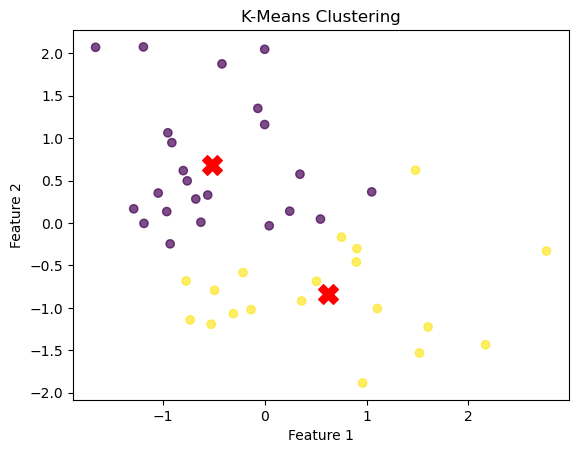
**yt​=ℓt−1​+bt−1​+st−m​+εt**

**It incorporates seasonability.**

**III) Model recommendation : Holt-winters model has lower Mean Squared Error, it is recommnded for this dataset.**

****Question 8.****

1. **The optimal number of clusters as seen are 2, indicated by the two large branches at the top.**
2. **The K-Means, identified 2 distinct groups in the datase, the clustering aligns well with the hierarchial clustering.**

Figure 7: K-Means clustering

**def dunn\_index(X, labels):**

unique\_clusters = np.unique(labels)

intra\_cluster\_distances = []

inter\_cluster\_distances = []

# Compute intra-cluster distances (maximum distance within each cluster)

for cluster in unique\_clusters:

points = X[labels == cluster]

if len(points) > 1:

intra\_cluster\_distances.append(np.max(pairwise\_distances(points)))

# Compute inter-cluster distances (minimum distance between clusters)

for i in range(len(unique\_clusters)):

for j in range(i + 1, len(unique\_clusters)):

points\_i = X[labels == unique\_clusters[i]]

points\_j = X[labels == unique\_clusters[j]]

inter\_cluster\_distances.append(np.min(cdist(points\_i, points\_j)))

return np.min(inter\_cluster\_distances) / np.max(intra\_cluster\_distances)

# Compute Dunn Index

dunn\_value = dunn\_index(X\_scaled, kmeans.labels\_)

print("Dunn Index:", dunn\_value)

**Output = Dunn Index: 0.2315306889122367**

****Question 9****

1. **Number of factors selected were 10, on basis of optimal balance between explained variance and factor interpretability. The selection is perfomed using Kaiser’s criterion.**
2. **Cummulative Variance Explained: 71.81%, this means the selected factors together explain 71.81 % of the dataset total variance.**
3. **The factor loadings after Varimax rotation show which variables are associated with strongly. Top variables associated with each factor:**

**Factor 9: Associated with q12, q13, q18**

**Factor 10L: Associated with q07, q21, q03**