Experiment 3

Aim: Perform Data Modeling.

Theory:

Data Partitioning:
Data partitioning is an essential step in data modeling and machine learning. It involves dividing the dataset into two subsets: training and testing datasets. The training set is used to build the model, while the testing set is used to evaluate its performance. A common split is 75% training and 25% testing.

Two-Sample Z-Test:
A two-sample Z-test is a statistical test used to determine if there is a significant difference between the means of two independent samples. This test is useful when comparing training and test datasets to check if their distributions are similar. The Z-test assumes that the data follows a normal distribution and that population variances are known or the sample size is large.

Problem Statement:

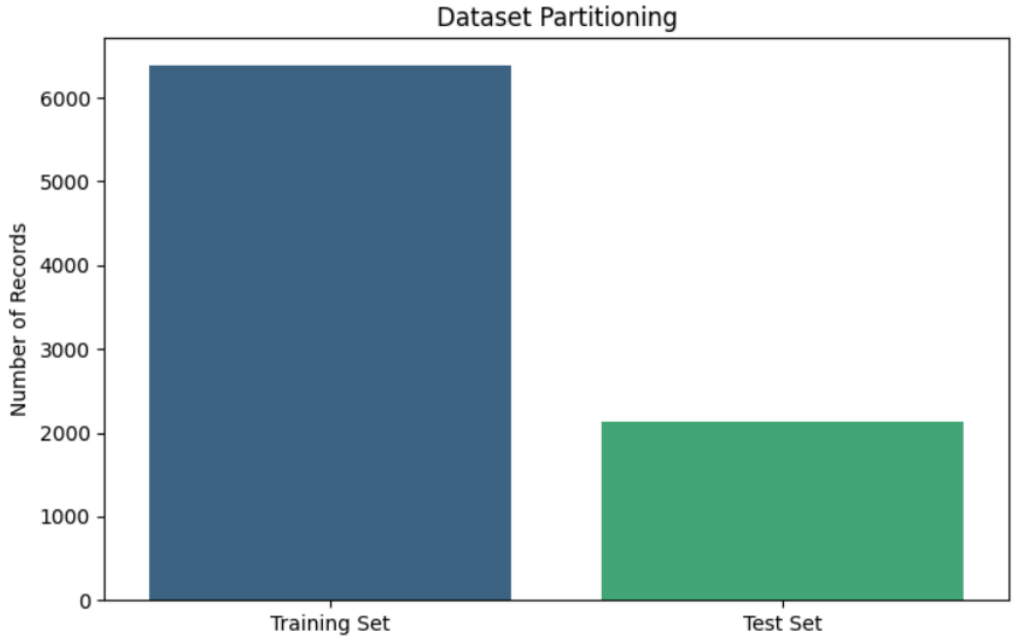
a. **Partition the data set**
   The dataset, consisting of 8,523 records, is partitioned into training (75%) and testing (25%) sets using train_test_split, ensuring a balanced distribution for model training and evaluation. The output confirms that 6,392 records are assigned to the training set and 2,131 records to the test set, maintaining the expected proportions. This step is crucial in data modeling to validate the model's performance on unseen data while preventing overfitting.

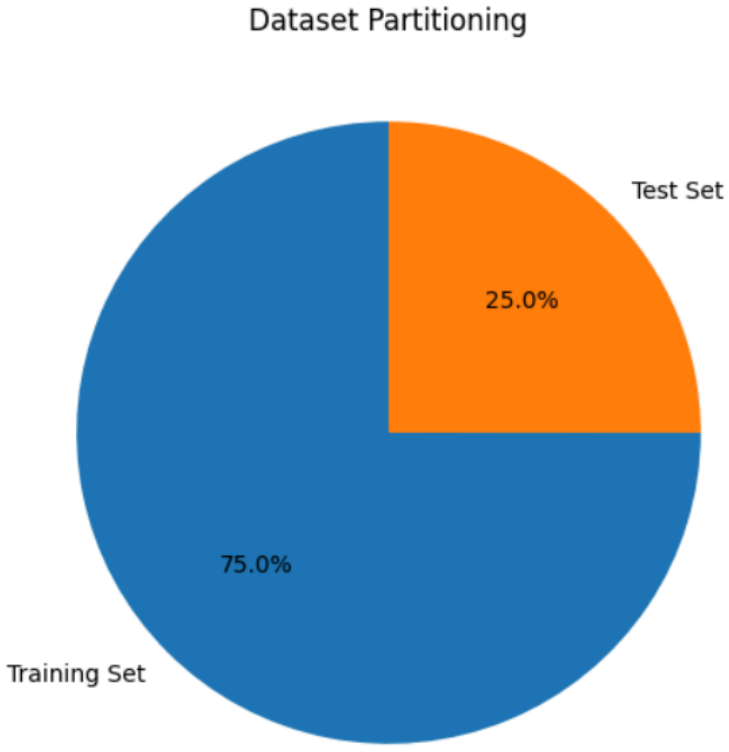
```
0s
train_df, test_df = train_test_split(df, test_size=0.25, random_state=42)

train_count = len(train_df)
test_count = len(test_df)
total_count = len(df)

print(f"Total records: {total_count}")
print(f"Training set records: {train_count} ({(train_count/total_count)*100:.2f}%)")
print(f"Test set records: {test_count} ({(test_count/total_count)*100:.2f}%)")
```

b. Use a bar graph and other relevant graphs to confirm your proportions.





Bar Graph: The bar graph displays the number of records in training as well as test set. The x axis represents the training and testing dataset while the y axis represents the number of instances in both the sets.

Pie Chart: The pie chart represents the percentage distribution of the data into training as well as testing dataset. It confirms our distribution of 75% for training and 25% for testing.

c. **Identify the total number of records in the training data set.**
We have successfully divided the dataset into training and testing data. The total number of instances in the original dataset were 8523 after preprocessing. Upon division the training data contained 6392 records (75% of original), while the testing data contained 2131 records (25% of original).

```
Total records: 8523
Training set records: 6392 (75.00%)
Test set records: 2131 (25.00%)
```

d. **Validate partition by performing a two-sample Z-test.**

A two-sample Z-test was conducted to compare the mean Item_Outlet_Sales values between the training and test datasets. The test aimed to determine whether there was a significant difference between the two groups. The calculated Z-score (-0.8980) and p-value (0.3692) suggest that the difference is not statistically significant at α = 0.05, indicating that the partitioning of this feature is balanced and does not introduce bias.

```
def sample_test(sample1, sample2):
    mean1, mean2 = np.mean(sample1), np.mean(sample2)
    std1, std2 = np.std(sample1, ddof=0), np.std(sample2, ddof=0)  # Use ddof=0 for Z-test
    n1, n2 = len(sample1), len(sample2)

    # Z-score formula for two-sample Z-test
    z_score = (mean1 - mean2) / np.sqrt((std1**2 / n1) + (std2**2 / n2))

    # Two-tailed p-value calculation
    p_value = 2 * (1 - norm.cdf(abs(z_score)))

    return z_score, p_value

# Call function and store output
z, p = sample_test(train_df["Item_Outlet_Sales"], test_df["Item_Outlet_Sales"])

# Print results
print(f"Z-score: {z:.4f}")
print(f"P-value: {p:.4f}")

# Interpretation
alpha = 0.05  # Significance level
if p < alpha:
    print("There is a significant difference in between training and test datasets.")
else:
    print("No significant difference in between training and test datasets.")
```

```
Z-score: -0.8980
P-value: 0.3692
No significant difference in between training and test datasets.
```

Conclusion:

The experiment successfully demonstrated the process of data partitioning and validation using statistical methods. The dataset was effectively split into 75% training (6,392 records) and 25% testing (2,131 records), ensuring a balanced distribution for

model training and evaluation. Visualization techniques such as bar graphs and pie charts confirmed the correct proportions of the partitioned data.

Furthermore, a two-sample Z-test was performed on Item_Outlet_Sales to validate the partitioning. The Z-score (-0.8980) and p-value (0.3692) indicated that there was no significant difference between the training and testing sets, confirming that the split was unbiased and statistically sound. This ensures that the model can be trained effectively without data distribution discrepancies affecting its performance.