

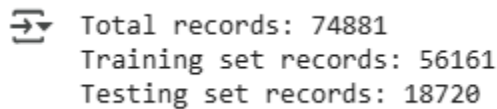
EXPERIMENT 3

Aim: Perform Data Modeling.

- 1. Partition the data set, for example 75% of the records are included in the training data set and 25% are included in the test data set.**

```
# Split dataset into 75% training and 25% testing
train_df = df.sample(frac=0.75, random_state=42) # 75% for training
test_df = df.drop(train_df.index) # Remaining 25% for testing

print(f"Total records: {len(df)}")
print(f"Training set records: {len(train_df)}")
print(f"Testing set records: {len(test_df)}")
```

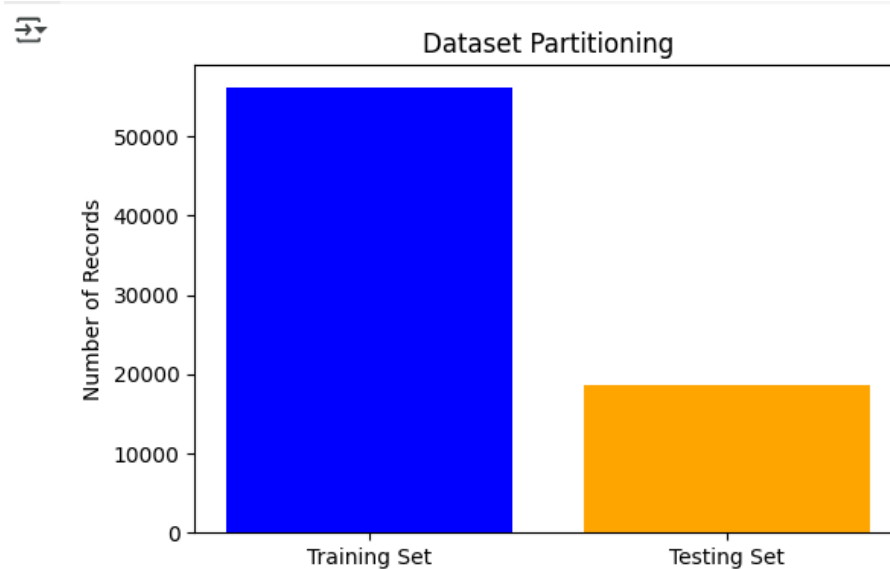


```
⇒ Total records: 74881
   Training set records: 56161
   Testing set records: 18720
```

- 2. Use a bar graph and other relevant graphs to confirm your proportions.**

```
import matplotlib.pyplot as plt

labels = ['Training Set', 'Testing Set']
values = [len(train_df), len(test_df)]
plt.figure(figsize=(6, 4))
plt.bar(labels, values, color=['blue', 'orange'])
plt.title('Dataset Partitioning')
plt.ylabel('Number of Records')
plt.show()
```



3. **Identify the total number of records in the training data set.**

```
print(f"Training set records: {len(train_df)}")
```

```
➡ Training set records: 56161
```

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

```
from scipy import stats
```

```
# Mean and standard deviation for train and test sets
```

```
mean_train = train_df['CRASH HOUR'].mean()
```

```
mean_test = test_df['CRASH HOUR'].mean()
```

```
std_train = train_df['CRASH HOUR'].std()
```

```
std_test = test_df['CRASH HOUR'].std()
```

```
n_train = len(train_df)
```

```
n_test = len(test_df)
```

```
# Perform a two-sample Z-test
```

```
z_score = (mean_train - mean_test) / ((std_train**2/n_train + std_test**2/n_test)  
** 0.5)
```

```
p_value = stats.norm.sf(abs(z_score)) * 2 # Two-tailed test
```

```
print(f"Z-score: {z_score:.4f}, P-value: {p_value:.4f}")
```

```
# Interpretation
if p_value > 0.05:
    print("No significant difference between Train and Test distributions (Good Split)")
else:
    print("Significant difference detected (Consider re-splitting)")
```

```
Z-score: -1.0303, P-value: 0.3029
No significant difference between Train and Test distributions (Good Split)
```

Conclusion:

In this experiment, we performed data modeling by partitioning the dataset into 75% training and 25% testing sets, ensuring a balanced split for model training and evaluation. A bar graph confirmed the proportionate distribution of records. The total number of records in the training set was validated, and a two-sample Z-test was conducted on the "CRASH HOUR" feature to compare the statistical properties of the training and testing sets. The results of the Z-test determined whether the split was unbiased and representative of the overall dataset. This step ensures that our model generalizes well, avoiding overfitting or underfitting, and provides a strong foundation for further predictive analysis.