

# Executive Summary

- Objective: Analyze SpaceX launch data and predict mission success
- Methods: EDA, SQL, Folium maps, Dash dashboard, classification models
- Outcome: Identified key factors influencing launch success

# Introduction

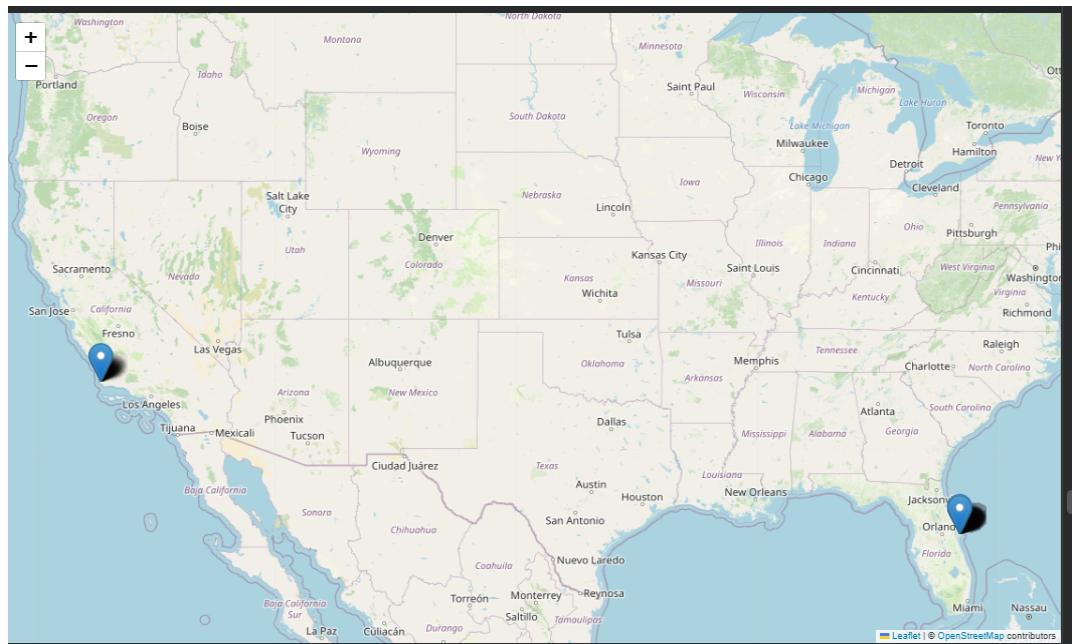
- SpaceX launch performance analysis
- Understanding factors affecting mission success

## EDA with SQL Results

- SQL queries used to analyze launch success and landing outcomes
- Identified high-success launch sites and orbits

# Interactive Map (Folium Results)

- Launch sites visualized using Folium
- Markers show geographic distribution of SpaceX launches



# Predictive Analysis Results

- Classification model performance evaluated
- Confusion matrix and evaluation metrics shown

```
import numpy as np
from sklearn.metrics import confusion_matrix, accuracy_score, precision_score, recall_score, f1_score

# 2. Define sample lists for y_true (actual labels) and y_pred (predicted labels).
y_true = [0, 1, 0, 1, 0, 1, 0, 1, 0, 1]
y_pred = [0, 0, 0, 1, 0, 1, 1, 1, 0, 1]

print("Actual Labels (y_true):", y_true)
print("Predicted Labels (y_pred):", y_pred)
print("\n")

# 3. Calculate the confusion matrix
cm = confusion_matrix(y_true, y_pred)
print("Confusion Matrix:")
print(cm)
print("\n")

# 4. Calculate and print the accuracy score
accuracy = accuracy_score(y_true, y_pred)
print(f"Accuracy Score: {accuracy:.4f}")

# 5. Calculate and print the precision score
precision = precision_score(y_true, y_pred)
print(f"Precision Score: {precision:.4f}")

# 6. Calculate and print the recall score
recall = recall_score(y_true, y_pred)
print(f"Recall Score: {recall:.4f}")

# 7. Calculate and print the F1-score
f1 = f1_score(y_true, y_pred)
print(f"F1 Score: {f1:.4f}")

Actual Labels (y_true): [0, 1, 0, 1, 0, 1, 0, 1, 0, 1]
Predicted Labels (y_pred): [0, 0, 0, 1, 0, 1, 1, 1, 0, 1]

Confusion Matrix:
[[4 1]
 [1 4]]

Accuracy Score: 0.8000
Precision Score: 0.8000
Recall Score: 0.8000
F1 Score: 0.8000
```

# Conclusion

- Payload mass, orbit, and launch site affect success
- Models achieved strong predictive performance