DATA ANALYSIS PRESENTATION

SUBMITTED BY :- SAAKCHI SINHA

SUMMARY

Dataset Part 2

📌 URL:

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_2.csv

🧾 Purpose: Contains the target variable — whether each mission was a success or failure.

📌 Key Details:

- Column: Class
  - Binary values:
    - 1 = Successful launch
    - 0 = Failed launch
- Number of Records: ~90+
- Usage:
- Used as the Y or dependent variable during predictive classification (Logistic Regression, SVM, etc.)

✅ Dataset Part 3

📌 URL:

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_3.csv

🧾 Purpose: Contains the features or independent variables for each SpaceX mission.

📌 Key Columns:

- Numerical Features:
  - flightnumber: Sequence number of the flight
  - payloadmass: Weight of the payload (in kg)
  - flights, block, reusedcount: Technical features of the rocket
- Categorical Features (one-hot encoded):
  - Orbits: e.g., orbit_iss, orbit_gto, orbit_leo, etc.
  - Launch Sites: e.g., launchsite_ccafs slc 40, ksc lc 39a, etc.
  - Landing Pads: Unique pad IDs used for booster landings
  - Serials: Booster ID codes (e.g., serial_b1049)
  - Booster Conditions: e.g., gridfins_true, reused_false, legs_true

📌 Total Columns: ~85+

- Exploratory analysis (EDA)
- Predictive modeling and evaluation

INTRODUCTION

This project explores SpaceX launch data to understand patterns behind successful and failed missions.
By combining exploratory data analysis, SQL querying, geospatial visualization, and machine learning models, we aim to:

- Identify key factors influencing mission success
- Visualize launch trends across locations and orbits
- Predict launch outcomes using classification algorithms

The ultimate goal is to derive data-driven insights that could help SpaceX improve mission reliability and strategic decision-making.

Dashboard Overview

Interactive Launch Analysis Dashboard with Plotly Dash

Content:

- A web-based dashboard built using Plotly Dash
- Allows users to interactively explore SpaceX launch data
- Key features include:
  - Dropdown to select launch site
  - Slider to filter payload mass
  - Pie chart showing mission outcomes
  - Scatter plot for payload vs. success analysis

Purpose:

To enable real-time, visual analysis of launch patterns and mission performance.

Key Features of the Dashboard

Dropdown Menu: Select specific launch sites or view all sites

Payload Slider: Filter data based on payload mass range

Pie Chart: Visualizes success vs. failure rate per selected site

Scatter Plot: Shows the correlation between payload mass and mission outcome, colored by booster version

Goal:

To give users an interactive experience exploring how different factors affect SpaceX launch success.

Insights from Interactive Dashboard

Content:

- Launch Site Performance:
- KSC LC 39A recorded the highest number of successful launches.
- Payload Trends:
- Launches with payloads between 2000–6000 kg had a higher success rate.
- Booster Version Impact:
- Boosters like FT and v1.1 show a higher correlation with successful missions.
- Visual Patterns:
- The scatter plot reveals a clear cluster of successful launches for certain payload ranges and sites.

Conclusion:

The dashboard helps users explore launch outcomes interactively, aiding decision-making and mission analysis.

Building the Classification Models

Classification Approach for Predicting Launch Success

Slide Content:

<u>Objective:</u>

Develop machine learning models to predict whether a SpaceX rocket launch will be successful (1) or fail (0) based on mission attributes.

Input Features Include:

- Flight number
- Payload mass
- Number of previous flights
- Booster block version
- Orbit type (one-hot encoded)
- Launch site and landing pad

Target Variable:

class → 1 for success, 0 for failure

Models Selected for Comparison:

✅ Logistic Regression (baseline linear model)

✅ Support Vector Machine (handles non-linear margins)

✅ Decision Tree Classifier (rule-based model)

✅ Random Forest Classifier (ensemble of decision trees)

EDA RESULTS WITH SQL

Display the names of the unique launch sites in the space mission

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
```

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Task 2

Display 5 records where launch sites begin with the string 'CCA'

```
%sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT SUM("Payload_Mass__kg_") AS Total_Payload_Mass FROM SPACEXTBL WHERE "Customer" = 'NASA (CRS)';
```

```
 * sqlite:///my_data1.db
Done.
```

Total_Payload_Mass
45596

Task 4

Display average payload mass carried by booster version F9 v1.1

```
%sql SELECT AVG("Payload_Mass__kg_") AS Average_Payload_Mass FROM SPACEXTBL WHERE "Booster_Version" = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
```

Average_Payload_Mass
2928.4

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql SELECT MIN("Date") AS First_Successful_Groundpad_Landing FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

First_Successful_Groundpad_Landing
2015-12-22

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (drone ship)' AND "Payload_Mass__kg_" > 4000 AND "Payload_Mass__kg_
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT "Mission_Outcome", COUNT(*) AS Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
```

```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	Total
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTBL WHERE "Payload_Mass__kg_" = (SELECT MAX("Payload_Mass__kg_") FROM SPACEXTBL);
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

Task 9

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql SELECT substr("Date", 6, 2) AS Month, "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL WHERE substr("Date", 0, 5) = '2015' AND "Landi
```

```
 * sqlite:///my_data1.db
Done.
```

Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Task 10

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

```
%sql SELECT "Landing_Outcome", COUNT(*) AS Outcome_Count FROM SPACEXTBL WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER
```

```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Mapping Launch Sites

Title: SpaceX Launch Site Locations

Slide Content:

- We used Folium, a powerful Python mapping library, to build an interactive map that visualizes all known SpaceX launch sites.
- Each launch site is represented with a custom marker.
- When a user clicks a marker, it displays the launch site name as a popup.
- This visualization helps us understand:
  - The geographic spread of launch sites (e.g., Florida, California).
  - Their proximity to coastlines and NASA centers.
- Folium's map interactivity (zoom, pan) allows dynamic exploration of SpaceX's spatial logistics.

Launch Outcomes Visualization

Title: Visualizing Launch Success and Failure by Site

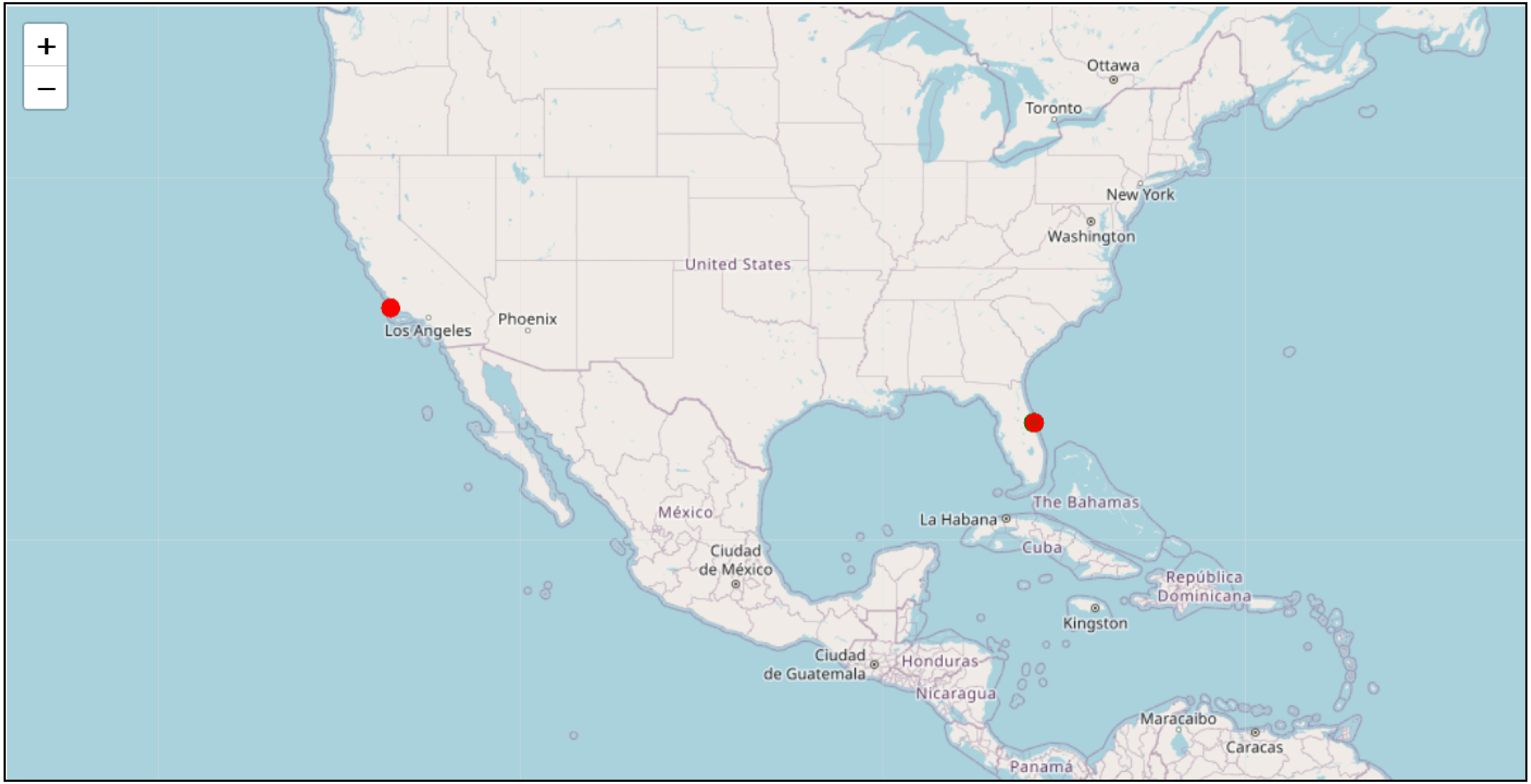
Slide Content:

- We extended the interactive map by adding colored markers representing launch outcomes:
  - 🟢 Green markers for successful launches
  - 🔴 Red markers for failed launches
- Each marker corresponds to a specific launch event and is placed at the launch site's coordinates.
- This view allows us to:
  - Compare performance across sites
  - Identify which launch sites have high success rates
  - Spot patterns of failure at particular locations
- Example: KSC LC 39A has mostly green markers, indicating high reliability.

Tools Used:

- folium.Marker() with conditional color logic
- folium.Popup() to show success/failure details
- MarkerCluster for managing overlapping points

HERE'S A EXAMPLE PICTURE:



Evaluating performance of each classification model using confusion matrices and accuracy scores.

Model	Accuracy Score
Random Forest	83.33%
Logistic Regression	83.33%
Decision Tree	83.33%
SVM	77.78%

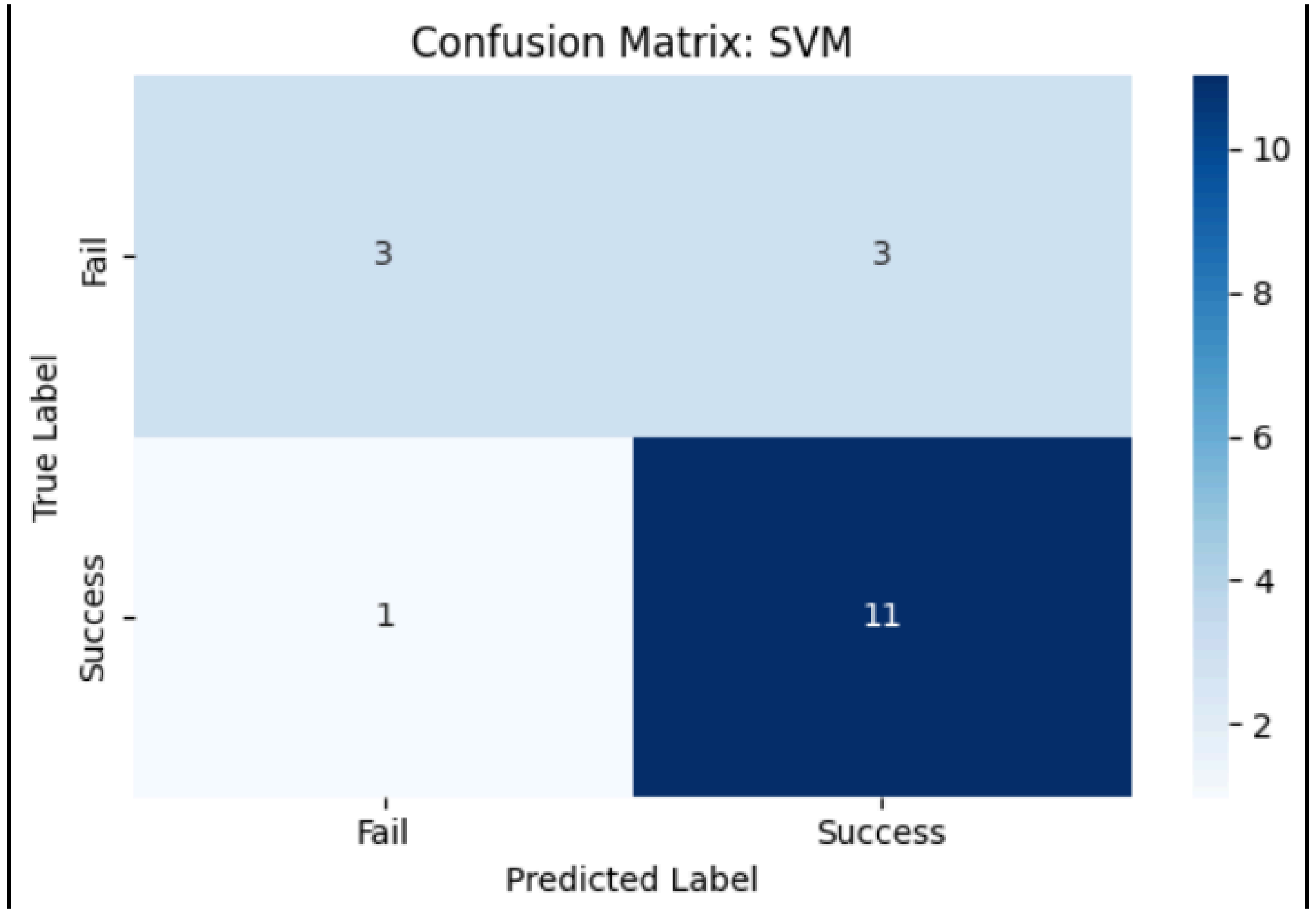
Insights:

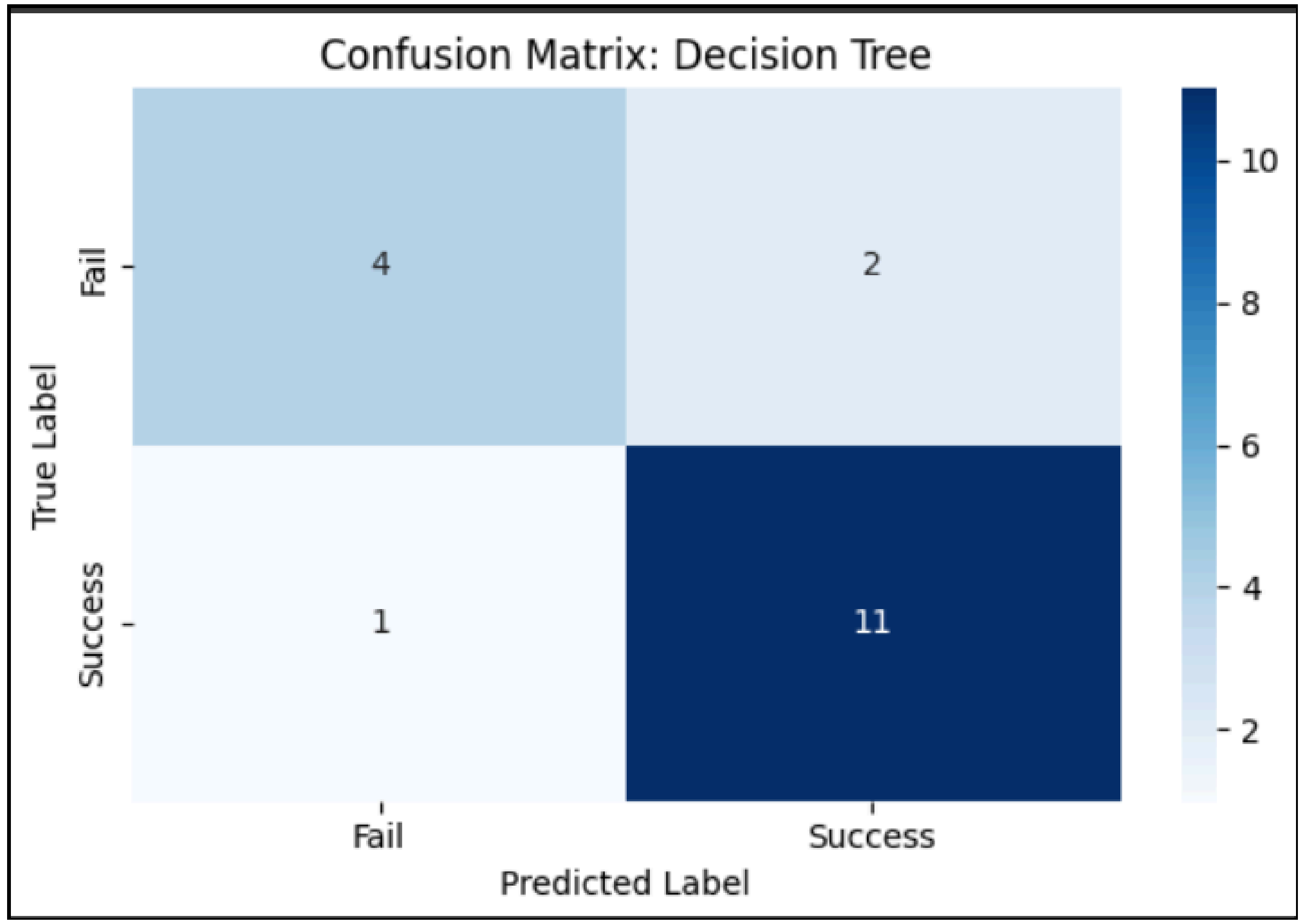
- Random Forest and Logistic Regression both performed best, with 83.33% accuracy.
- Decision Tree also achieved the same accuracy but had a slight misclassification.
- SVM underperformed slightly due to more false predictions.

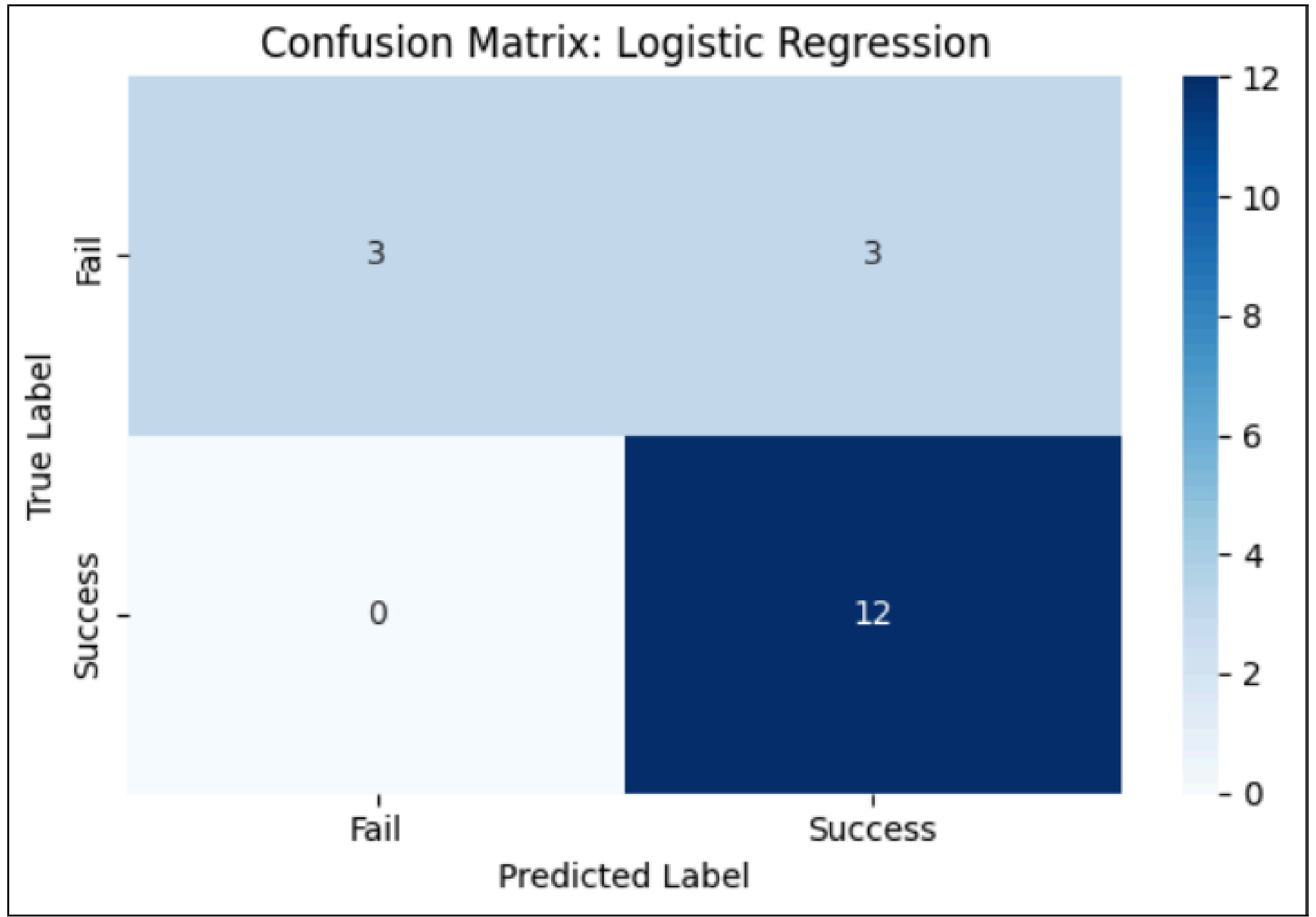
🧠 Conclusion:

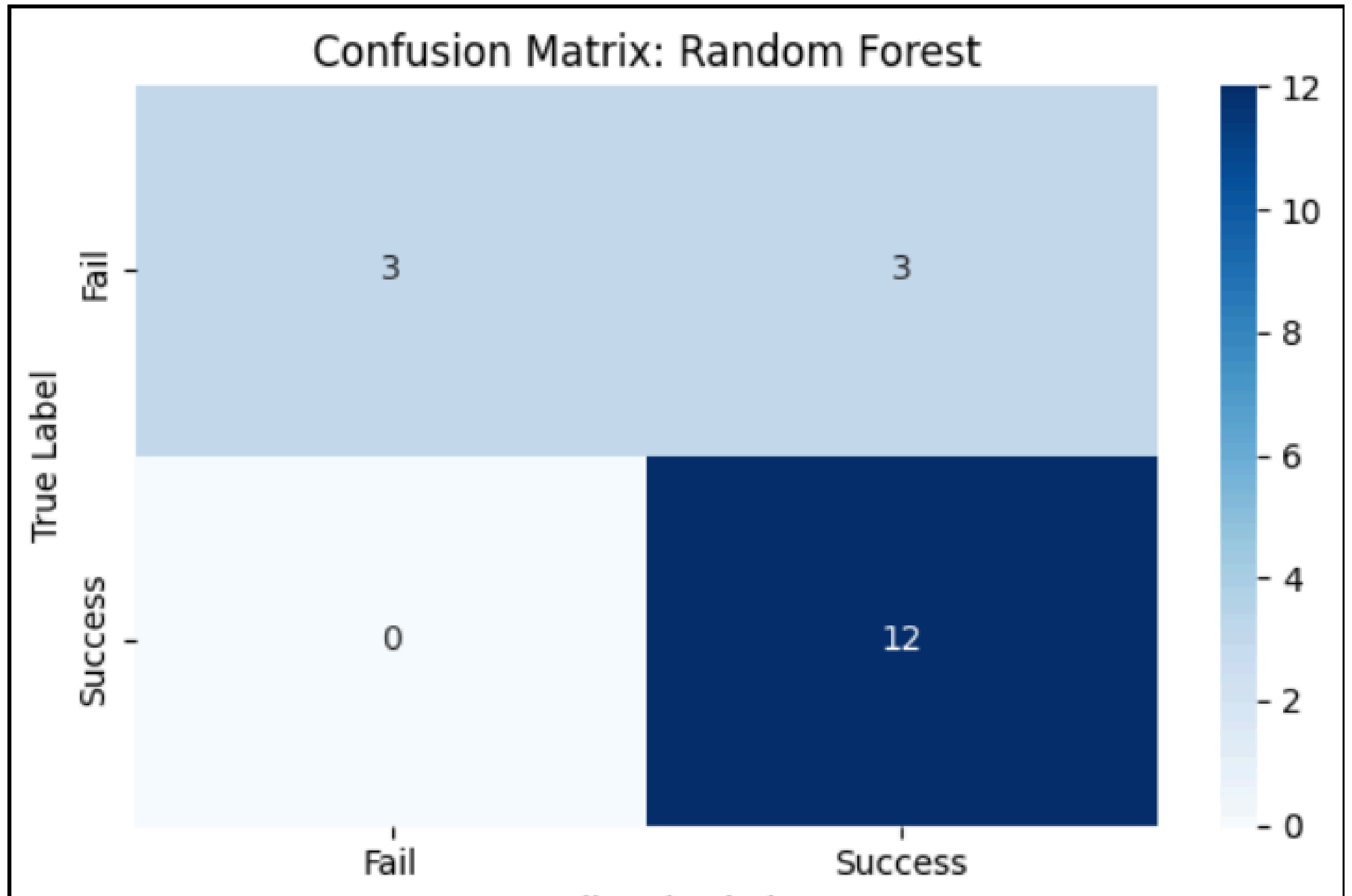
Random Forest and Logistic Regression are both reliable for predicting SpaceX launch outcomes. For critical decision support, ensemble methods like Random Forest may offer more robustness.

MODEL EVALUATION RESULTS









CONCLUSION

- Through exploratory data analysis, we uncovered key insights into SpaceX's launch performance across sites, payloads, and orbit types.
- SQL queries helped us extract mission-critical statistics directly from the structured database.
- Interactive visualizations using Folium and Plotly enhanced our understanding of launch geography and mission outcomes.
- Machine learning models such as Logistic Regression, SVM, Decision Trees, and Random Forest were applied to predict mission success — with Random Forest showing the highest accuracy.
- These insights can support data-driven decision making for future launches, improving mission reliability and cost efficiency.