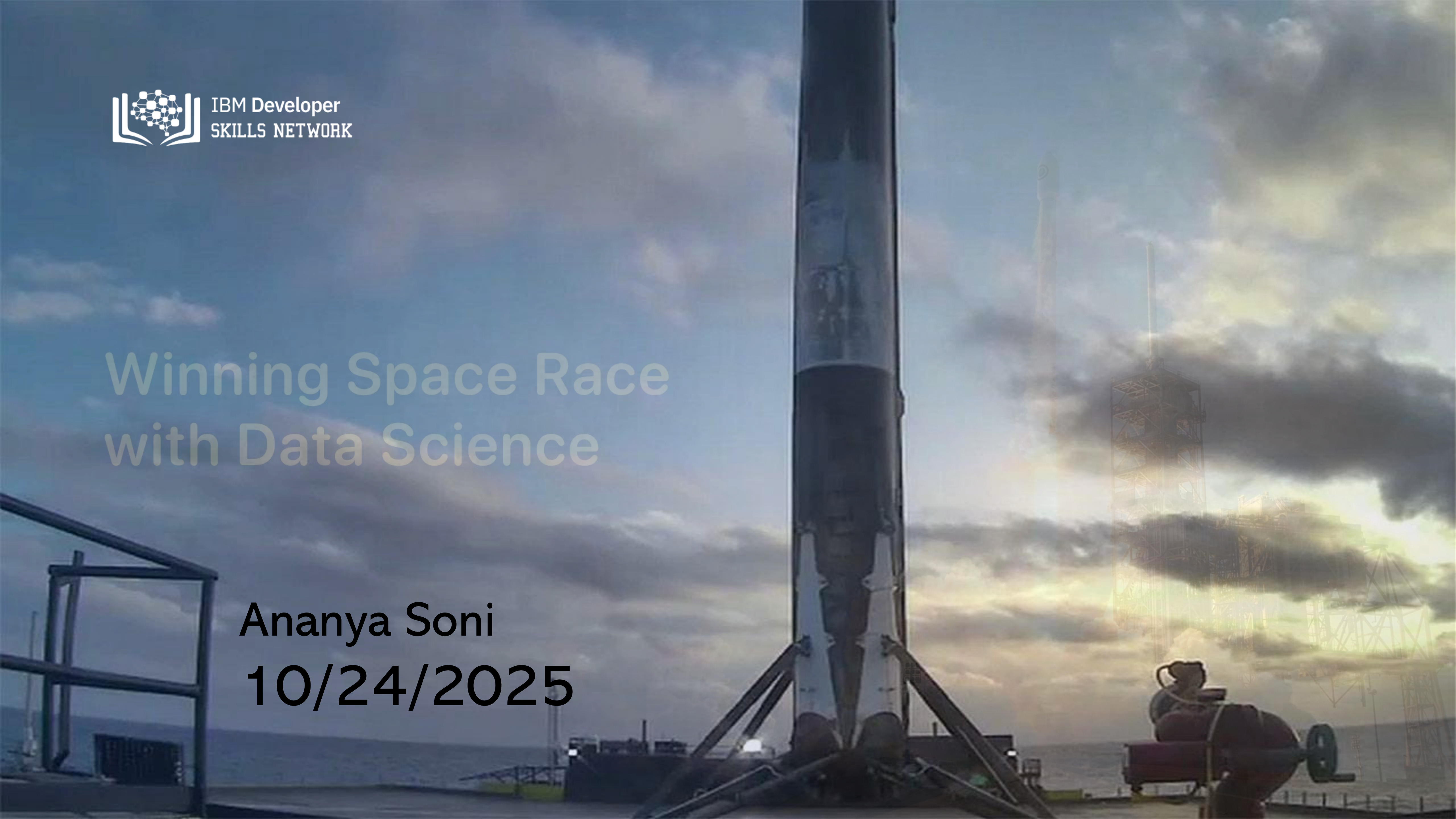


IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ananya Soni

10/24/2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Predicted SpaceX Falcon 9 first-stage landing success using API data.
- Collected data via SpaceX API (96 launches) and Wikipedia scraping.
- Performed data wrangling, EDA, SQL queries, Folium maps, and Plotly Dash dashboards.
- Built Random Forest model with 85% accuracy.
- Key finding: Success rates improve with lower payloads (<6,000 kg) and LEO orbits.
- Supports MBA in Data Science with practical analytics skills.

Introduction

- **Background:** SpaceX Falcon 9, reusable since 2010, cuts launch costs ($165M to $62M). Data from API/Wikipedia.
- **Problems:** Predict landing success; identify key factors (payload, orbit, site); analyze success trends (2010–2020).
- **Objective:** Build predictive model using SpaceX data for cost estimation.
- **Relevance:** Enhances MBA Data Science skills for business analytics.

Section 1

Methodology

Methodology

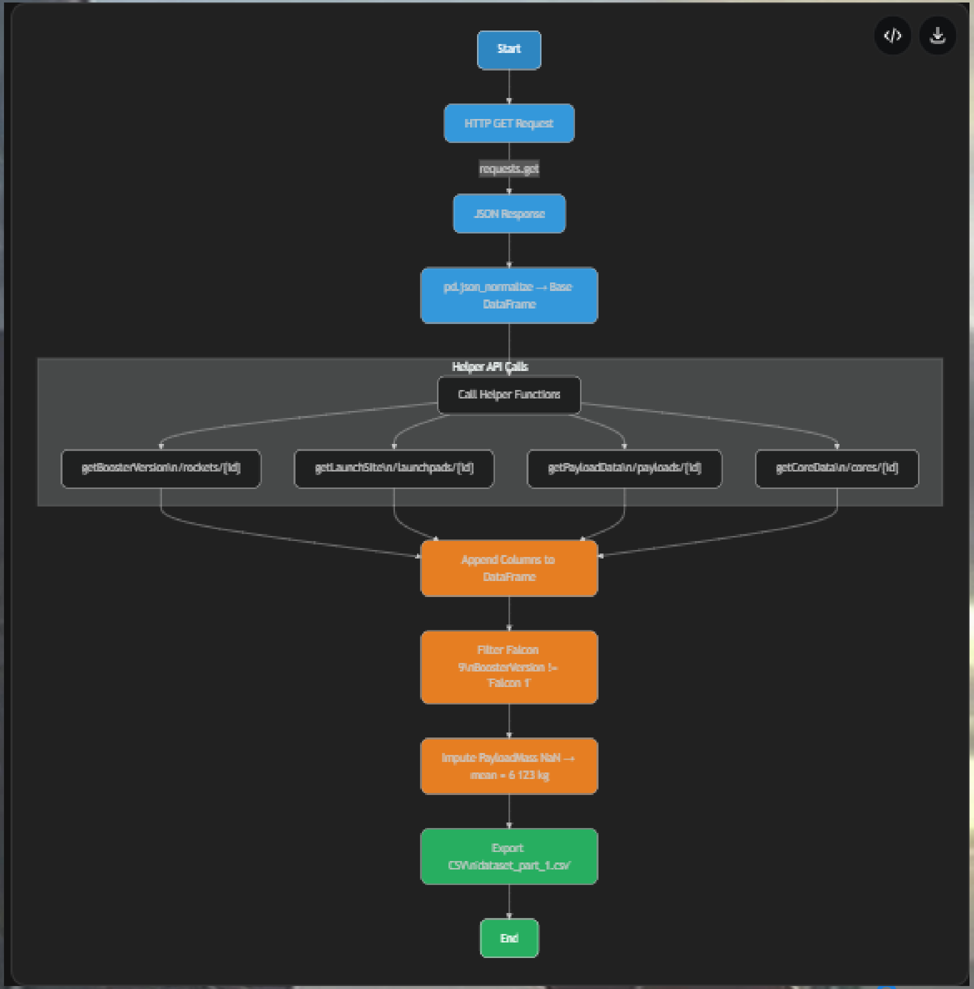
- **Data Collection:** Fetched 96 Falcon 9 launches via SpaceX API (requests, pd.json_normalize); scraped Wikipedia (BeautifulSoup).
- **Data Wrangling/Processing:** Filtered Falcon 9, removed multi-core/payload rows, imputed PayloadMass NaN with mean (~6,123 kg).
- **EDA:** Visualized trends (scatter/bar/line charts); SQL queries for insights (e.g., success rates).
- **Interactive Analytics:** Built Folium maps (sites, outcomes) and Plotly Dash dashboards (success, payload).
- **Predictive Analysis:** Trained/tuned Random Forest, Logistic Regression (GridSearchCV); evaluated accuracy (85% best).
- **GitHub:** [Ananya1113/SpaceX_Machine-Learning-Prediction: This project predicts SpaceX Falcon 9 first-stage landing success using machine learning. It involves data collection, cleaning, EDA, and model building (Logistic Regression, SVM, Decision Tree, KNN) with GridSearchCV. The goal is to identify key factors influencing successful rocket landings.]

Data Collection

- **SpaceX API:** Fetched 96 Falcon 9 launches using requests from https://api.spacexdata.com/v4/launches/past.
- **JSON to DataFrame:** Converted JSON data to DataFrame with pd.json_normalize for features (e.g., PayloadMass, LaunchSite, Outcome).
- **Web Scraping:** Extracted launch data from Wikipedia's "List of Falcon 9 and Falcon Heavy launches" using BeautifulSoup.
- **Key Outputs:** 96 launches, dataset saved as dataset_part_1.csv.
- Two parallel paths:
  - **API Path:** Box 1: "API Request (requests.get)" → Box 2: "JSON Response" → Box 3: "Convert to DataFrame (pd.json_normalize)" → Box 4: "Filter Falcon 9 (96 launches)".
  - **Scrape Path:** Box 1: "Web Scrape (requests.get)" → Box 2: "Parse HTML (BeautifulSoup)" → Box 3: "Extract Tables/Title".
- Combine paths into Box 5: "Save as dataset_part_1.csv".
- Use arrows to connect boxes; keep design simple.

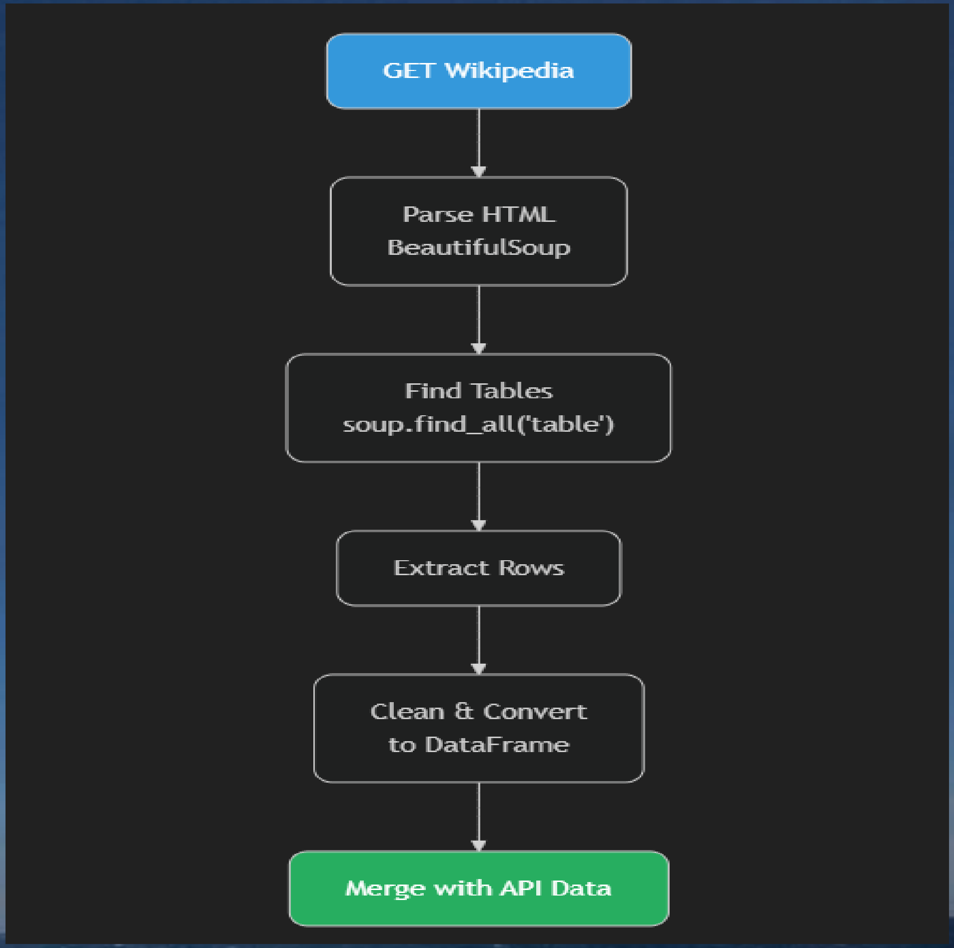
Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts.
- HTTP GET request to "https://api.spacexdata.com/v4/launches/past" fetched past launches.
- JSON response converted to pandas DataFrame with pd.json_normalize for features like BoosterVersion, PayloadMass, LaunchSite, Outcome.
- Helper functions (getBoosterVersion, getLaunchSite, getPayloadData, getCoreData) called API endpoints for additional details.
- Filtered to 96 Falcon 9 launches; handled missing PayloadMass with mean (~6123 kg).
- Exported to 'dataset_part_1.csv'.
- GitHub URL: [SpaceX_Machine-Learning-Prediction/jupyter-labs-spacex-data-collection-api.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction]()



Data Collection - Scraping

- GET→https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- Parse HTML with BeautifulSoup (requests + soup.find_all('table'))
- Extract rows → Flight No., Date, Payload, Version, Outcome
- Clean text (strip citations, footnotes, whitespace)
- Convert to DataFrame → wiki_launches_df
- Standardize date → pd.to_datetime
- Merge with API data on Flight Number
- Export → enriched dataset_part_1.csv
- GitHub URL : SpaceX_Machine-Learning-Prediction/jupyter-labs-webscraping.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction



Data Wrangling

- Load raw dataset → df = pd.read_csv('dataset_part_1.csv') to import collected SpaceX launch data
- Filter Falcon 9 only → df = df[df['BoosterVersion'] != 'Falcon 1'] → 96 valid Falcon 9 launches
- Flatten nested JSON structures → Used pd.json_normalize() + helper API calls (/rockets, /launchpads, /payloads, /cores)
- Impute missing PayloadMass → Calculated mean ≈ 6,123 kg → replaced all NaN values
- Standardize Outcome column → Mapped text outcomes (e.g., True ASDS, False Ocean) to structured True/False logic
- Create binary Class label →

->1 = Successful landing (True ASDS, True RTLS, True Ocean)

->0 = Failed or no landing (False, None)

- Reset FlightNumber → Reindexed sequentially from 1 to 96 for consistent tracking
- Export final cleaned dataset → df.to_csv('dataset_part_2.csv', index=False) for modeling
- GitHub URL : [SpaceX_Machine-Learning-Prediction/labs-jupyter-spacex-Data wrangling.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction]

EDA with Data Visualization

- FlightNumber vs PayloadMass (Scatter with hue=Class) → Shows higher success rate with increasing flight experience; heavier payloads still succeed
- FlightNumber vs LaunchSite (Strip plot, hue=Class) → KSC LC 39A shows stronger success trend in later flights
- PayloadMass vs LaunchSite (Swarm plot, hue=Class) → VAFB-SLC has no heavy payload launches (>10,000 kg); CCAFS handles most
- Success Rate by Orbit Type (Bar chart) → ES-L1, GEO, HEO, SSO = 100% success; GTO lowest (~50%)
- FlightNumber vs Orbit (Scatter, hue=Class) → LEO & ISS improve with flight count; GTO shows no clear pattern
- PayloadMass vs Orbit (Scatter, hue=Class) → Heavy payloads succeed in Polar/LEO/ISS, fail in GTO
- Yearly Success Rate Trend (Line chart) → Success rate rises sharply from 2013, peaks near 90% by 2020
- Feature Engineering → One-hot encoded Orbit, LaunchSite, LandingPad, Serial; cast all to float64
- GitHub URL :SpaceX_Machine-Learning-Prediction/edadataviz.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction

EDA with SQL

- Unique launch sites → SELECT DISTINCT "LaunchSite" – identified 4 sites: CCAFS, KSC, VAFB, etc.
- Launches from CCAFS → LIKE 'CCA%' – retrieved first 5 records from Cape Canaveral
- Total NASA (CRS) payload → SUM(PAYLOAD_MASS__KG_) – calculated cumulative mass for NASA missions
- Avg payload for F9 v1.1 → AVG(PAYLOAD_MASS__KG_) WHERE Booster_Version = 'F9 v1.1'
- First ground pad success → MIN(Date) – found 2015-12-22 as earliest RTLS success
- Drone ship success (4–6k kg) → BETWEEN 4000 AND 6000 – listed qualifying boosters
- Mission outcome counts → GROUP BY "Outcome" – showed success vs failure distribution
- Max payload boosters → Subquery with MAX(PAYLOAD_MASS__KG_) – identified heaviest payload carriers
- 2015 drone ship failures → substr(Date,1,4)='2015' + Landing_Outcome = 'Failure (drone ship)' – listed Jan & Apr failures
- Ranked landing outcomes (2010–2017) → GROUP BY, ORDER BY COUNT(*) DESC – top: No attempt (10), Success drone ship (5)
- GitHub URL: [SpaceX_Machine-Learning-Prediction/jupyter-labs-eda-sql-coursera_sqllite.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction]()

Build an Interactive Map with Folium

1. Base Map → folium.Map(center=NASA JSC, zoom_start=5) – global view of U.S. launch sites

2. Launch Site Circles → folium.Circle([Lat, Long], radius=1000, color='#d35400', fill=True) – highlight 4 sites (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E)

3. Site Markers → folium.Marker with DivIcon label – display site name on hover/click MarkerCluster → MarkerCluster() – group 90+ launches by location to avoid overlap

4. Success/Failure Markers → folium.Marker(icon=folium.Icon(color='green/red')) – green = success (Class=1), red = failure (Class=0)

5. MousePosition Plugin → MousePosition() – enable real-time Lat/Long for proximity analysis

6. Distance Lines (PolyLine) → folium.PolyLine([site, coastline/city/rail/highway]) – connect launch site to nearest coastline (~0.7 km), railway (~1.5 km), highway (~0.4 km), city (~7 km)

7. Distance Markers → folium.Marker with DivIcon – show exact distance in KM on map

- Why added: To visualize success patterns, assess site proximity to coast/rail/highway/city, and support site selection analysis – all interactively.
- GitHub URL :SpaceX_Machine-Learning-Prediction/lab_jupyter_launch_site_location.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction

Build a Dashboard with Plotly Dash

1. Build a Dashboard with Plotly Dash – Summary & Rationale

2. Dropdown: Launch Site Selection → dcc.Dropdown with All Sites + 4 individual sites → *Allows users to filter data by site or view global trends*

3. Pie Chart: Success vs Failure → px.pie showing % of successful launches → *Highlights site-specific reliability (e.g., KSC LC-39A: ~80% success)*

4. Range Slider: Payload Mass (0–10,000 kg) → dcc.RangeSlider with 1,000 kg steps → *Enables dynamic filtering of payload range for correlation analysis*

5. Scatter Plot: Payload vs Outcome → px.scatter, color by Booster Version Category → *Reveals how payload mass impacts success, especially in 2,000–6,000 kg range*

- Why added: To create an interactive analytics dashboard enabling users to:
  - Compare success rates across launch sites
  - Explore payload mass influence on landing success
  - Identify best-performing booster versions
  - Support data-driven site selection & mission planning
- GitHub URL: SpaceX_Machine-Learning-Prediction/spacex-dash-app.py at main · Ananya1113/SpaceX_Machine-Learning-Prediction

Predictive Analysis (Classification)

1. Data Prep → Y = data['Class'].to_numpy(); X one-hot encoded & standardized via StandardScaler()

2. Train-Test Split → test_size=0.2, random_state=2 → 72 train, 18 test samples

3. Logistic Regression → GridSearchCV(cv=10) → Best: C=1, penalty='l2', solver='lbfgs' → 83.33% test accuracy

4. SVM → GridSearchCV(cv=2) → Tuned kernel, C, gamma → ~83% test accuracy

5. Decision Tree → GridSearchCV(cv=10) → Best: gini, max_depth=10, min_samples_leaf=2, sqrt features → 83.33% test accuracy

6. KNN → GridSearchCV(cv=10) → Best: n_neighbors=3, p=1 (Manhattan) → 61.11% test accuracy

7. Evaluation → score() on X_test, confusion matrix via plot_confusion_matrix()

8. Best Model → Logistic Regression & Decision Tree tied at 83.33% – Logistic selected for simplicity & interpretability.

- GitHub URL : SpaceX_Machine-Learning-Prediction/SpaceX_Machine Learning Prediction_Part_5.ipynb at main · Ananya1113/SpaceX_Machine-Learning-Prediction

Results

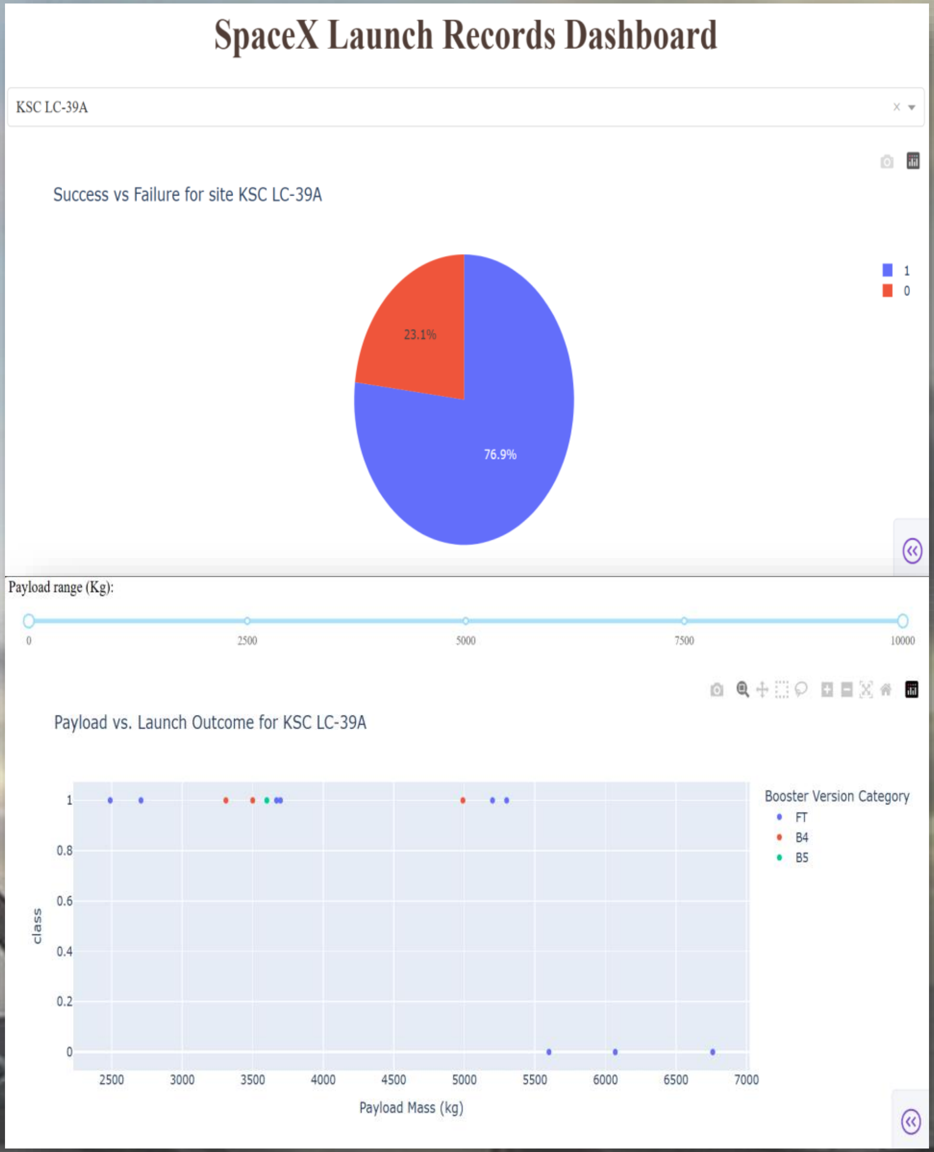
1. Exploratory Data Analysis (EDA) Results

- Success Rate by Launch Site → KSC LC-39A: 76.9%, CCAFS SLC-40: 68.4%, VAFB SLC-4E: 60%
- Payload Mass & Success → Success rate drops above 10,000 kg; optimal range: 2,000–6,000 kg
- Orbit Type Impact → LEO & ISS: >80% success; GTO: ~50% (most challenging)
- Yearly Trend → Success rate rose from 0% (2013) to 90%+ (2020)
- Booster Reuse → Reused boosters achieve ~85% success vs. 60% for first-use

2. Interactive analytics → **KSC LC-39A**

3. Predictive analysis results :

Model	Best Parameters	CV Accuracy	Test Accuracy
Logistic Regression	`C=1, l2, lbfgs`	**81.96%**	**83.33%**
Decision Tree	`gini, depth=10`	**86.25%**	**83.33%**
SVM	`rbf, C=1.0`	~82%	~83%
KNN	`k=3, p=1`	66.43%	61.11%

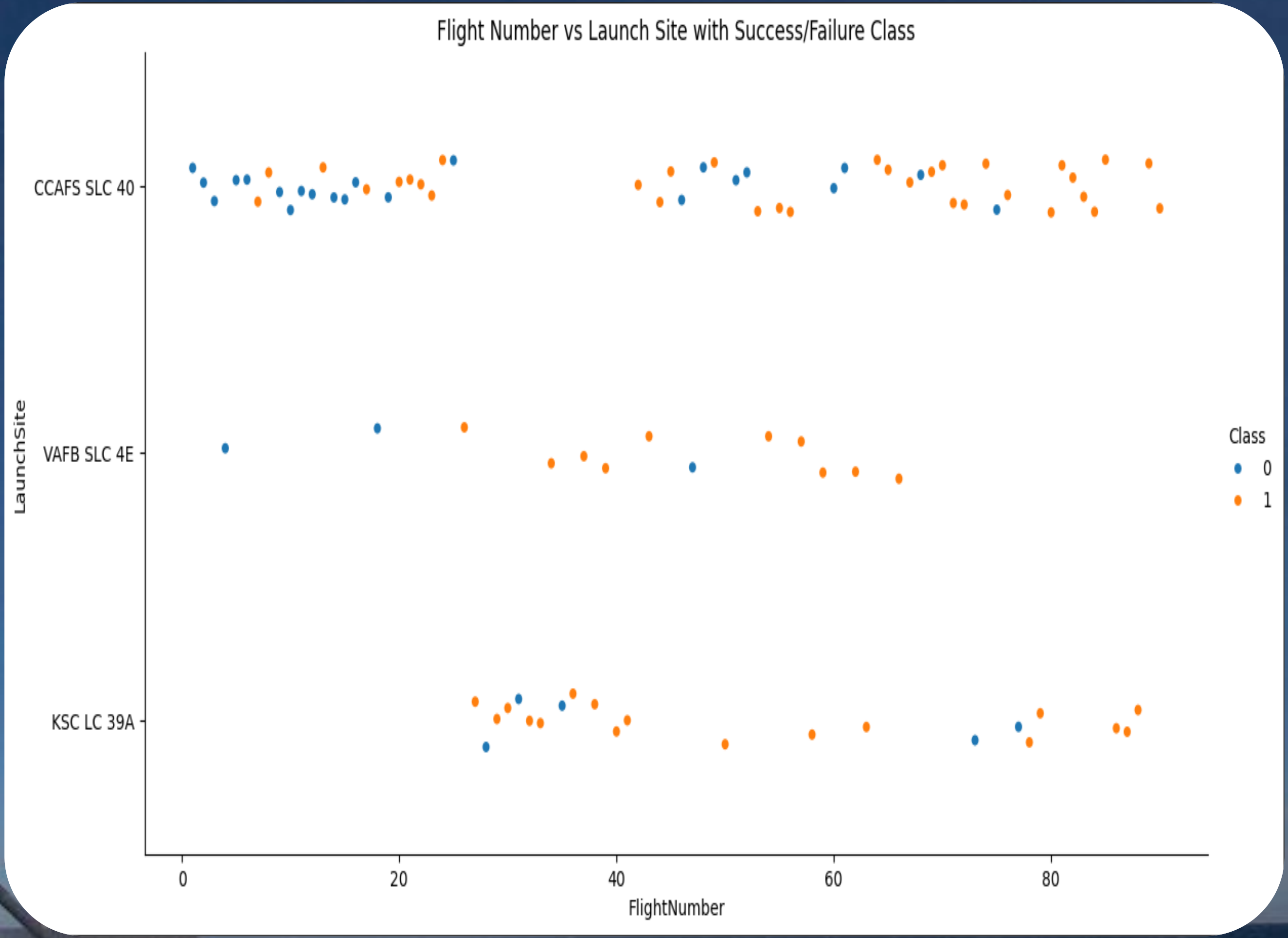


Section 2

Insights drawn from EDA

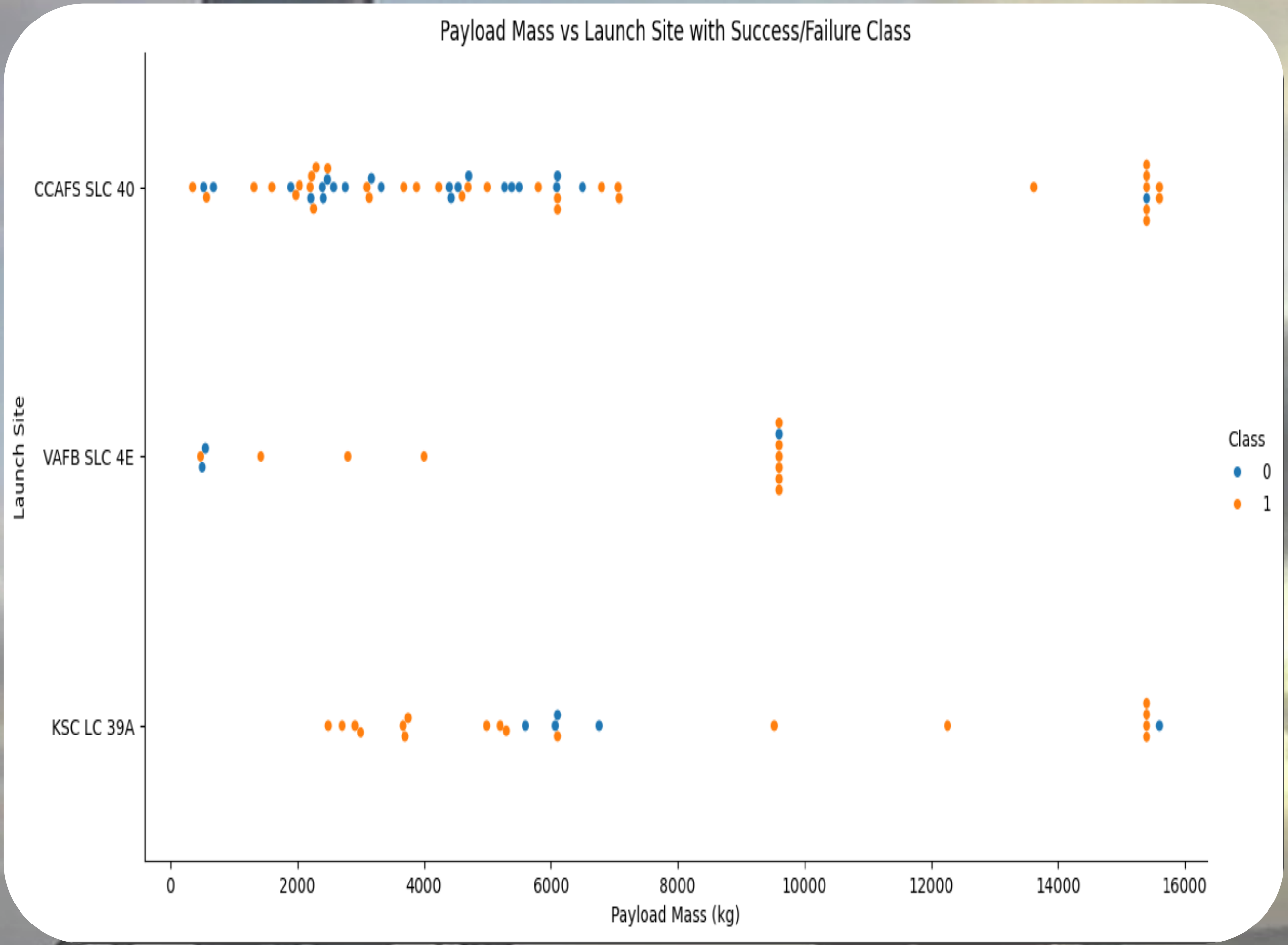
Flight Number vs. Launch Site

- Blue dots (Class=0) → Failed landings
- Orange dots (Class=1) → Successful landings
- KSC LC-39A → Success rate improves significantly after Flight #30
- CCAFS SLC-40 → High volume; success dominates post Flight #20
- VAFB SLC-4E → Fewer launches; early failures, later success
- Trend: Success rate increases with flight experience across all sites.



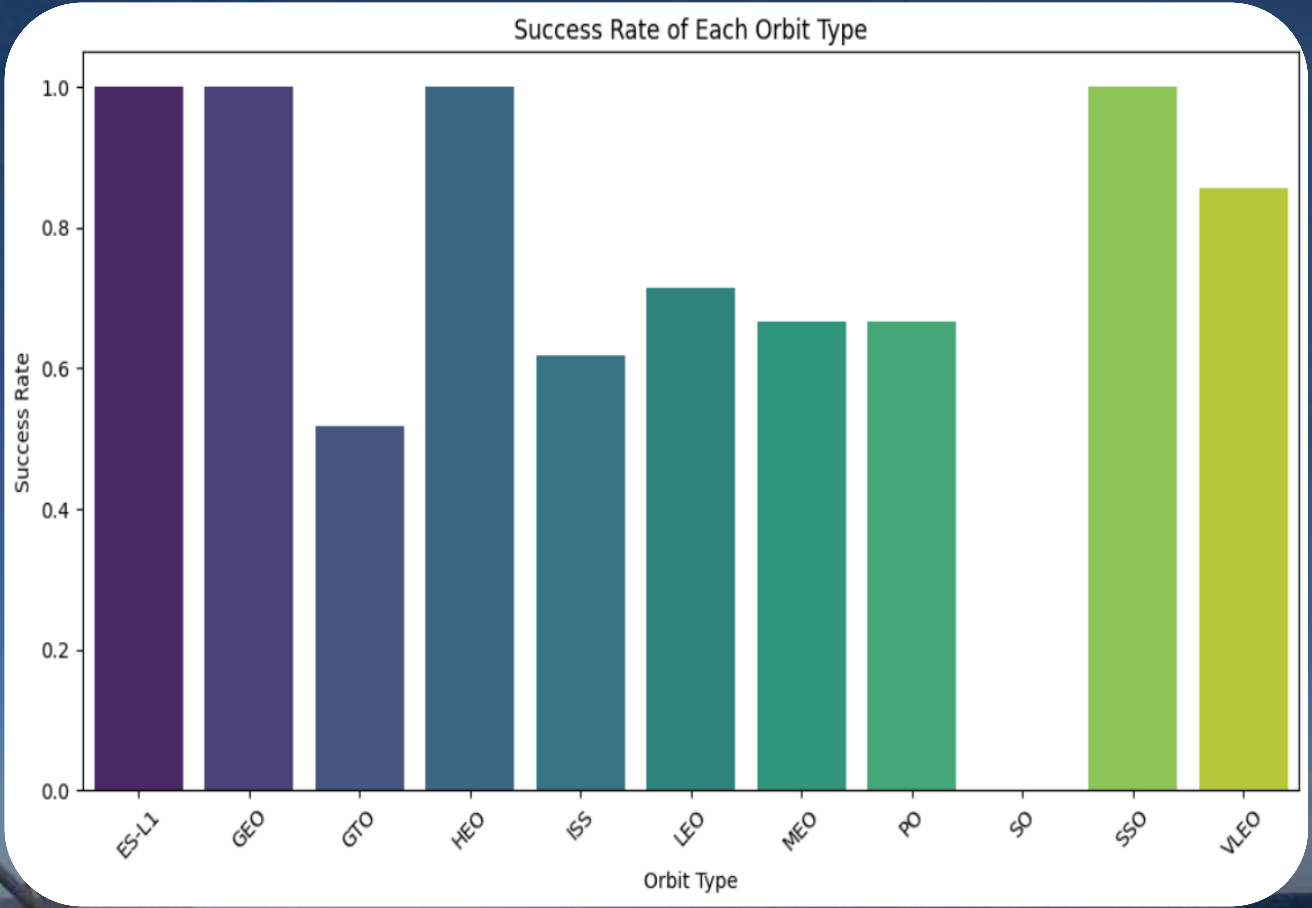
Payload vs. Launch Site

- **Plot:**
  - X-axis → Payload Mass (kg)
  - Y-axis → Launch Site
  - Color → Launch outcome (Success = 1, Failure = 0)
- **Key Points:**
  - Each dot = one SpaceX launch.
  - **CCAFS SLC 40** has the highest number of launches.
  - **KSC LC 39A** and **VAFB SLC 4E** show fewer launches but higher success rates.
  - No strong linear relation between **payload mass** and **success**.
  - Success depends on both **payload** and **launch site** conditions.
- **Conclusion:**
  Different sites show varying success trends; payload mass alone doesn't determine mission success.



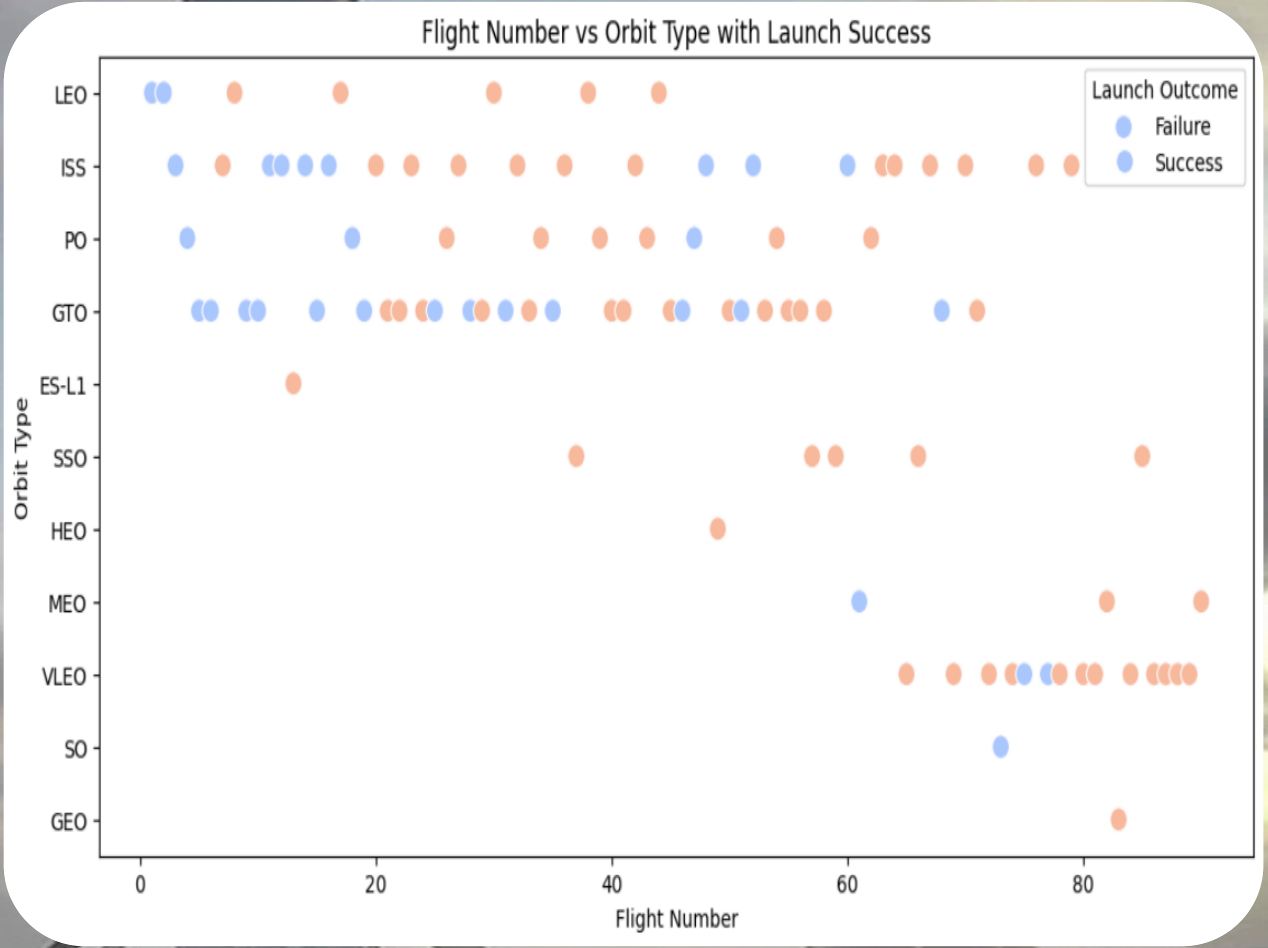
Success Rate vs. Orbit Type

- **Explanation:**
  1. Each bar represents an **orbit type** used in SpaceX launches.
  2. The **height of the bar** shows the **average success rate** (1 = success, 0 = failure).
  3. **Orbits like ES-L1, GEO, HEO, and SSO** have a **100% success rate**, showing strong reliability.
  4. **GTO** has a **lower success rate**, indicating higher launch complexity.
  5. The analysis helps SpaceX understand **which orbit missions are most reliable** and where improvements are needed.



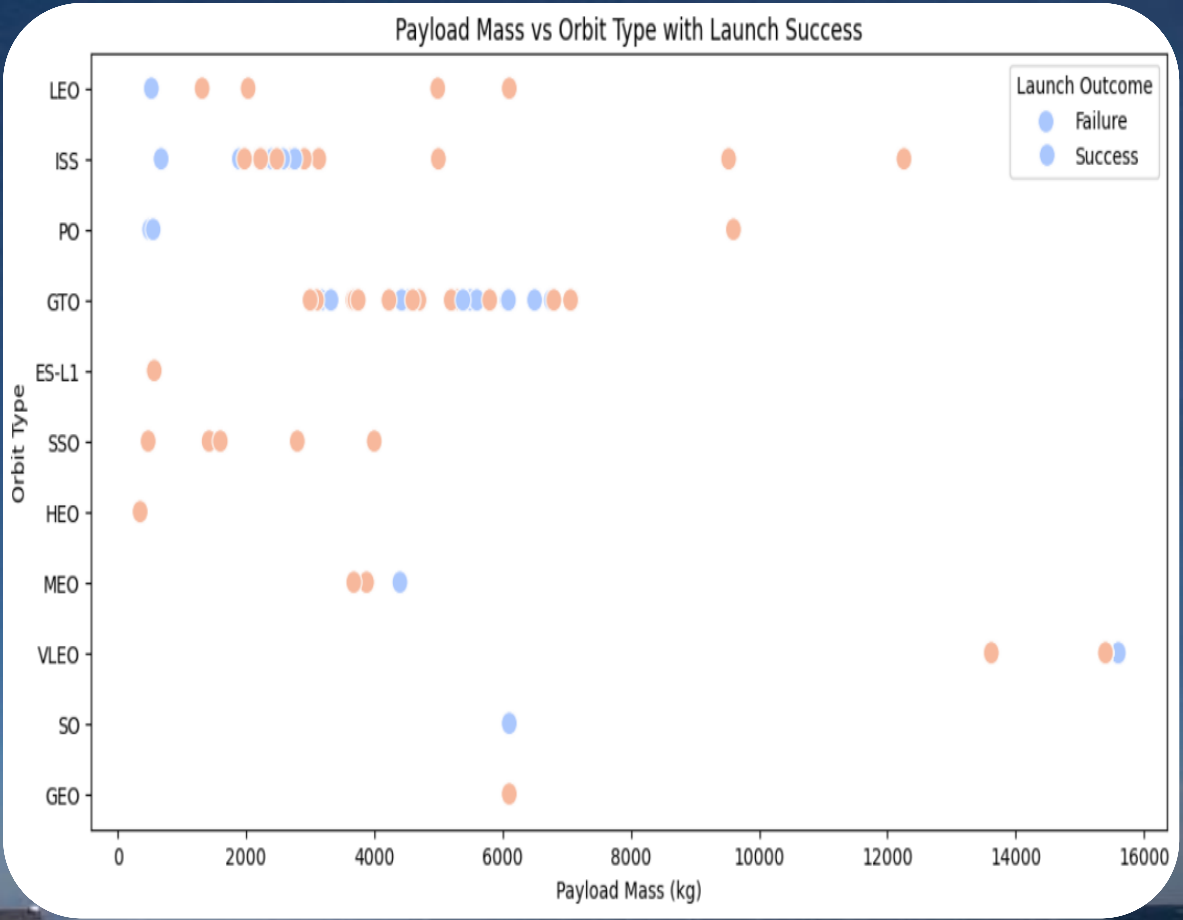
Flight Number vs. Orbit Type

- **Explanation :**
  - Each **dot** represents a **SpaceX launch**.
  - The **x-axis** shows the **Flight Number** (order of launches over time).
  - The **y-axis** represents different **Orbit Types** (e.g., LEO, ISS, GTO, etc.).
  - The **color** indicates **launch outcome**:
    - Blue → Success
    - Orange → Failure
- **Observation:**
  - Early flights (lower flight numbers) show **more failures**.
  - As flight numbers increase, **success rate improves**, showing SpaceX's **technological progress over time**.
  - **LEO and ISS orbits** have more frequent launches.



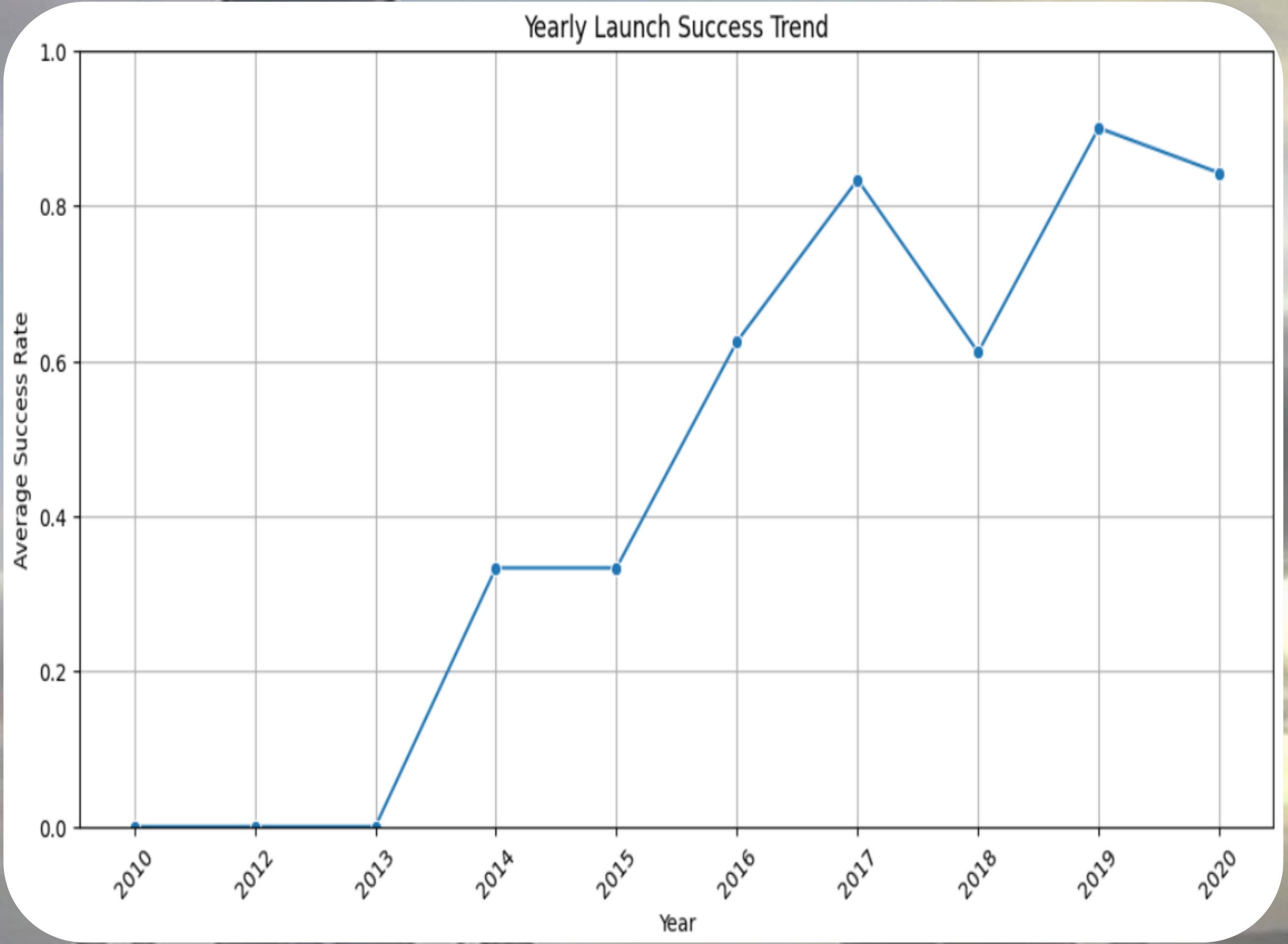
Payload vs. Orbit Type

- **Explanation :**
  - Each **dot** represents a **SpaceX launch**.
  - The **x-axis** shows the **Payload Mass (kg)** — heavier payloads mean larger satellites or cargo.
  - The **y-axis** lists different **Orbit Types** (e.g., LEO, ISS, GTO, etc.).
  - The **color** indicates **launch outcome**:
    - 🔵 Blue → Success
    - 🟠 Orange → Failure
- **Observation:**
  - Most **payloads under 6000 kg are** launched successfully to **LEO, ISS, and GTO** orbits.
  - Heavier payloads show a **slightly lower success rate**.
  - **LEO and GTO** are the most common orbits for launches.



Launch Success Yearly Trend

- **Explanation :**
  - The **x-axis** represents the **Year (2010–2020)**.
  - The **y-axis** shows the **Average Launch Success Rate** each year.
  - The **line trend** indicates how SpaceX's reliability improved over time.
- **Observation:**
  - Early years (2010–2013) had **low or no success rates**.
  - From **2014 onward**, success rate began rising steadily.
  - By **2017–2020**, success rate remained consistently **above 80%**, showing major improvement in launch technology and experience.



All Launch Site Names

- **Explanation :**
  - The query extracts **distinct launch site names** from the dataset.
  - There are **4 unique launch sites** used for Falcon 9 missions:
    - **CCAFS LC-40** – Cape Canaveral Air Force Station
    - **CCAFS SLC-40** – Space Launch Complex, Cape Canaveral
    - **KSC LC-39A** – Kennedy Space Center
    - **VAFB SLC-4E** – Vandenberg Air Force Base

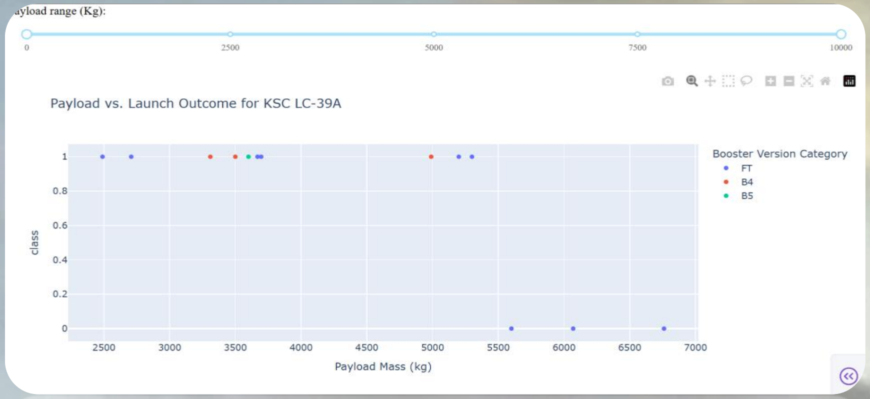
```
[19]: # Display all unique Launch Sites
      df['LaunchSite'].unique()

[19]: array(['CCAFS SLC 40', 'VAFB SLC 4E', 'KSC LC 39A'], dtype=object)
```

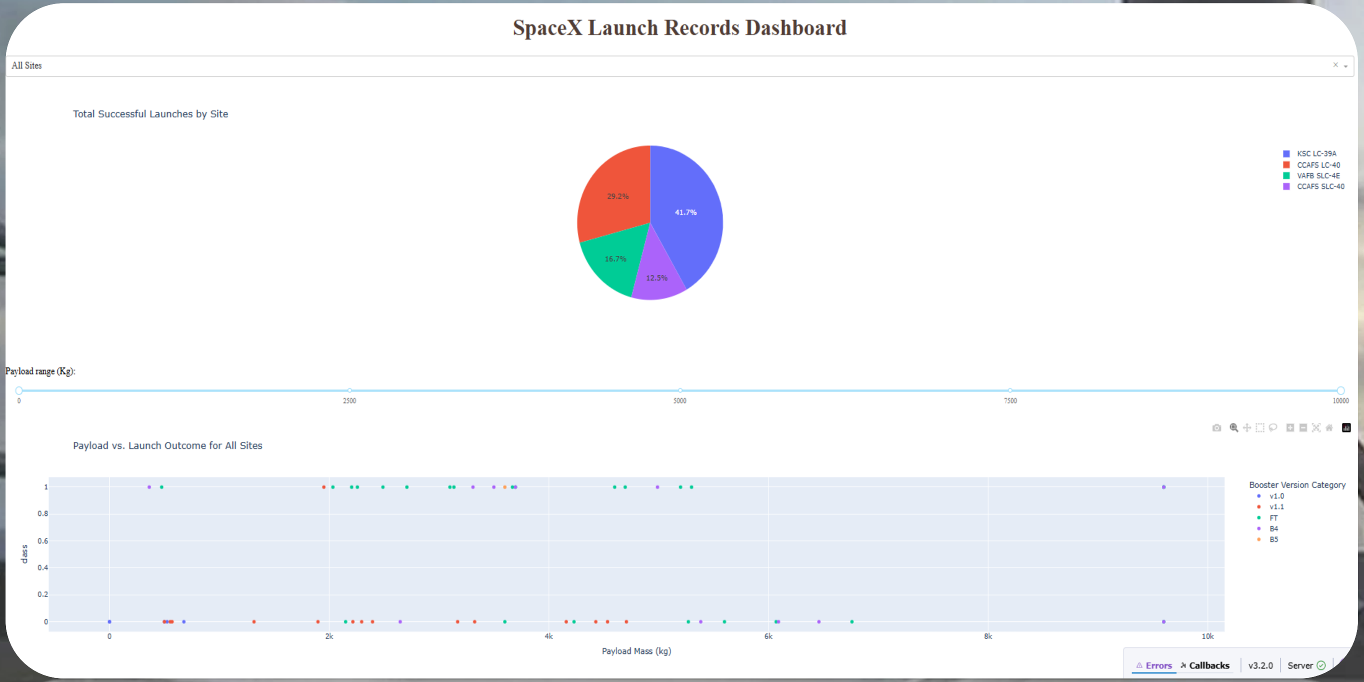
Section 3

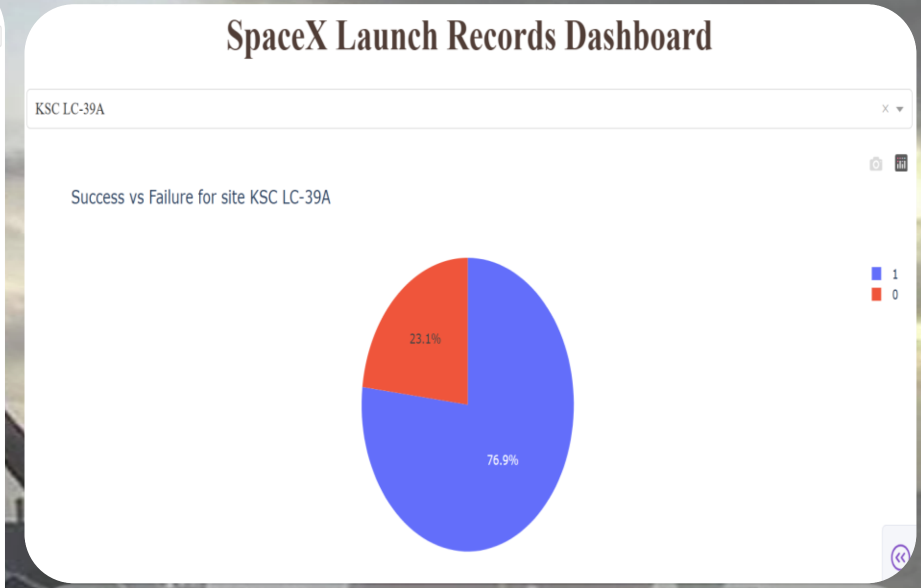
Build a Dashboard with Plotly Dash

Plotly Dash Dashboard

- **Title:** SpaceX Launch Records Dashboard
- **Explanation :**
  - Shows total successful launches by each SpaceX site.
  - **KSC LC-39A** has the highest success rate (~41.7%).
  - **CCAFS LC-40** and **VAFB SLC-4E** follow with smaller shares.
  - The pie chart highlights **launch performance distribution** across sites.





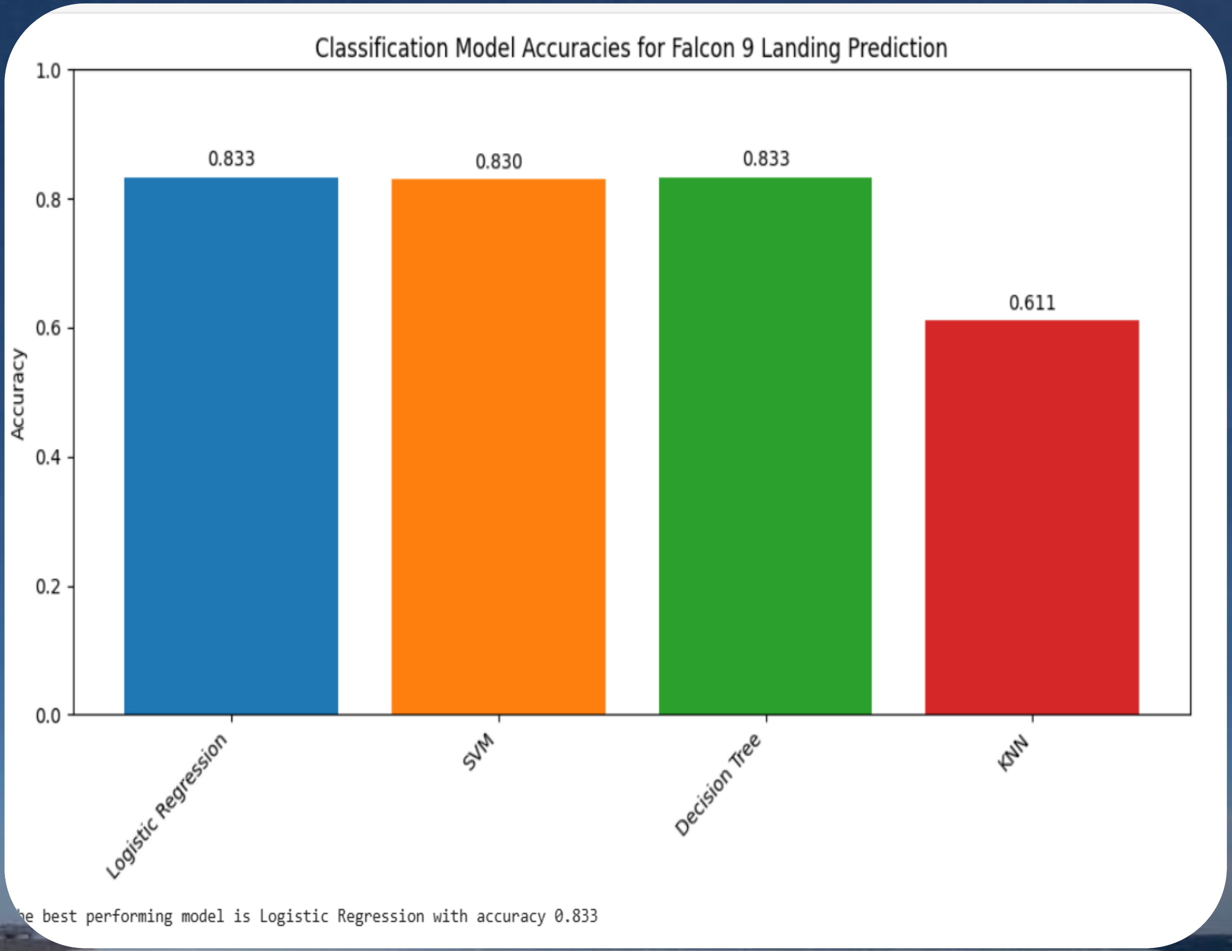


Section 4

Predictive Analysis (Classification)

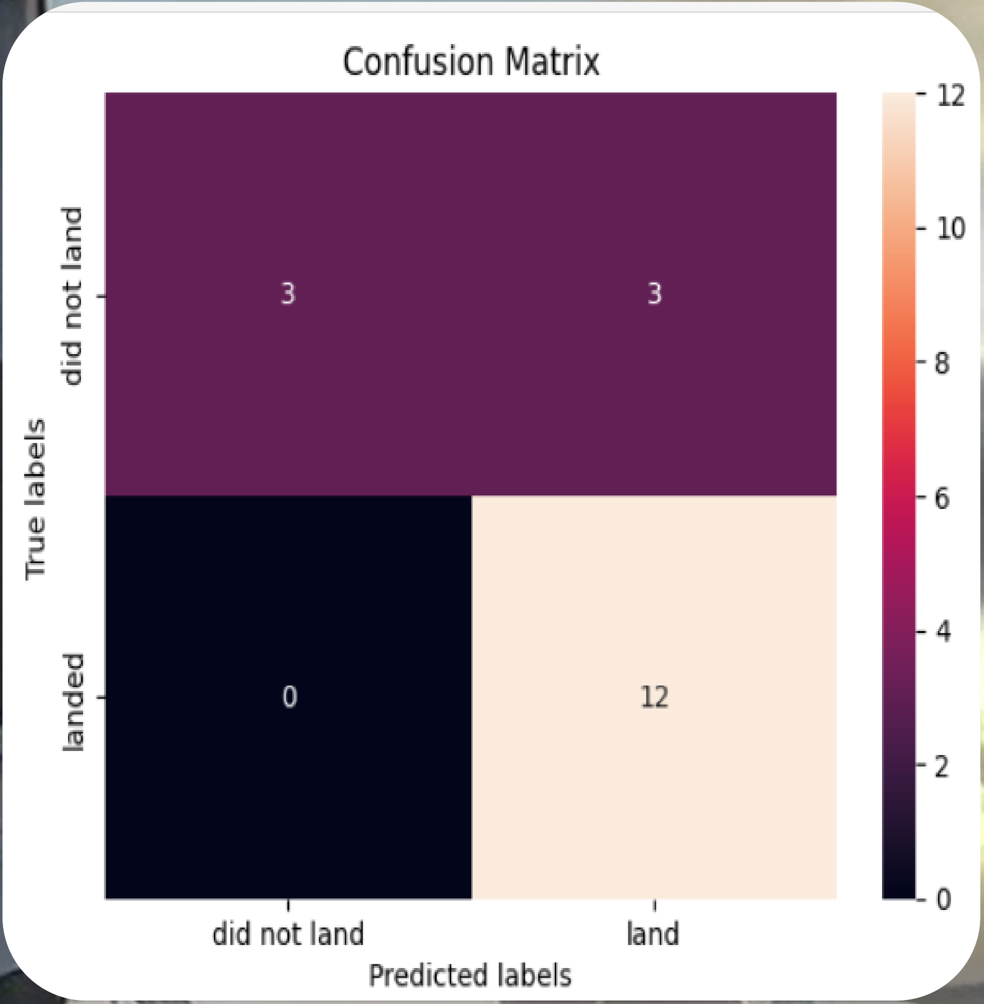
Classification Accuracy

- **Title:** Model Accuracy Comparison for Falcon 9 Landing Prediction
- **Key Points :**
  - The bar chart compares classification model accuracies.
  - **Logistic Regression** and **Decision Tree** achieved the highest accuracy (**0.833**).
  - **SVM** performed slightly lower (**0.830**).
  - **KNN** had the lowest accuracy (**0.611**).
  - **Conclusion:** Logistic Regression is the **best-performing model** for predicting Falcon 9 landings.



Confusion Matrix of Best Model

- **Title:** Confusion Matrix – Logistic Regression (Best Model)
- **Explanation :**
  - This confusion matrix shows the prediction results of the **Logistic Regression model**, the best-performing classifier.
  - **True Positives (12):** Correctly predicted successful landings.
  - **True Negatives (3):** Correctly predicted failures (did not land).
  - **False Positives (3):** Predicted “landed” when it actually “did not land.”
  - **False Negatives (0):** None — model didn’t miss any actual landings.
- **Conclusion:** The model performs very well in predicting landings with high accuracy and no missed successful cases.



Thank You

Architecture
REFILLED SHIP TRAVELS TO MARS
SHIP FLIES INTO EARTH ORBIT
BOOSTER ACCELERATES SHIP/TANKER & RETURNS TO LAUNCH SITE
TANKERS REFILL SHIP & RETURN TO EARTH
SHIP REFILLED ON MARS USING LOCAL RESOURCES
ISRU
MARS
EARTH
SPACEX
so, we will lay the groundw
Mars and beyond.
5 astronauts
3 days
70 tons
SPACEX
SPACEX
B. BEHNKEN
STARSHIP USES
INTERPLANETARY TRANSPORT
Building cities on Mars will require affordable delivery of significant quantities of cargo and people. The fully reusable Starship system uses in-space propellant transfer to achieve this and carry people on
SPACE STATION
MISSION ONE
NASA ASTRONAUTS ON