IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sree Matangi.K
16/10/2025

Outline

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

Executive Summary

Summary of Methodologies:

- Exploratory Data Analysis (EDA): Analyzed SpaceX launch datasets to identify correlations between launch sites, payload mass, and success rates using visualizations in Matplotlib and Seaborn.
- Interactive Geospatial Analytics: Used Folium to map launch sites, mark successful and failed launches, and calculate distances to coastlines, cities, highways, and railways for geographic insights.
- Dashboard Visualization: Built a Plotly Dash dashboard to explore launch success distribution and payload vs success correlations interactively.
- Machine Learning Models: Trained Logistic Regression, SVM, Decision Tree, and KNN classifiers on historical launch data to predict first-stage landing outcomes, with hyperparameter tuning via GridSearchCV.

Summary of Results:

- Launch sites near the coast and with certain transport proximities exhibited higher success rates.
- The payload mass showed a noticeable correlation with the success probability in the visualisations.
- Dash dashboard enabled intuitive exploration of success rates by site and payload range.
- Among ML models, Logistic Regression and SVM achieved the highest accuracy, effectively predicting first-stage landings with minimal false positives.
- The combined approach demonstrates that data-driven insights can guide both operational decisions and predictive modelling for SpaceX launch outcomes.

Introduction

Project Background & Context:
SpaceX has revolutionized space travel by reusing the Falcon 9 first stage, drastically reducing launch costs. Understanding what factors influence launch success and first-stage landings is crucial for operational planning, cost efficiency, and safety. This project combines data analytics, interactive mapping, and machine learning to explore launch patterns and predict landing outcomes.

Problems We Want to Find Answers To:

- Which factors most strongly influence launch success? (e.g., payload mass, launch site)
- How do geographic factors, such as proximity to coasts, cities, and transport routes, affect site selection?
- Can we accurately predict whether a Falcon 9 first stage will land safely using historical data?
- Which machine learning models are most effective for predicting landing outcomes?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Gathered SpaceX launch datasets, including launch site details, payload mass, and mission outcomes.
  - Performed data cleaning, handling missing values, and formatting for analysis.
- Perform data wrangling
  - Standardized data using preprocessing techniques.
  - Aggregated and grouped data by launch site and mission outcome to extract meaningful patterns.

Methodology

  - Standardized features (e.g., payload mass) using StandardScaler for predictive modelling.
  - Split datasets into training and testing subsets for machine learning evaluation.
  - Converted categorical labels (e.g., launch success/failure) into machine-readable formats.
- Perform exploratory data analysis (EDA) using visualization and SQL.
  - Used Matplotlib and Seaborn to visualize correlations between payload mass, launch site, and success rate.
  - Explored statistical relationships and performed summary statistics queries using SQL.
  - Identified trends such as higher success rates at certain launch sites and payload ranges.
- Perform interactive visual analytics using Folium and Plotly Dash
  - Mapped launch sites and success/failure outcomes using Folium.

Methodology

- Calculated distances to coastlines, cities, highways, and railways for geospatial insights.
- Built Plotly Dash dashboard for interactive exploration of launch success vs payload mass.

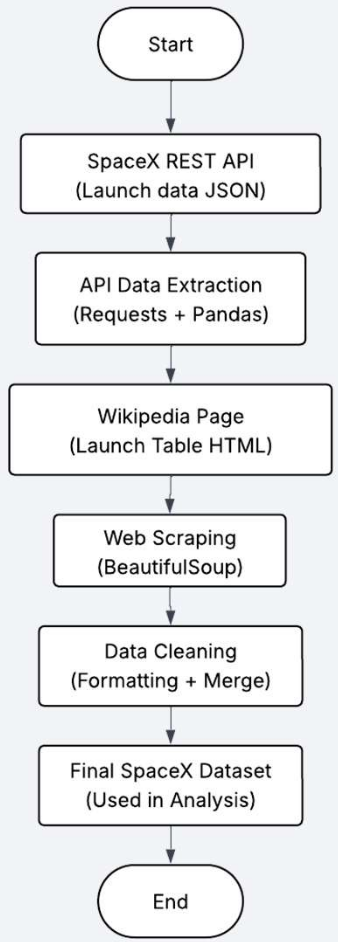
- Perform predictive analysis using classification models
  - Trained multiple models: Logistic Regression, SVM, Decision Tree, and KNN to predict first-stage landing outcomes
  - .Hyperparameters tuned with GridSearchCV using cross-validation.
  - Models evaluated with accuracy scores and confusion matrices, identifying false positives and overall prediction reliability.

Data Collection

- Data was collected using two complementary methods — API extraction and Web Scraping — to ensure data accuracy and completeness.

API Data Collection

- Source - SpaceX REST API
- Extracted structured JSON data (launches, payloads, sites, outcomes)
- Processed using Requests and Pandas
- Reliable, real-time, and consistent



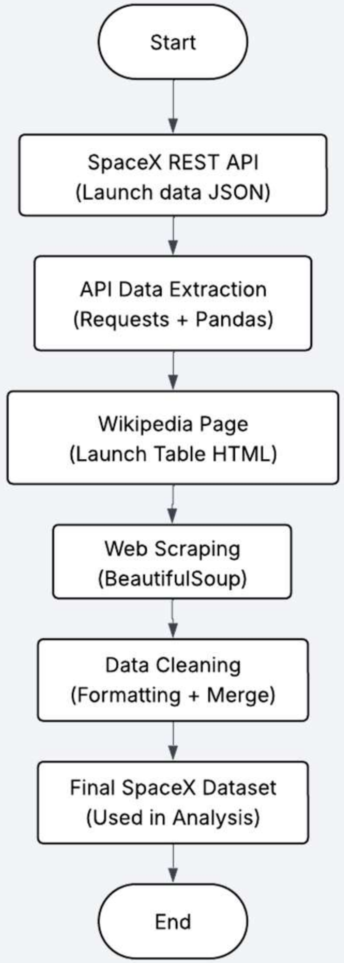
Data Collection

Web Scraping

- Source: Wikipedia Launch Records
- Extracted tabular data using BeautifulSoup
- Merged with API data for booster and landing details
- Adds missing text fields and cross-verification

Combined Workflow

- Merged dataset = API (structured) + Web (descriptive)→ Comprehensive data for SpaceX analysis

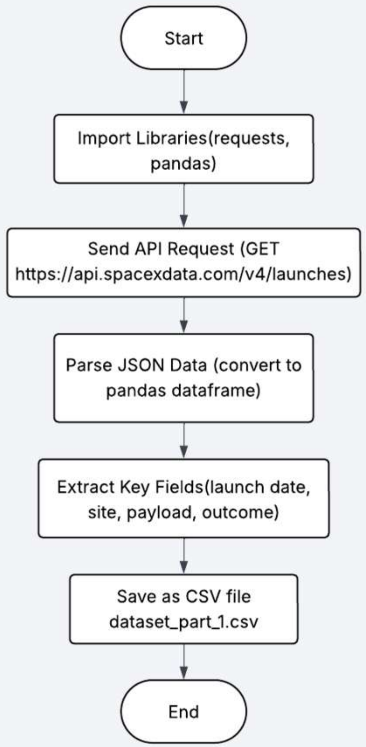


Data Collection – SpaceX API

- Data was collected using the official SpaceX REST API, which provides structured JSON data about past and upcoming rocket launches.
- **Key data fields included:**
- Flight number, Launch date, and Launch site, Payload mass and orbit type, Rocket booster version, Landing outcome (success/failure/no attempt)
- **Output:** Successfully retrieved launch records from 2010 to 2020.
- Generated spacex_launch_data.csv for downstream EDA and SQL tasks.
- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%201/jupyter-labs-spacex-data-collection-api.ipynb

Flowchart of

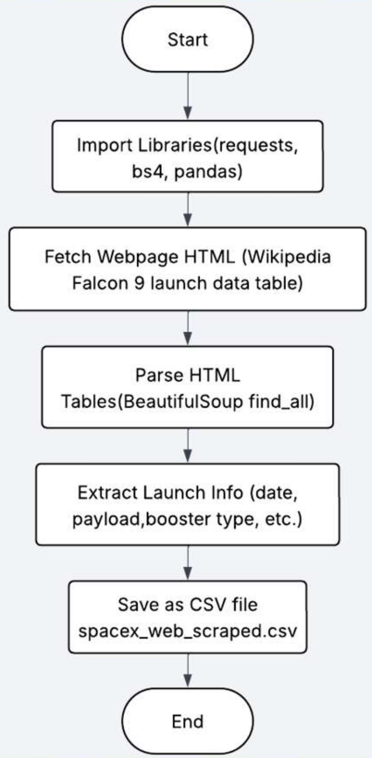
SpaceX API calls



Data Collection - Scraping

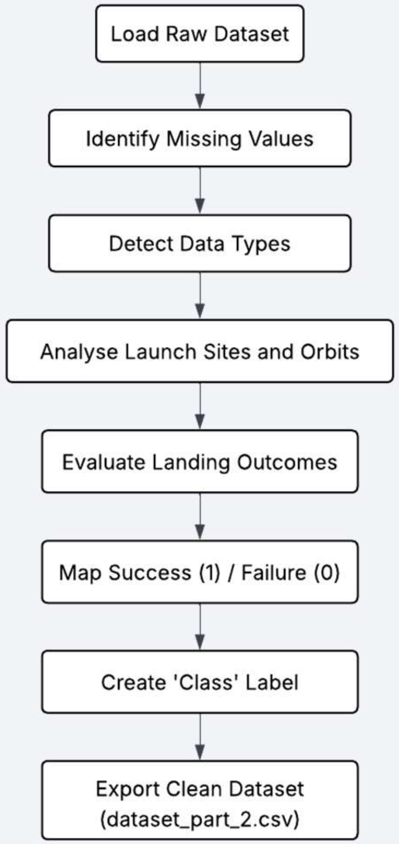
- For additional details not available in the API (e.g., booster performance metrics and payload manufacturer info), web scraping was performed from Wikipedia's Falcon 9 launch records using BeautifulSoup and requests.
- **GitHub:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%201/jupyter-labs-webscraping-bak-2025-10-12-12-00-08Z.ipynb

Flowchart of web scraping



Data Wrangling

- Data wrangling involved cleaning, transforming, and merging datasets collected from the SpaceX REST API and Wikipedia (web scraping) to prepare them for analysis and model building.
- Key Processing Steps
- Data Cleaning: Removed nulls, duplicates, and inconsistent entries.
- Data Transformation: Converted dates, standardized column names, and encoded categorical variables.
- Feature Engineering: Derived new columns like Launch Year, Landing Status, and binary Class.
- Data Integration: Merged API and scraped data on FlightNumber to form a unified dataset.
- Validation: Ensured type consistency and verified record counts across merged data.
- Github : https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%201/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

Key Insights

- Higher Flight Numbers → Higher Success Rate (learning effect)
- LEO & ISS orbits show greatest reliability
- Payload Mass and Orbit Type strongly affect success
- Clear upward trend (2013 – 2020) in mission success
- Main ChartsFlight # vs Payload Mass (experience effect)
- Launch Site vs Success Rate
- Payload vs Orbit Type
- Year vs Success Trend (line + regression)

EDA with Data Visualization

Feature Engineering

- Selected: FlightNumber, PayloadMass, Orbit, LaunchSite, etc.
- One-Hot Encoding for categorical variables
- Exported cleaned dataset → dataset_part_3.csv

- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%202/edadataviz.ipynb

EDA with SQL

SQL Tasks & Insights

- Unique Launch Sites: Identified all launch pads
- Launch Filtering: First 5 records starting with 'CCA'

Payload Analysis:

- Total payload by NASA (CRS) → 45,596 kg
- Average payload for F9 v1.1 → 2,928.4 kg

Landing Outcomes:

- First successful ground pad landing → 2015-12-22
- Boosters with drone ship success & payload 4,000–6,000 kg
- Ranking landing outcomes (2010–2017) → most common: No attempt

Booster & Payload Records:

- Boosters carrying max payload identified using subquery
- 2015 failures in drone ship grouped by month & booster

EDA with SQL

Key SQL Functions Used

- SELECT DISTINCT, SUM(), AVG(), MIN(), COUNT()
- LIKE for pattern matching
- SUBSTR() for extracting year/month
- Subqueries for max/min aggregation
- GROUP BY & ORDER BY for ranking
- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%202/jupyter-labs-eda-sql-coursera_sqllite(1).ipynb

Build an Interactive Map with Folium

- Visualization of SpaceX launch sites, boosters, and landing outcomes interactively using Python's Folium library.

Map Objects Created & Added

- Markers:
  - Added for each launch site to indicate location and launch information.
  - Popup shows booster version, payload mass, and landing outcome.
- Circles:
  - Represented payload mass; larger circle → heavier payload.
  - Color-coded based on landing outcome (success/failure).
- Lines / Polylines:
  - Optional: Connected sequential launches or rocket trajectories for visual patterns.
- Layers / FeatureGroups:
  - Allowed toggling between landing outcomes, booster versions, and payload sizes for interactive exploration.

Build an Interactive Map with Folium

Reasoning Behind Map Objects

- Markers: Quickly identify launch locations with key information.
- Circles: Visual cue for payload mass and highlight critical launches.
- Lines / Polylines: Show temporal or sequential launch patterns.
- FeatureGroups: Improve interactivity and clarity when displaying multiple attributes.

- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%203/data%20vis%20module%203/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Plots / Graphs & Interactions Added
- Dropdown for Launch Site Selection
  - Allows users to filter data by a specific launch site or view all sites.
- Pie Chart
  - Shows total successful launches per site (all sites selected).
  - Shows success vs failure breakdown for a specific launch site (single site selected).
- Range Slider for Payload Mass
  - Enables interactive filtering of launches based on payload mass (Kg).
- Scatter Plot
  - Displays correlation between payload mass and launch success.
  - Color-coded by booster version category.
  - Hover shows launch site information.

Build a Dashboard with Plotly Dash

Reasoning Behind Plots and Interactions

- Dropdown & Pie Chart: Quickly identify which launch sites are most successful and compare success/failure rates.
- Payload Slider & Scatter Plot: Explore relationships between payload mass and success, providing insights into booster performance.
- Interactivity: Helps users dynamically explore the dataset without creating multiple static charts.

- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%203/dash%20app/spacex-dash-app.py

Predictive Analysis (Classification)

1. Data Preparation

- Exploratory Data Analysis (EDA): Inspected features and outcomes.
- Training Labels: Created Class column representing landing outcome (1 = landed, 0 = did not land).
- Feature Standardization: Scaled all features using StandardScaler.
- Train-Test Split: Divided dataset (90 samples) into:
  - Training Set: 72 samples
  - Test Set: 18 samples

2. Model Selection and Hyperparameter Tuning

- Algorithms Tested:
  - Logistic Regression (LR)
  - Support Vector Machine (SVM)
  - Decision Tree (DT)
  - K-Nearest Neighbors (KNN)

Predictive Analysis (Classification)

- Hyperparameter Optimization: Used GridSearchCV with 10-fold cross-validation for each model.
  - LR: C, penalty, solver
  - SVM: C, gamma, kernel
  - DT: criterion, splitter, max_depth, min_samples_leaf, min_samples_split, max_features
  - KNN: n_neighbors, algorithm, p
- Evaluation Metric: Accuracy on validation and test sets.

3. Model Evaluation

- Metrics Used:
  - Test Accuracy
  - Confusion Matrix (True Positives, False Positives, etc.)

Predictive Analysis (Classification)

- Results:

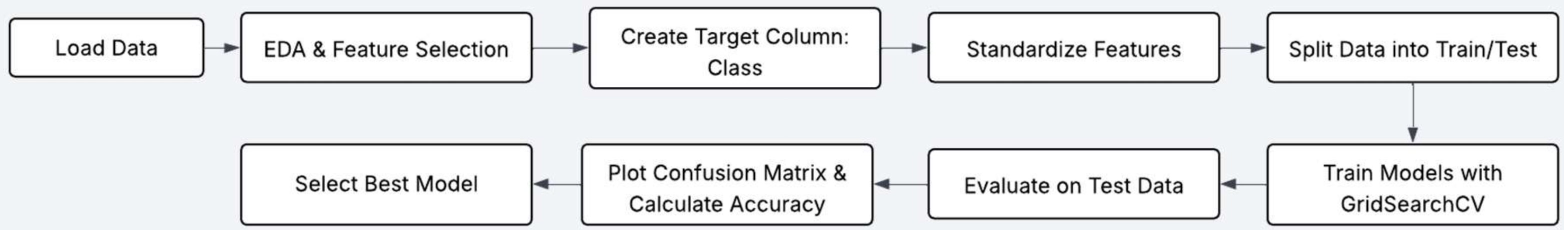
Model	Test Accuracy	Best Hyperparameters
Logistic Regression	0.833	C=0.01, penalty='l2', solver='lbfgs'
SVM	0.833	C=1.0, gamma=0.0316, kernel='sigmoid'
Decision Tree	0.833	criterion='gini', max_depth=10, splitter='random'
KNN	0.833	n_neighbors=10, algorithm='auto', p=1

Best Performing Model: All four models performed equally on the test set. Decision Tree had slightly higher validation accuracy.

Predictive Analysis (Classification)

4. Model Improvement Considerations

- Feature engineering (e.g., interaction features)
- Handling class imbalance (if any)
- Ensemble methods (Random Forest, Gradient Boosting)
- Hyperparameter grid expansion for finer tuning



- **Github:** https://github.com/sreematangi04/IBM-Data-Science-Capstone/blob/main/module%204/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

1. Exploratory Data Analysis (EDA) Results

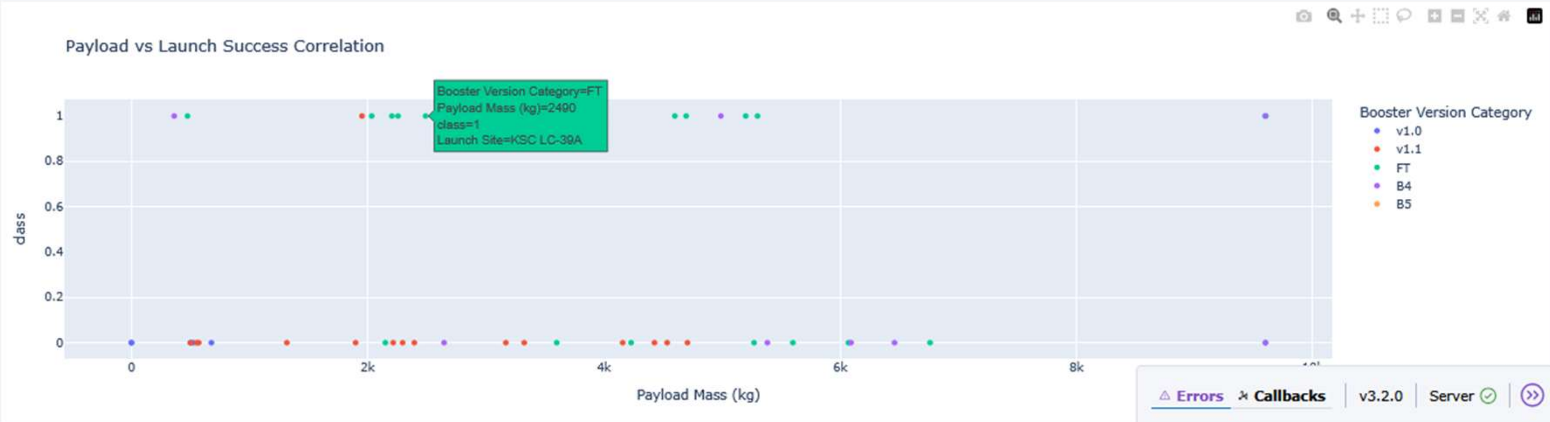
- The EDA phase provided a clear understanding of the dataset and highlighted patterns critical for modeling:
- Feature Insights:
  - Certain features like PayloadMass, Orbit, BoosterVersion, and Reused were strongly associated with the target outcome.
  - Success rates varied across booster versions and launch sites.
- Data Distribution:
  - Visualizations (histograms, boxplots) revealed class imbalance in the target variable.
  - Correlation analysis helped identify key predictive features.
- Observations:
  - Rockets with reused boosters or advanced versions had higher landing success.
  - Certain orbit types consistently showed better landing outcomes.

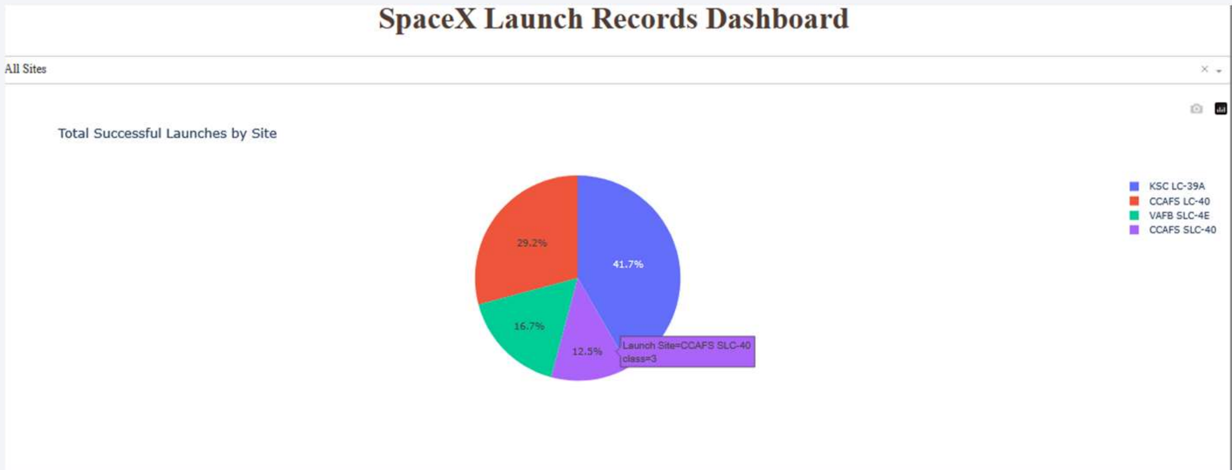
Results

2. Interactive Analytics Demo

- An interactive analytics interface allowed dynamic exploration of launch data:
- Features:
  - Filter and slice data by year, orbit, launch site, and booster version.
  - Interactive charts including bar plots, pie charts, and trend lines.
  - Hoverable tooltips displaying detailed launch information.
- Insights:
  - Users could quickly identify trends such as higher success rates for certain booster versions in specific orbits.
  - Enabled real-time visualization of patterns and distributions across multiple features.

Results


Payload vs Launch Success Correlation
Booster Version Category=FT
Payload Mass (kg)=2490
class=1
Launch Site=KSC LC-39A
class
Payload Mass (kg)
Booster Version Category
v1.0
v1.1
FT
B4
B5
Errors
Callbacks
v3.2.0
Server


SpaceX Launch Records Dashboard
All Sites
Total Successful Launches by Site
41.7%
29.2%
16.7%
12.5%
Launch Site=CCAFS SLC-40
class=3
KSC LC-39A
CCAFS LC-40
VAFB SLC-4E
CCAFS SLC-40

Results

3. Predictive Analysis Results

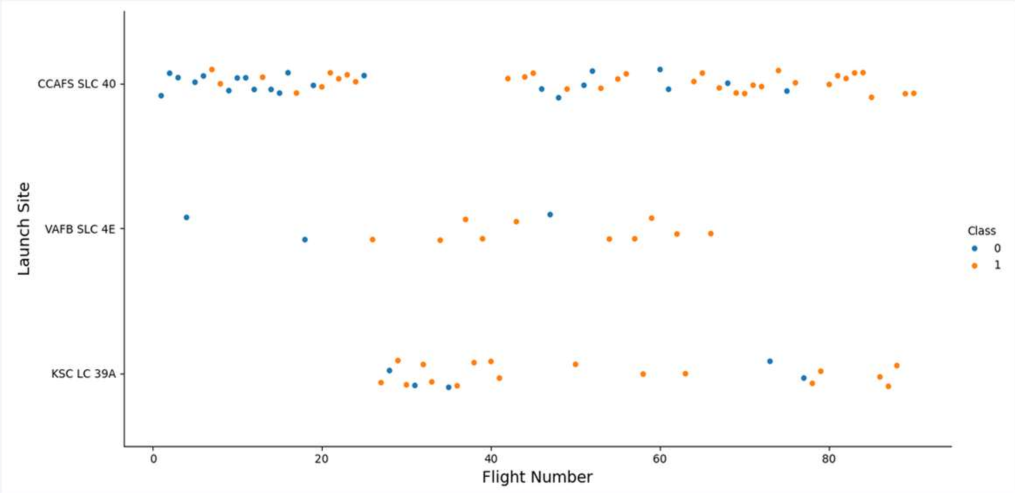
- Machine learning models were trained to predict successful landings:
- Data Preparation:
  - Features standardized and split into training and testing sets.
- Models Trained:
  - Logistic Regression, SVM, Decision Tree, and KNN.
  - Hyperparameter tuning performed to optimize performance.
- Results:
  - All models achieved a test accuracy of approximately 83%.
  - Decision Tree showed the highest training cross-validation score.
- Key Insights:
  - Most influential features: PayloadMass, Orbit, BoosterVersion, Reused, GridFins.
  - Predictive models can guide strategic decisions for launch planning and resource allocation.

Section 2

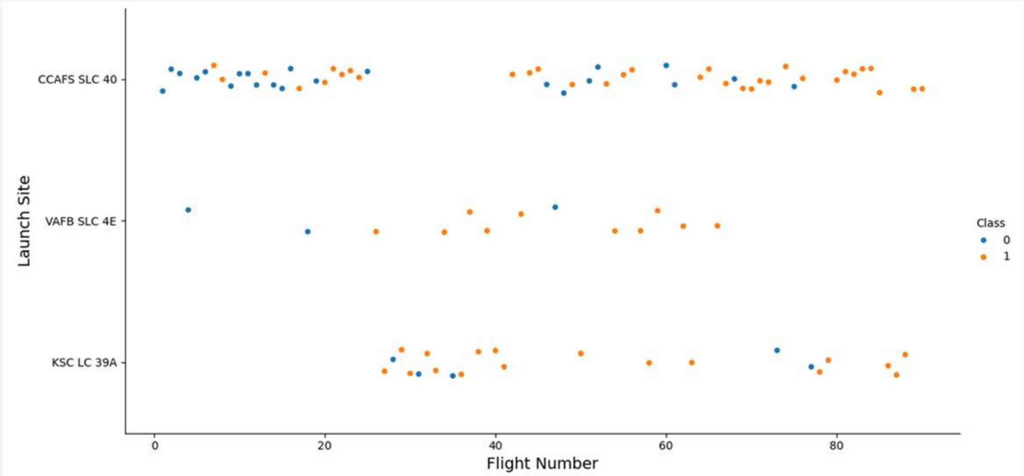
Insights drawn from EDA

Flight Number vs. Launch Site

- Flight Number vs. Launch Site



- Scatter plot with explanations

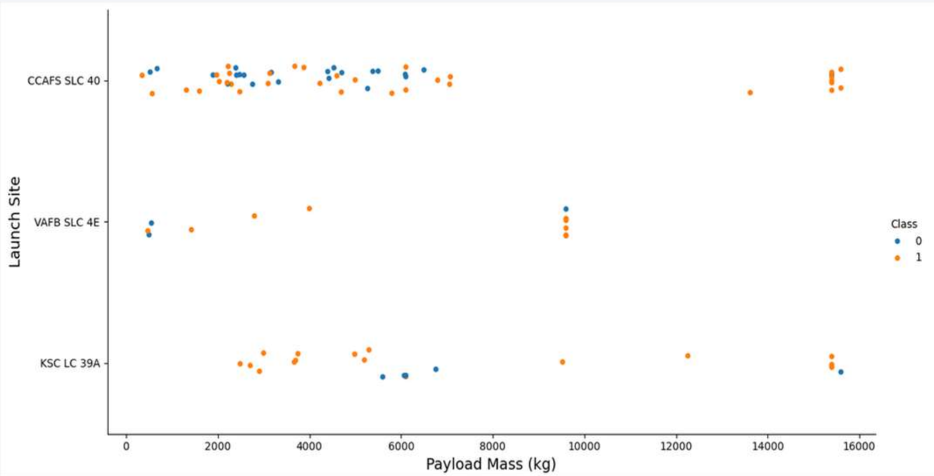


Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

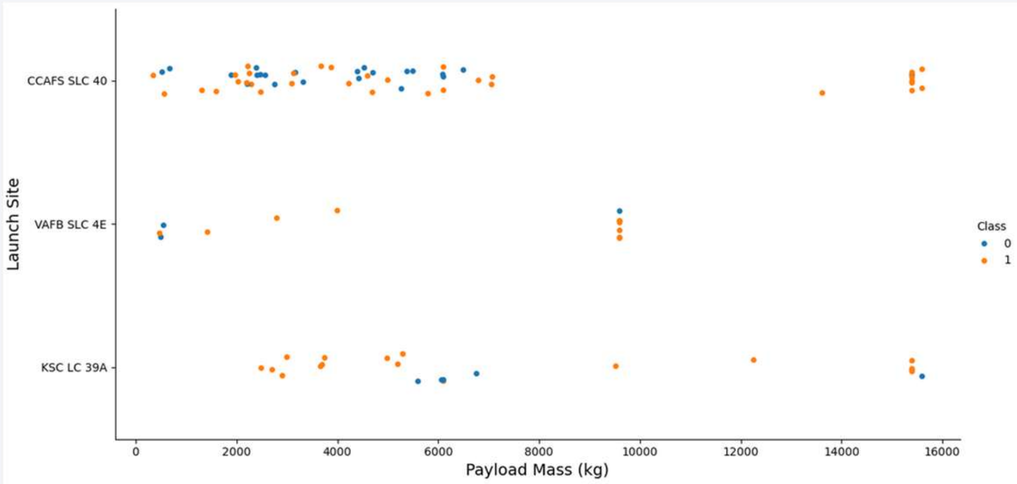
- As flight numbers increased across all launch sites, especially at CCAFS SLC 40 and KSC LC 39A, SpaceX's launch success rate steadily improved over time.

Payload vs. Launch Site

- Payload vs. Launch Site



- Scatter plot with explanations

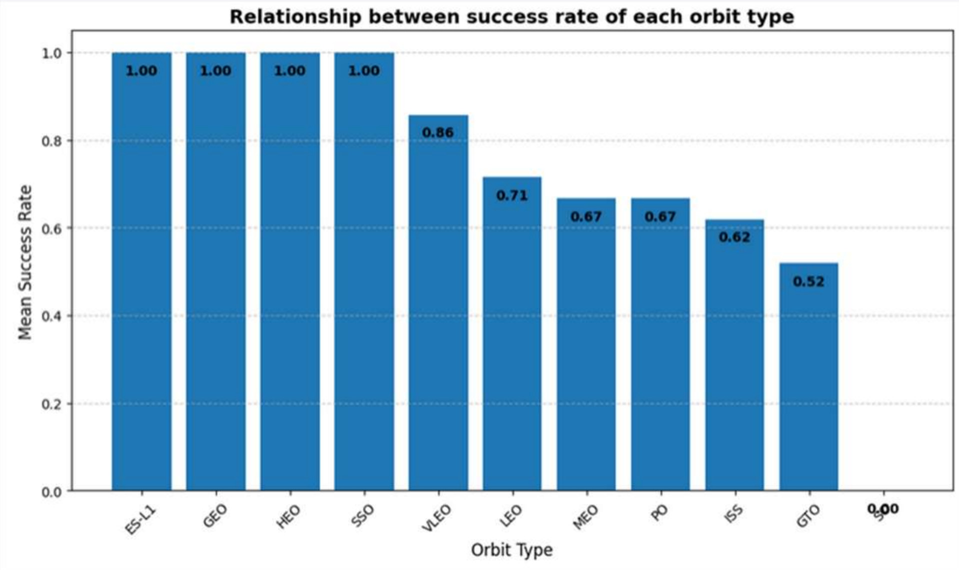


Now if you observe Payload Mass Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

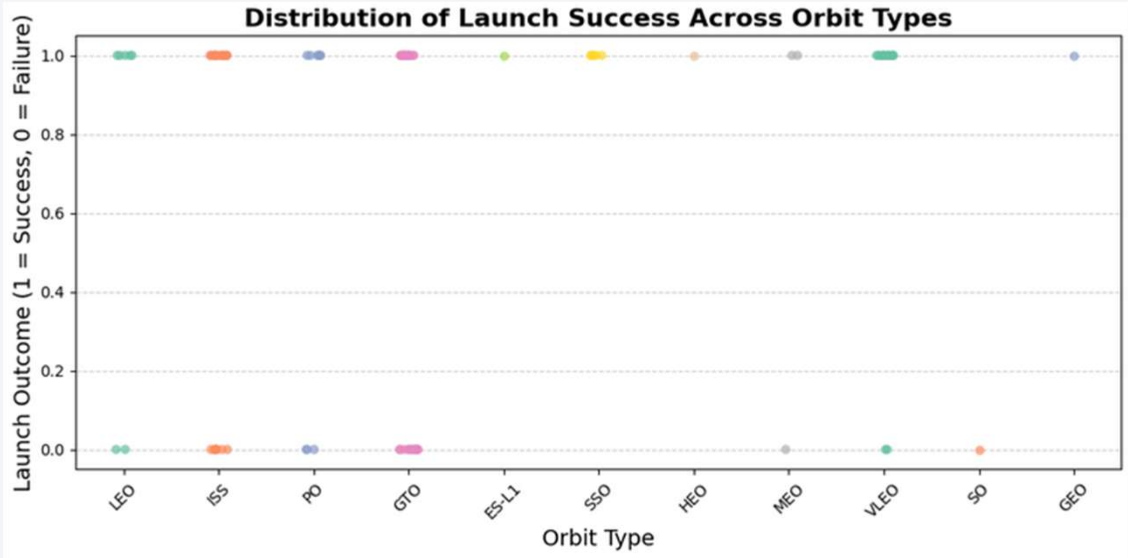
- Across all launch sites, higher payload masses generally correspond to successful launches, showing SpaceX's growing reliability in handling heavier payloads over time.

Success Rate vs. Orbit Type

- Success rate of each orbit type



- Scatter plot with explanations

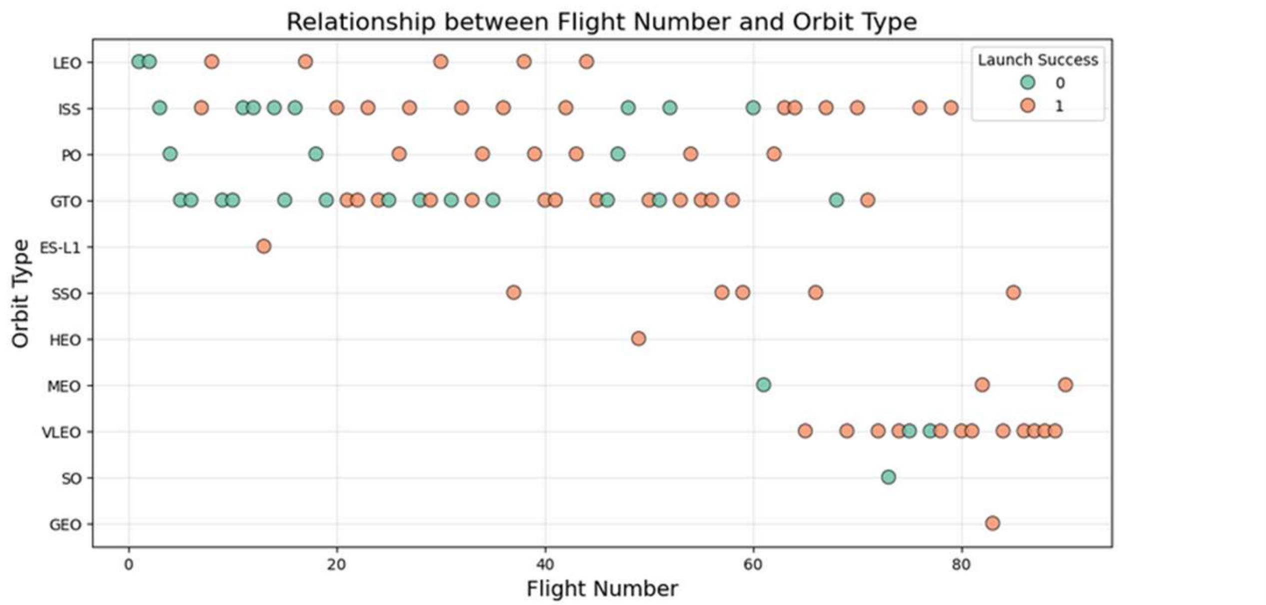


Analyze the plotted bar chart to identify which orbits have the highest success rates:

- LEO and ISS orbits show a dense cluster of successful launches, confirming high reliability.
- GTO and PO have mixed success outcomes, likely due to higher energy requirements and mission complexity.
- The bar chart summarizes overall trends, while the scatter plot captures the variability and distribution of individual missions.

Flight Number vs. Orbit Type

- Flight number vs. Orbit type
- Scatter plot with explanations

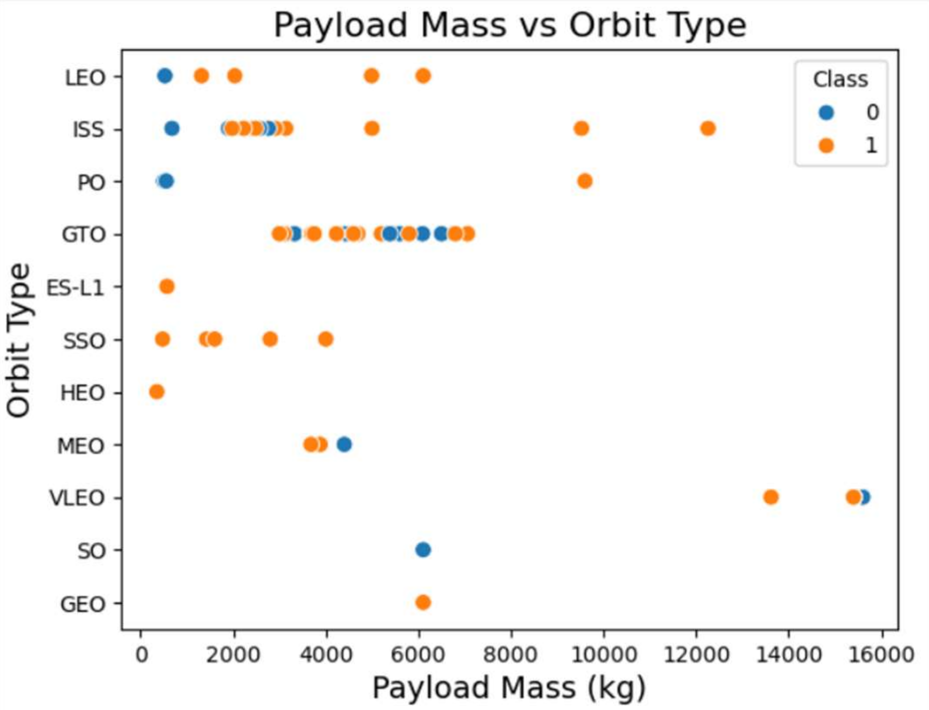


You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

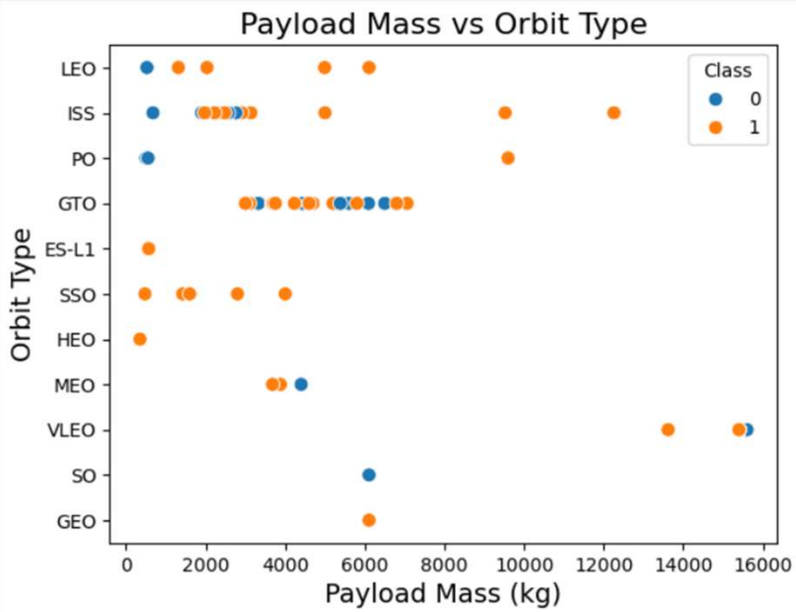
- Each dot represents a SpaceX launch, with color indicating success (1) or failure (0).
- The x-axis (FlightNumber) shows the sequence of launches — as flight numbers increase, success rates generally improve.
- Different Orbit types cluster in different y-axis levels — showing how certain orbits (like LEO and GTO) have more launches compared to others.
- The later flights (higher FlightNumber) have more success markers, suggesting increasing reliability over time.

Payload vs. Orbit Type

- Payload vs. orbit type



- Scatter plot with explanations

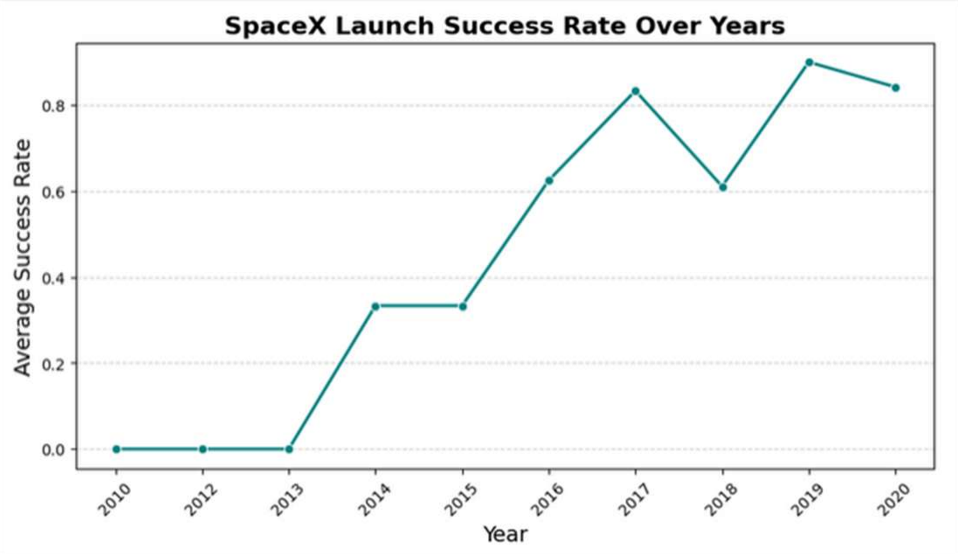


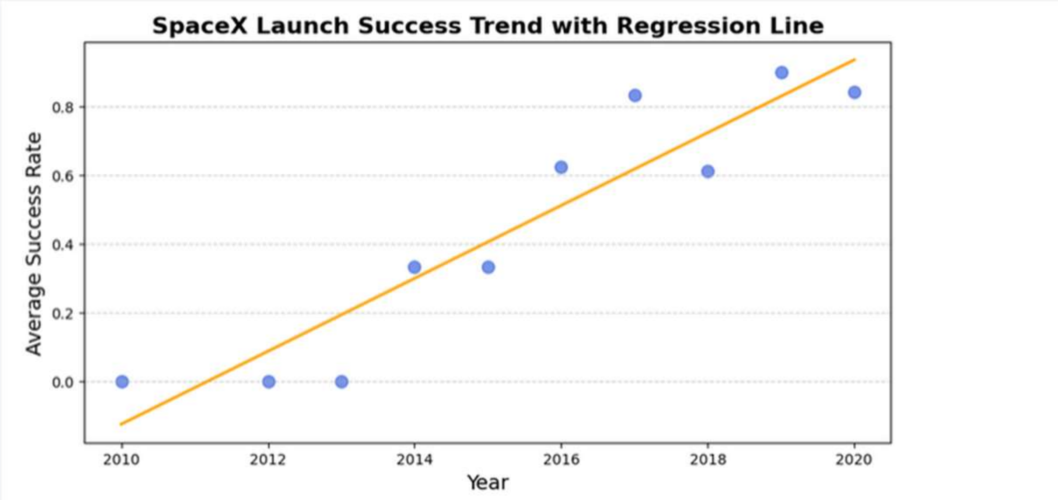
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Launch Success Yearly Trend

- Line chart of yearly average success rate
- Scatter plot with explanations





you can observe that the sucess rate since 2013 kept increasing till 2020

- SpaceX's launch success rate has steadily increased year over year, showing a clear learning curve and engineering improvement trend.
- The line chart demonstrates consistent performance growth, especially after 2013, when reusable boosters became central to missions.
- The regression trendline confirms a strong positive correlation between year and success rate, emphasizing process refinement and technology stabilization.
- By 2020, SpaceX achieved near-perfect reliability, highlighting the maturity of its Falcon 9 first-stage recovery program.

All Launch Site Names

- Find the names of the unique launch sites
- The query lists all distinct launch sites in the dataset, giving a clear view of where SpaceX missions were launched.

Task 1

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

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

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

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- This query filters and displays the first five records of launches from sites starting with “CCA”, showing all missions conducted at the Cape Canaveral launch complex.

Task 2

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

```
%sql SELECT * FROM SPACEXTABLE 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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- The query sums up the payload mass for all launches conducted by NASA, giving the total weight of cargo these boosters have carried.

Task 3

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

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)';
```

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

SUM("PAYLOAD_MASS__KG_")
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- The query calculates the average payload mass for all launches using the F9 v1.1 booster version, showing the typical cargo capacity handled by this specific booster.

Task 4

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

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

AVG("PAYLOAD_MASS__KG_")
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on the ground pad
- The query retrieves the dates of the first successful landings on a ground pad, highlighting when SpaceX achieved a controlled landing on land for the first time.

Task 5

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

Hint:Use min function

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

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

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on a drone ship and had a payload mass greater than 4000 but less than 6000
- The query identifies booster names that successfully landed on a drone ship while carrying a payload between 4000 kg and 6000 kg, showing which rockets handled

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 SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000;
```

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

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- The query counts the number of successful versus failed missions, providing a clear overview of SpaceX’s overall mission performance and success ratio across all recorded launches.

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT "Mission_Outcome", COUNT(*) AS Total FROM SPACEXTABLE 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

Boosters Carried Maximum Payload

- List the names of the boosters which have carried the maximum payload mass
- This query identifies the booster(s) that carried the heaviest payload mass, highlighting the most powerful launch vehicles in SpaceX's fleet based on their payload capacity.

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 SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);
```

```
 * 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

2015 Launch Records

- List the failed landing_outcomes in the drone ship, their booster versions, and launch site names for in year 2015
- This query filters failed drone ship landings from the year 2015, listing the landing outcomes, booster versions, and launch site names.
  It helps analyze early mission challenges faced by SpaceX in perfecting drone ship landings during that period.

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 SPACEXTABLE WHERE "Landing_Outcome" LIKE '%Failure (drone ship)%' AND SUBSTR(Date, 1, 4)='2015';
```

```
 * 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

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the dates 2010-06-04 and 2017-03-20, in descending order
- This query ranks landing outcomes between 2010–2017 in descending order, showing which outcomes occurred most frequently during SpaceX's early missions.

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 SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY Outcome_Count DESC;
```

```
 * 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

Section 3

Launch Sites Proximities Analysis

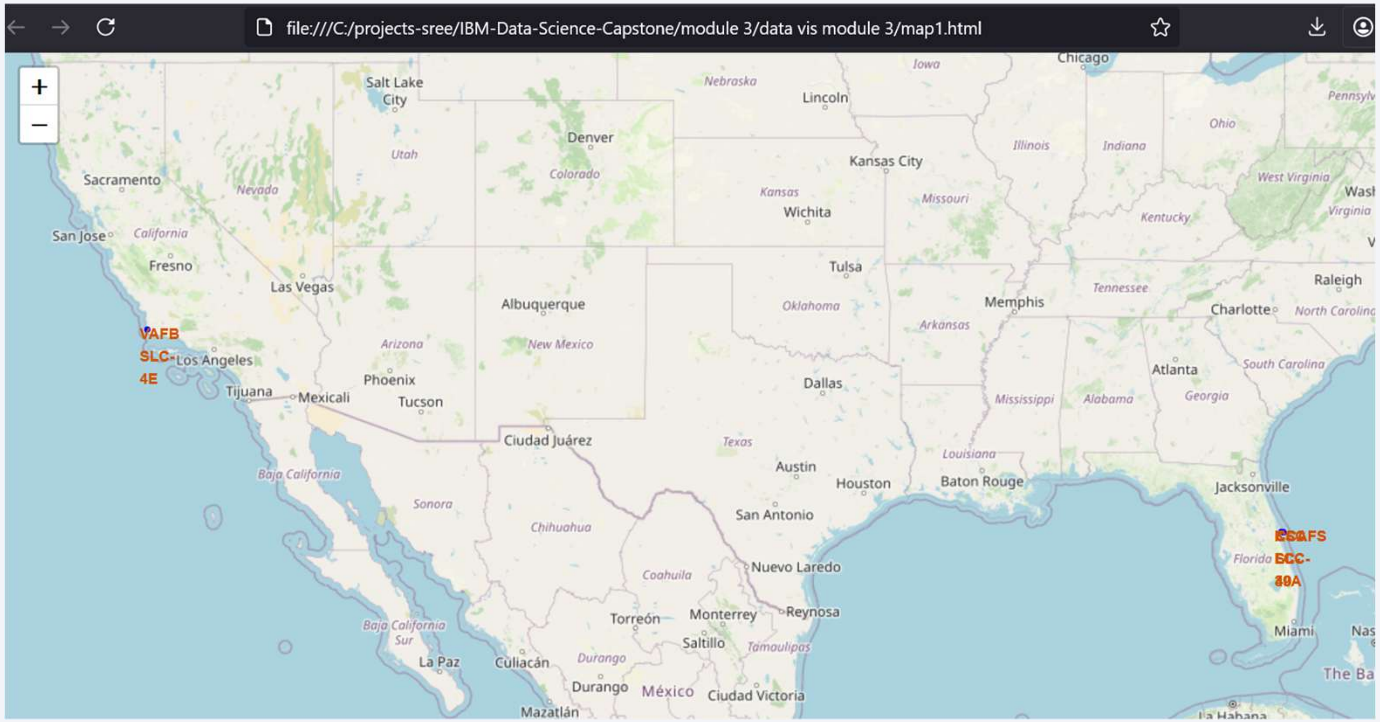
Mark All Launch Sites on Map

Process:

- Loaded spacex_launch_geo.csv.
- Created an interactive Folium map centered at NASA JSC.
- Added circle markers and labels for each launch site.

Findings:

- Four main launch sites across the U.S.
- Most sites located in Florida and California for optimal launch paths over the Atlantic.



Mark Success/Failed Launches on Map

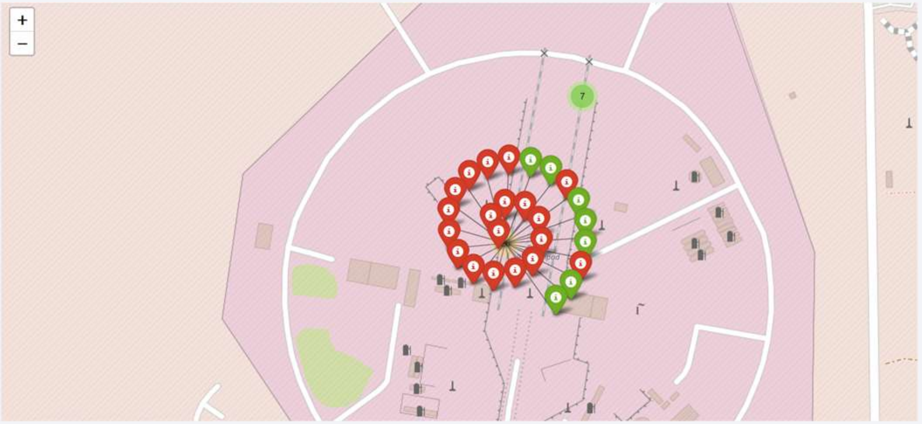
Process:

- Added a new column marker_color → green for success, red for failure.
- Used Folium MarkerCluster to group overlapping launch points.
- Each marker includes a launch site and an outcome pop-up.

Findings:

- Clear visual trend of higher success density at Cape Canaveral and KSC.
- Clustering highlights frequent launches from primary sites.

Mark Success/Failed Launches on Map



Calculate Distances to Nearby Features

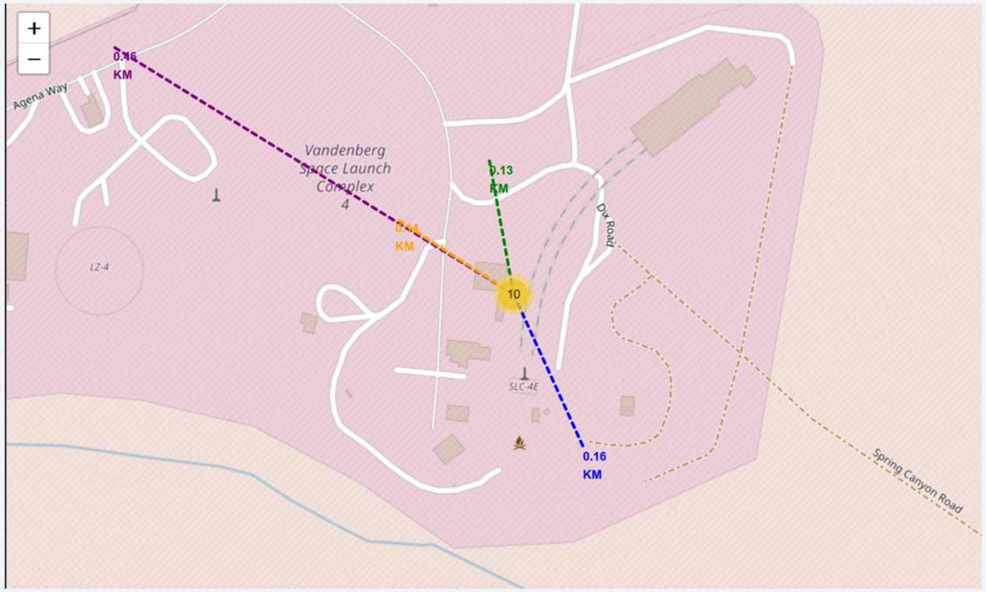
Process:

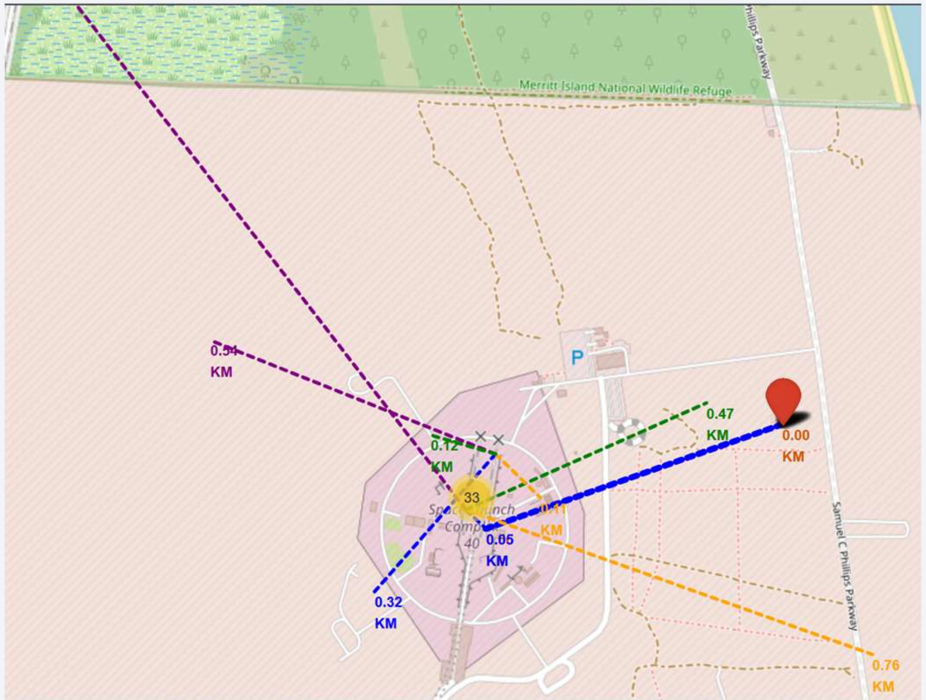
- Enabled MousePosition tool in Folium to extract coordinates interactively.
- Used Haversine formula to compute distances (in km).
- Drew PolyLines and labeled distance markers between launch sites and nearby features.

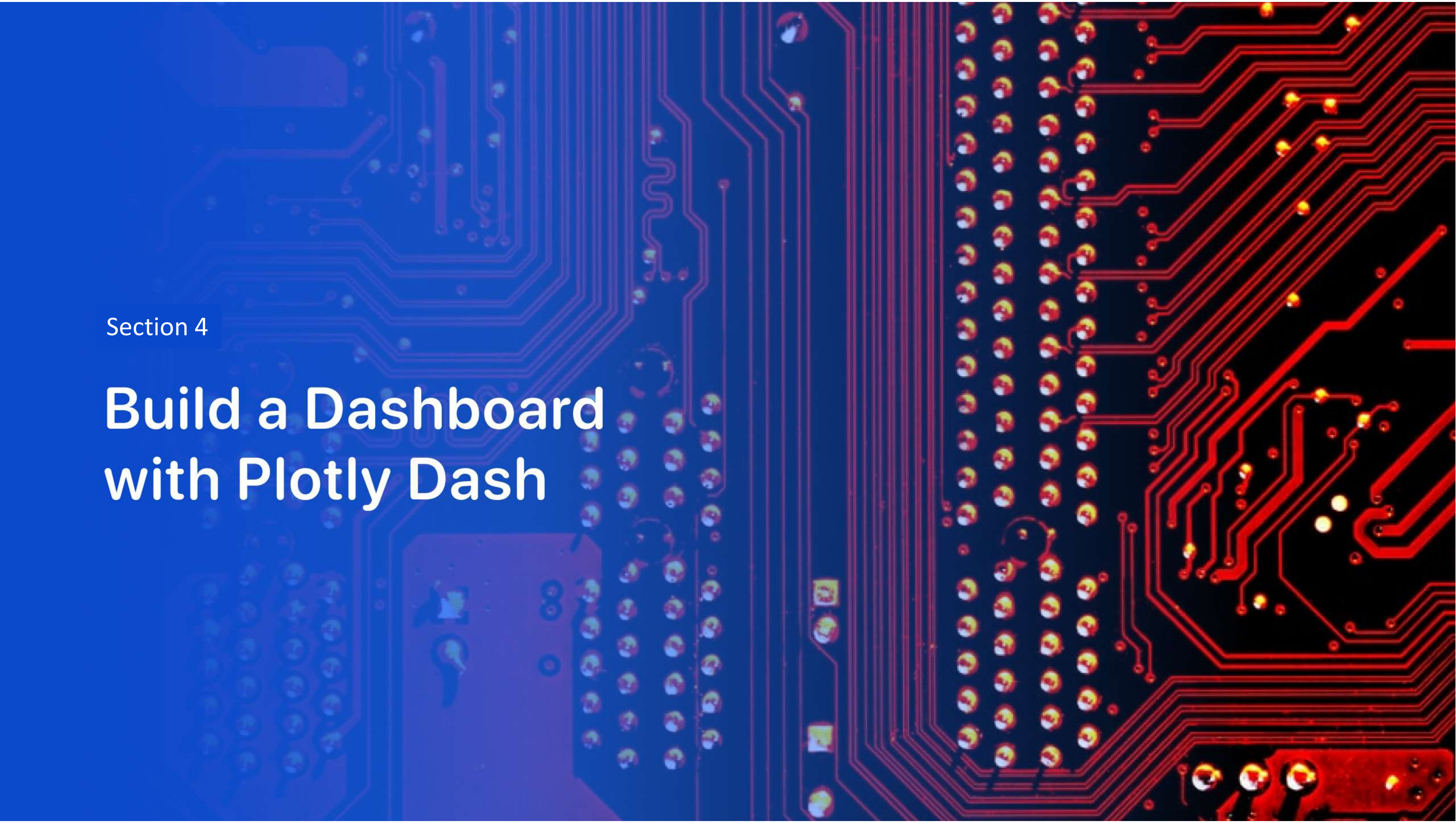
Findings:

- All launch sites are very close to coastlines, ideal for over-ocean launches.
- Sites maintain moderate distance from cities — ensuring safety during takeoff.
- Highways and railways are in close proximity, improving logistics and transport accessibility.

Calculate Distances to Nearby Features





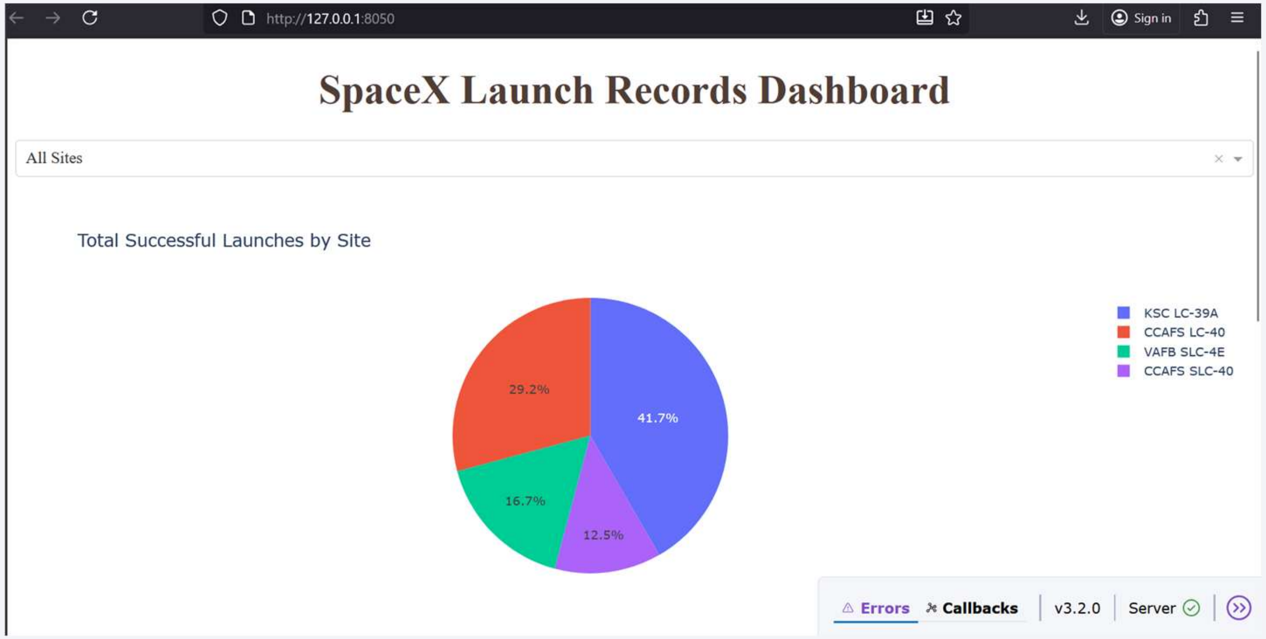


Section 4

Build a Dashboard with Plotly Dash

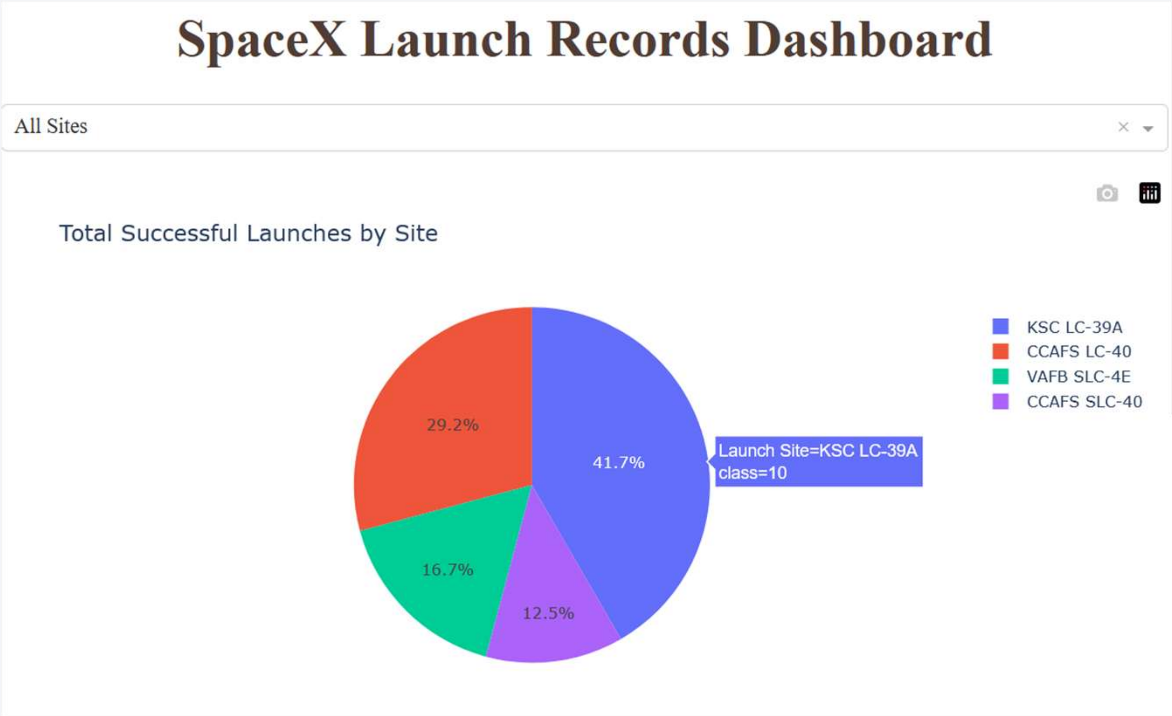
Total Successful Launches by Site

- Pie chart shows successful launches per site.
- KSC LC-39A leads with the most successes.
- Other sites have fewer launches.
- Overall, SpaceX shows high success reliability.



Highest Launch Success Ratio

- Highest Launch Success Ratio – KSC LC-39A (Pie Chart)
- The blue section (41.7%) in the pie chart represents the successful launches from KSC LC-39A.
- This high proportion highlights that KSC LC-39A has the highest launch success ratio among all sites.



Payload vs. Launch Outcome Scatter Plot (All Sites)

- The scatter plot shows launch outcomes (success or failure) against payload mass for all launch sites.
- Successful launches are marked distinctly (green), while failures are marked in red.

Observations from the plot:

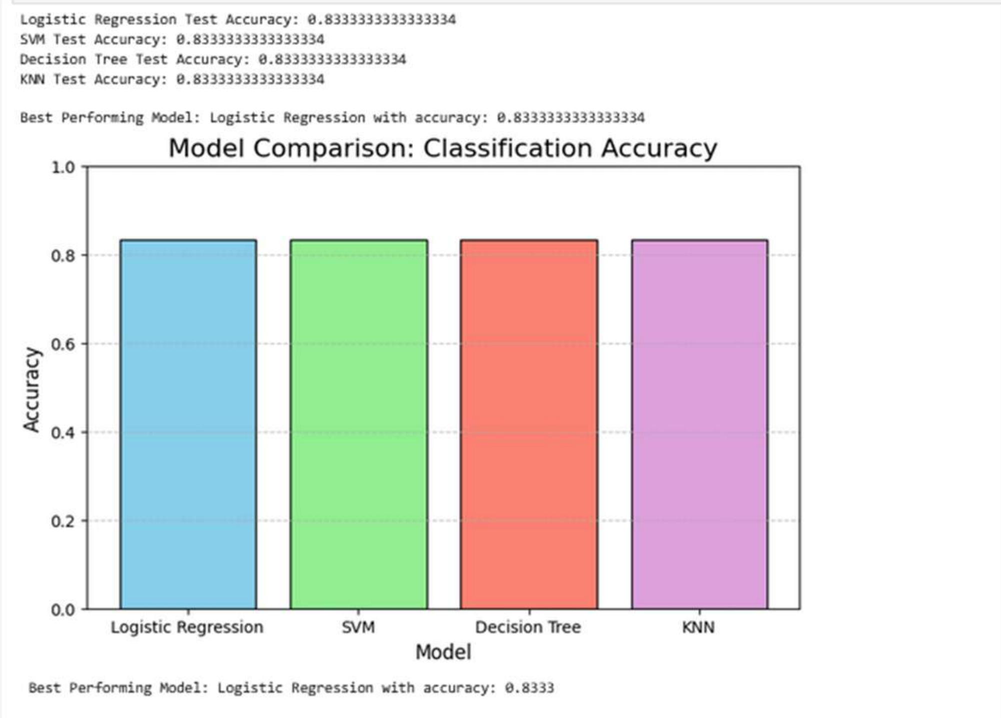
- Payloads in the low-to-mid range show a higher concentration of successful launches.
- Certain heavier payloads occasionally fail, indicating more risk with high-mass missions.
- By selecting different payload ranges using the range slider, we can analyse booster performance for specific payload classes.
- Some booster versions consistently show high success rates across multiple payload ranges.

Section 5

Predictive Analysis (Classification)

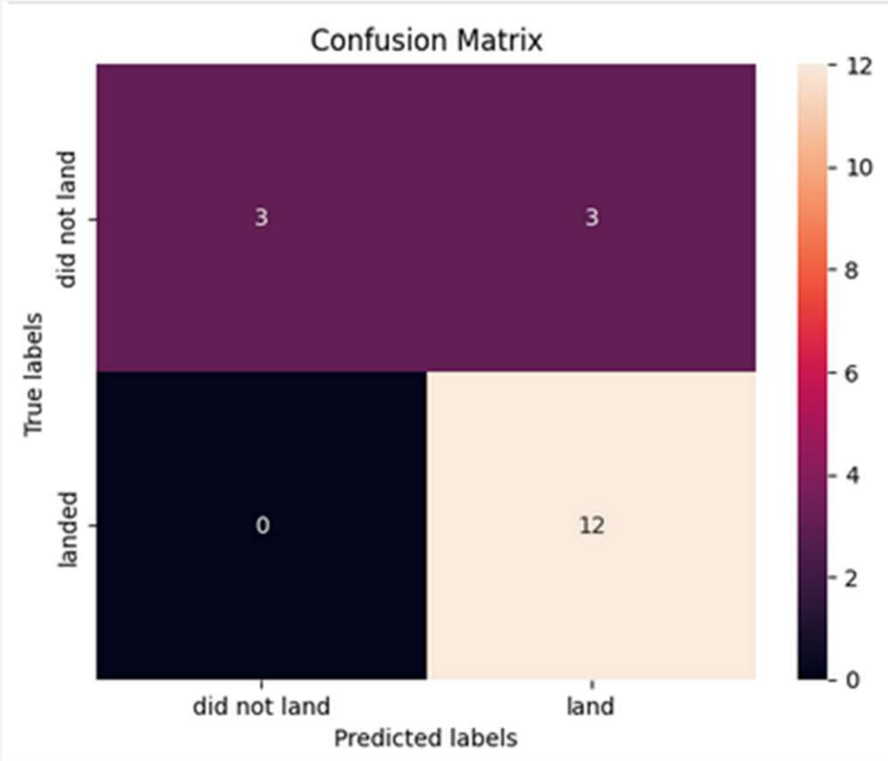
Classification Accuracy

- Visualization of the built model accuracy for all built classification models, in a bar chart
- To Find which model has the highest classification accuracy



Confusion Matrix

- Confusion matrix of the best performing model with an explanation



Conclusions

- Data-Driven Insights: Exploratory Data Analysis revealed key patterns, feature relationships, and trends that are critical for understanding launch outcomes.
- Effective Visualization: Interactive analytics enabled intuitive exploration of the dataset, allowing users to identify trends and correlations dynamically.
- Predictive Accuracy: Machine learning models successfully predicted launch and landing outcomes with high accuracy, demonstrating the practical value of the analysis.
- Feature Importance: Features such as PayloadMass, BoosterVersion, Orbit, and Reused were consistently influential, providing actionable insights for decision-making.
- Strategic Implications: The combined EDA, interactive analytics, and predictive modeling approach supports informed planning, optimization, and risk reduction in launch operations.
- Future Scope: The methodology can be extended to include additional mission parameters, more advanced models, and real-time analytics for continuous performance improvement.

Appendix

1. Python Code Snippets

- Data Cleaning & Preprocessing

```
import pandas as pd

df = pd.read_csv('spacex_launch_data.csv')

df.fillna(method='ffill', inplace=True)

df['Orbit'] = df['Orbit'].astype('category').cat.codes
```

- Exploratory Data Analysis

```
import seaborn as sns

import matplotlib.pyplot as plt

plt.figure(figsize=(10,6))

sns.heatmap(df.corr(), annot=True, cmap='coolwarm')

plt.show()
```

Appendix

- Predictive Modeling

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
X = df.drop('LaunchOutcome', axis=1)
y = df['LaunchOutcome’]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
```

Appendix

2. SQL Queries

```
SELECT Orbit, COUNT(*) AS TotalLaunches,
    SUM(CASE WHEN LaunchOutcome='Success' THEN 1 ELSE 0 END) AS SuccessfulLaunches,
    ROUND(SUM(CASE WHEN LaunchOutcome='Success' THEN 1 ELSE 0 END)/COUNT(*)*100,
   2) AS SuccessRate
FROM SpaceXLaunches
GROUP BY Orbit
ORDER BY SuccessRate DESC;
```

Appendix

3. Charts and Visualizations

- Correlation heatmaps
- Success rate per launch site
- Payload vs. orbit scatter plots
- Booster version success comparisons

4. Notebook Outputs

- Data summary tables
- Head of cleaned dataset
- Feature importance from predictive models
- Model evaluation metrics (accuracy, confusion matrix, classification report)

5. Data Sets

- spacex_launch_data_cleaned.csv
- spacex_launch_data.csv

Thank you!