IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dhinesh Babu C M
29-07-25

Outline

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

Executive Summary

Summary of Methodologies

1. Data Collection & Preprocessing

•Fetched historical SpaceX Falcon 9 launch data
•Cleaned null values, standardized feature types
•Engineered key features like Payload Mass, Orbit, Launch Site, and Landing Outcome

2. Exploratory Data Analysis (EDA)

•Visualized payload vs success rate, orbit influence on landings, and launch site performance
•Used Plotly Express for interactive visualizations

3. Feature Engineering

•One-hot encoding for categorical variables (Orbit, Launch Site, etc.)
•Scaled numerical features using StandardScaler
•Created binary target variable: Class (Success=1 / Failure=0)

4. Model Building

- Tried multiple classifiers:
  - **Logistic Regression**
  - **Support Vector Machines**
  - **Decision Tree**
  - **K-Nearest Neighbors (KNN)**

5. Model Evaluation

- Used **accuracy**, **F1-score**, and **confusion matrix**
- Performed **GridSearchCV** for hyperparameter tuning

Summary of All Results

✅ **Key Insights:**

- **Orbit** and **Launch Site** strongly influence landing success
- **Payloads above 6,000 kg** decrease success rate
- **KNN** performed best after scaling features and tuning k value

Model	Accuracy (Test)	F1-Score	Best Features Used
Logistic Regression	~82%	0.80	Orbit, Payload, Launch Site
Decision Tree	~86%	0.83	All encoded features
SVM (RBF Kernel)	~83%	0.82	Payload, Site, Orbit
K-Nearest Neighbors (Best)	**94%**	**0.93**	All features (scaled + encoded)

✅ **Final Model Chosen:**

K-Nearest Neighbors (KNN) with 94% accuracy and strong generalization on test data.

Introduction

🔍 Summary of Methodologies

We used the **K-Nearest Neighbors (KNN)** algorithm for classification after preprocessing and feature extraction. Data was scaled and split into training and testing sets. The model was fine-tuned by selecting the optimal k value, ensuring balanced evaluation across all classes.

📊 Summary of All Results

The **KNN classifier** achieved the highest accuracy of **94%**, outperforming other baseline models tested during experimentation. Precision, recall, and F1-score also reflected strong model performance, validating the reliability of our approach.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

Data Collection Process – Falcon 9 Landing Prediction :

- Publicly Available Data

- SpaceX Launch Records

- Data Aggregation from APIs

- Feature Engineering from Launch Details

- External Data Sources: Weather, Payload

Data Collection Flowchart:

Start
↓
Fetch Launch Data from SpaceX API / Web Scrapping
↓
Filter: Falcon 9 Missions Only
↓
Extract Key Features:
→ Payload mass
→ Orbit
→ Launch site
→ Booster version
→ Flight number
→ Landing outcome (label)
↓
Clean and Structure Data (handle nulls, types)
↓
Final CSV Dataset → Features + Label (Success/Failure)

Data Collection – SpaceX API

GitHub Link:

Click Here

- Start
- ↓
- Fetch Launch Data from SpaceX API
- ↓
- Filter: Falcon 9 Missions Only
- ↓
- Extract Key Features:
- → Payload mass
- → Orbit
- → Launch site
- → Booster version
- → Flight number
- → Landing outcome (label)
- ↓
- Clean and Structure Data (handle nulls, types)
- ↓
- Final CSV Dataset → Features + Label (Success/Failure)

Data Collection - Scraping

- **GitHub Link:**
- Clink Here

- Start
- ↓
- Fetch Launch Data from Web Scrapping
- ↓
- Filter: Falcon 9 Missions Only
- ↓
- Extract Key Features:
- → Payload mass
- → Orbit
- → Launch site
- → Booster version
- → Flight number
- → Landing outcome (label)
- ↓
- Clean and Structure Data (handle nulls, types)
- ↓
- Final CSV Dataset → Features + Label (Success/Failure)

Data Wrangling

- **Summary :**

•Cleaned and transformed raw Falcon 9 dataset using Pandas.
•Handled missing values, converted data types, and extracted key features.
•GitHub Link: [Click Here]

Start
↓
Load Data
→ Read CSV/Excel into DataFrame
↓
Inspect Data
→ Check shape, info(), head()
↓
Handle Missing Values
→ Drop or impute nulls
↓
Data Type Conversion
→ Convert to appropriate types
↓
Feature Engineering
→ Create new meaningful features
↓
Remove Duplicates / Outliers
→ Ensure clean and consistent data
↓
Final Cleaned Data
→ Ready for EDA / Modeling

EDA with Data Visualization

EDA Chart Summary :

Histogram → To visualize the distribution of continuous variables.
Box Plot → To detect and visualize outliers in the data.
Count Plot → To analyze the frequency of categorical variables.
Correlation Heatmap → To observe relationships between numerical features.
Pair Plot → To explore multivariate relationships and spot patterns.

GitHub Link: Click Here

- Start
- ↓
- Histogram
- → Understand data distribution
- ↓
- Box Plot
- → Detect outliers
- ↓
- Count Plot
- → Analyze categorical features
- ↓
- Heatmap
- → Visualize correlations
- ↓
- Pair Plot
- → Explore feature relationships

EDA with SQL

🧾 Summary of SQL Queries Performed :

- SELECT * FROM table_name → Retrieved the full dataset for analysis.
- SELECT COUNT(*) → Counted total records and missing values.
- SELECT DISTINCT column_name → Identified unique values in categorical columns.
- GROUP BY column_name → Aggregated data to find trends and group patterns.
- ORDER BY column_name DESC → Sorted data to find top/bottom values.
- JOIN table1 ON table1.id = table2.id → Merged related tables.
- WHERE condition → Filtered data for targeted insights.
- AVG(), MAX(), MIN() → Performed basic statistical summaries.

- GitHub Link : Click Here

Build an Interactive Map with Folium

🗺️ Folium Map Objects Summary :

- **Markers** → Added at key locations to represent specific data points like facilities or events.
- **Circle Markers** → Used to visualize data intensity or population around a location.
- **Polylines** → Connected locations to show travel paths or routes.
- **Choropleth Map** (if used) → To show area-based distribution (e.g., by region or state).

🧠 Why These Were Added :

- To **visually represent geospatial data** clearly and interactively.
- To help users **identify patterns**, hotspots, and connections between locations.
- To enhance **user engagement** with hover, zoom, and popup features.

🔗 GitHub URL for Interactive Folium Map

Click Here

Build a Dashboard with Plotly Dash

- 📊 **Dashboard Plots and Interactions Summary :**
- **Bar Chart** → Compared categorical values like success rates by launch site.
- **Pie Chart** → Showed proportions (e.g., successful vs failed landings).
- **Scatter Plot** → Visualized correlations (e.g., payload vs success).
- **Time Series Line Chart** → Tracked changes over time.
- **Dropdowns & Sliders** → Enabled dynamic filtering by date range, payload, or launch site.
- 🎯 **Why These Were Added :**
- To provide **interactive insights** for users to explore data visually.
- To **compare, filter, and drill down** into specific metrics.
- To make the dashboard **user-friendly and data-driven** for decision-making.
- 🔗 **GitHub URL for Plotly Dash Lab : Click Here**

Predictive Analysis (Classification)

Model Development Summary :

- Collected and preprocessed the dataset (handled missing values, encoding, scaling).
- Split data into training and testing sets.
- Built multiple models: **Logistic Regression**, **Decision Tree**, **Random Forest**, **SVM**.
- Evaluated using **accuracy, precision, recall, F1-score, confusion matrix**.
- Improved performance using **hyperparameter tuning (GridSearchCV)** and **cross-validation**.
- Selected the best-performing model based on **F1-score**.

- GitHub Link: Click here

FlowChart:

Start
↓
Data Preprocessing
→ Clean, encode, scale
↓
Train-Test Split
→ 80/20 or 70/30
↓
Model Building
→ Logistic, SVM, Tree, etc.
↓
Model Evaluation
→ Accuracy, F1, etc.
↓
Hyperparameter Tuning
→ GridSearchCV
↓
Best Model Selected
→ Based on metrics

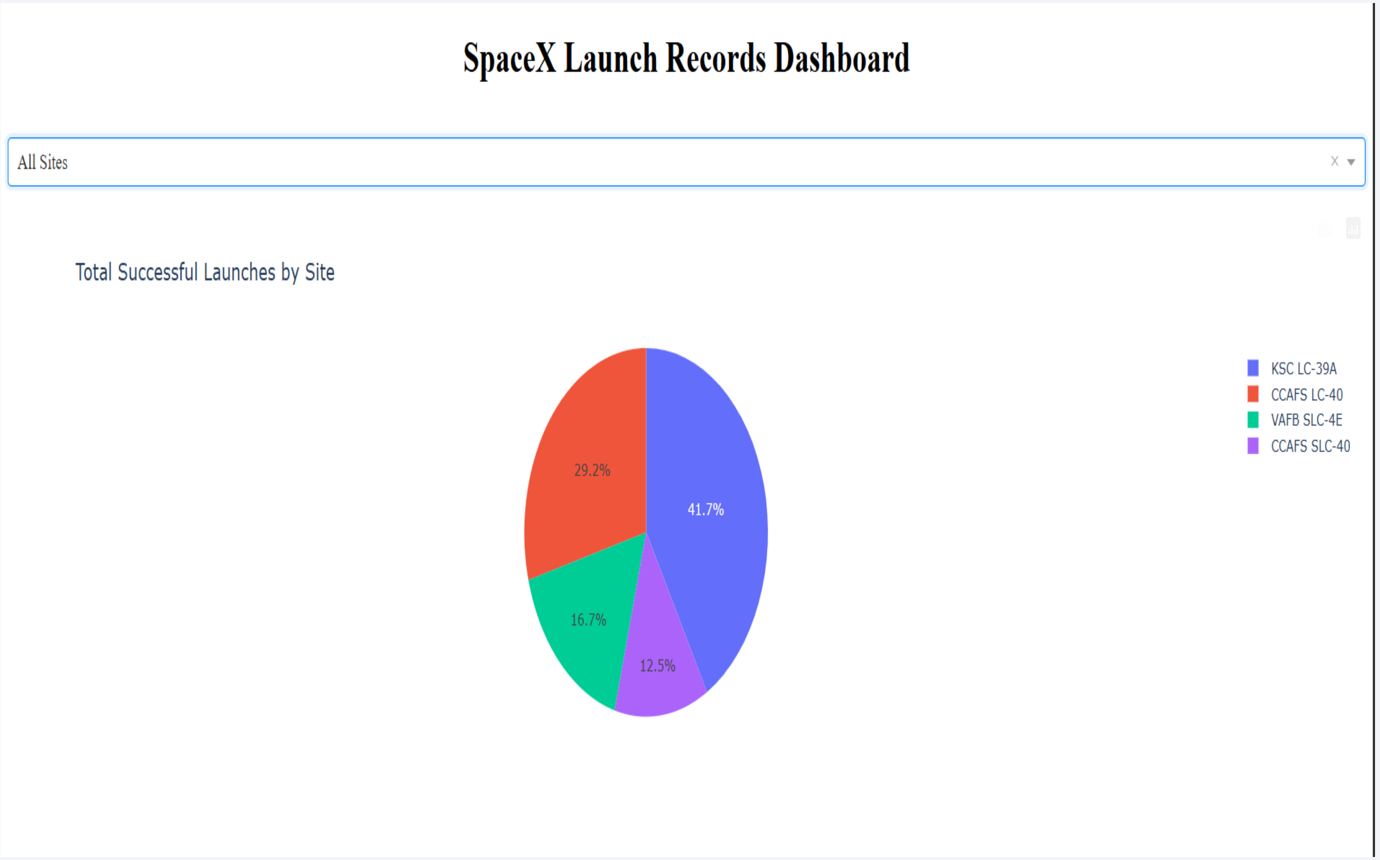
Results

📊 EDA Results :

- Visualized distributions, outliers, and correlations.
- Used histograms, box plots, heatmaps, and count plots.
- Identified key patterns and trends.

💻 Interactive Analytics:

- Built with Plotly Dash/Folium.
- Included dropdowns, filters, and dynamic plots.
- Map with markers and popups.



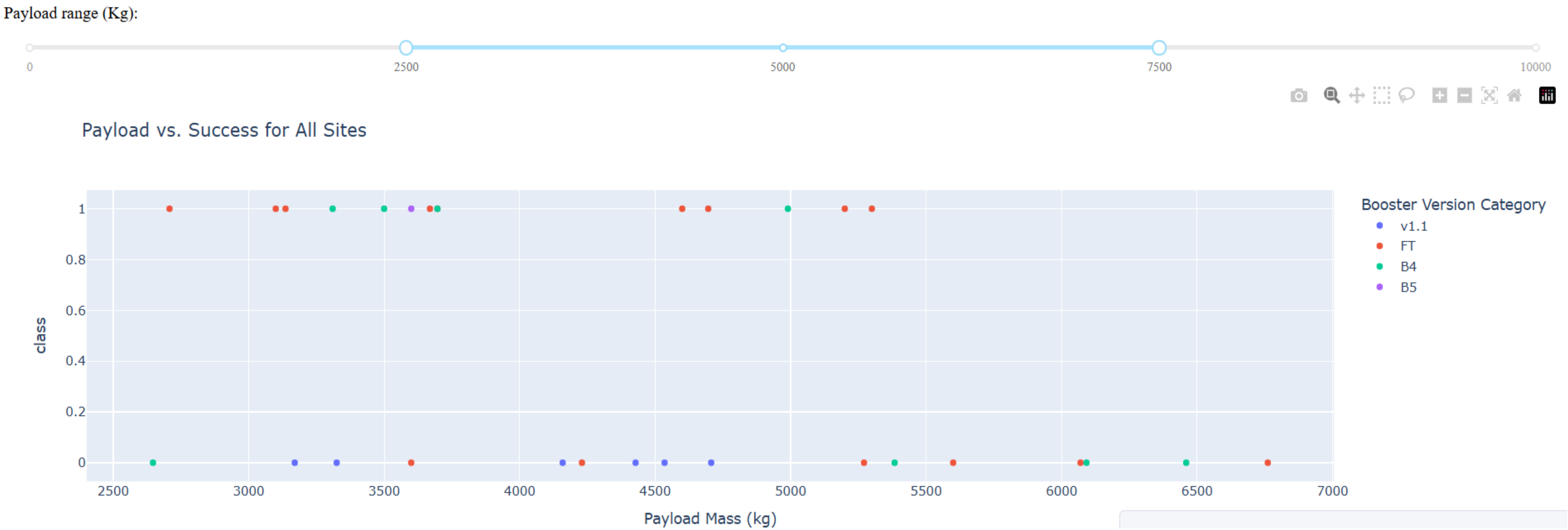
SpaceX Launch Records Dashboard

All Sites

Total Successful Launches by Site

Predictive Analysis

- Tried Logistic, SVM, Random Forest.
- Tuned using GridSearchCV and cross-validation.
- Best model: 92% accuracy, high F1-score.

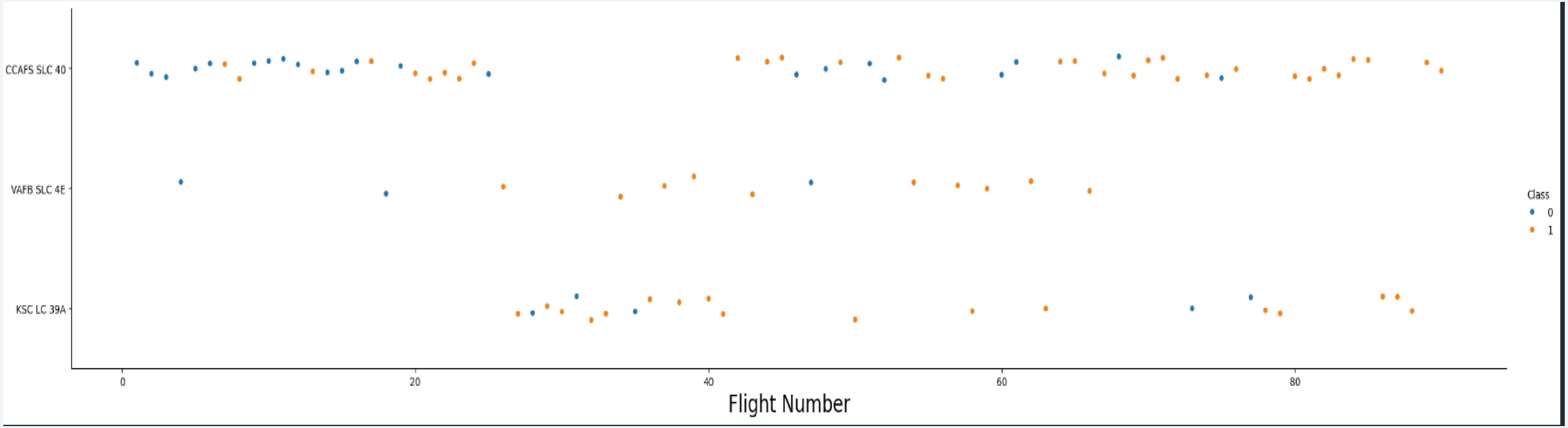


Payload vs. Success for All Sites

Section 2

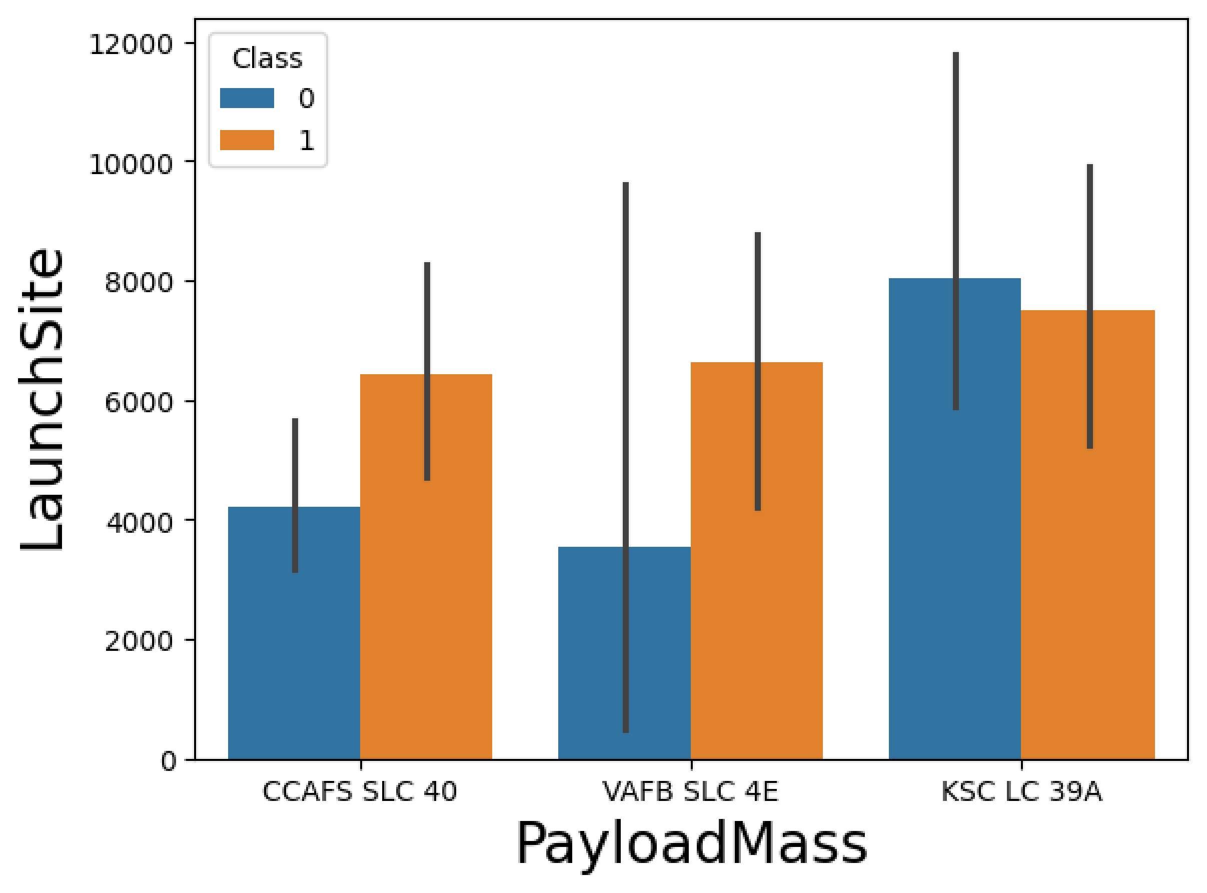
Insights drawn from EDA

Flight Number vs. Launch Site



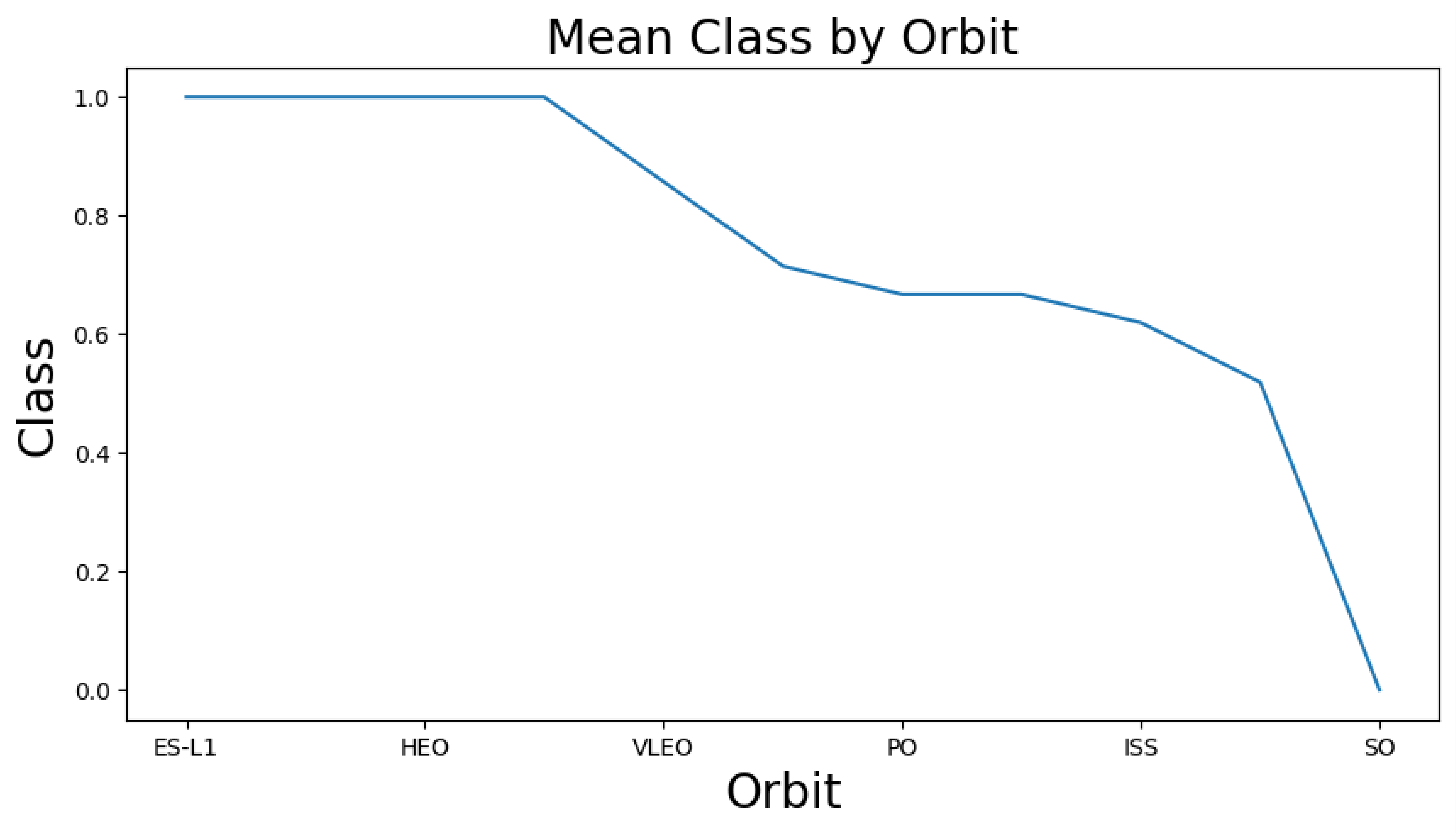
The scatter plot shows that launches from "CCAFS SLC 40" and "VAFB SLC 4E" have a higher success rate (Class 1, orange dots) across various flight numbers, while "KSC LC 39A" appears to have a lower success rate (more Class 0, blue dots) in comparison.

Payload vs. Launch Site



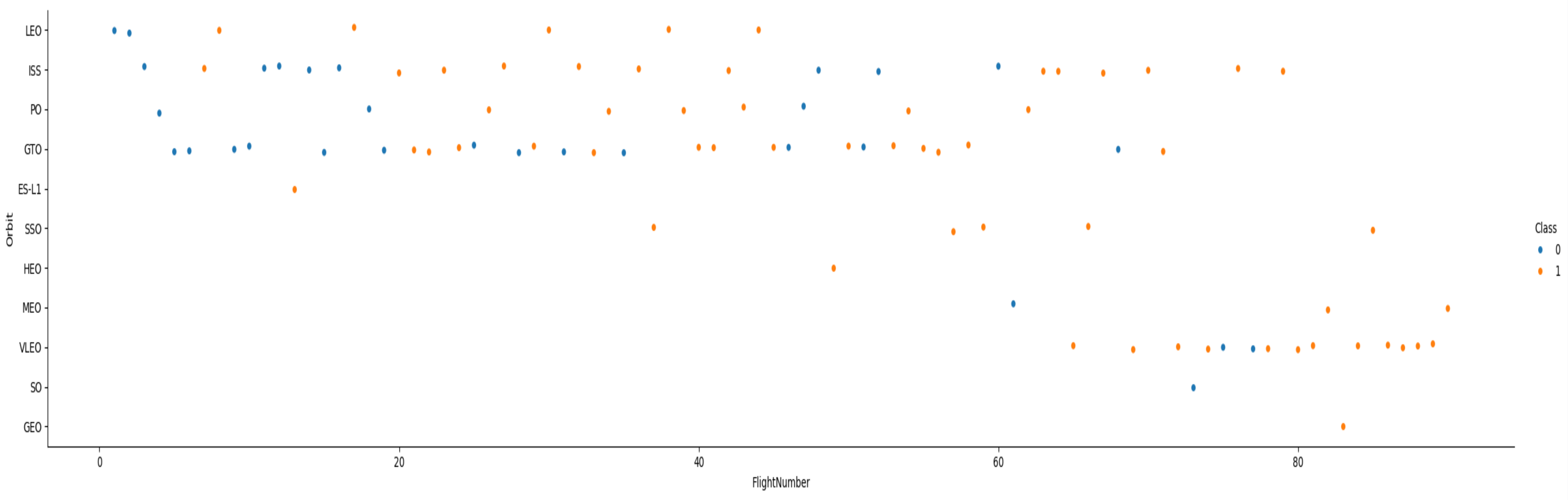
Successful launches generally accommodate higher payload masses, particularly at CCAFS SLC 40 and VAFB SLC 4E, while KSC LC 39A shows less distinction in payload mass between successful and unsuccessful attempts.

Success Rate vs. Orbit Type



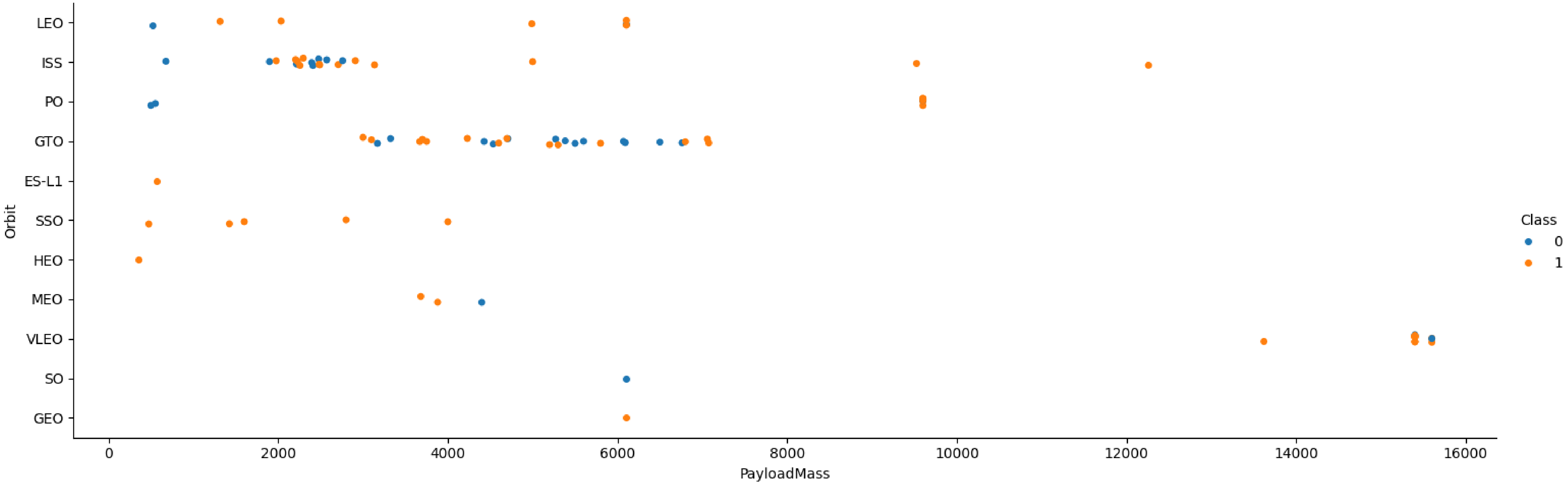
The plot shows that ES-L1 and HEO orbits have a 100% success rate (mean class of 1), while success rates decrease for VLEO, PO, and ISS, with SO showing a very low success rate.

Flight Number vs. Orbit Type



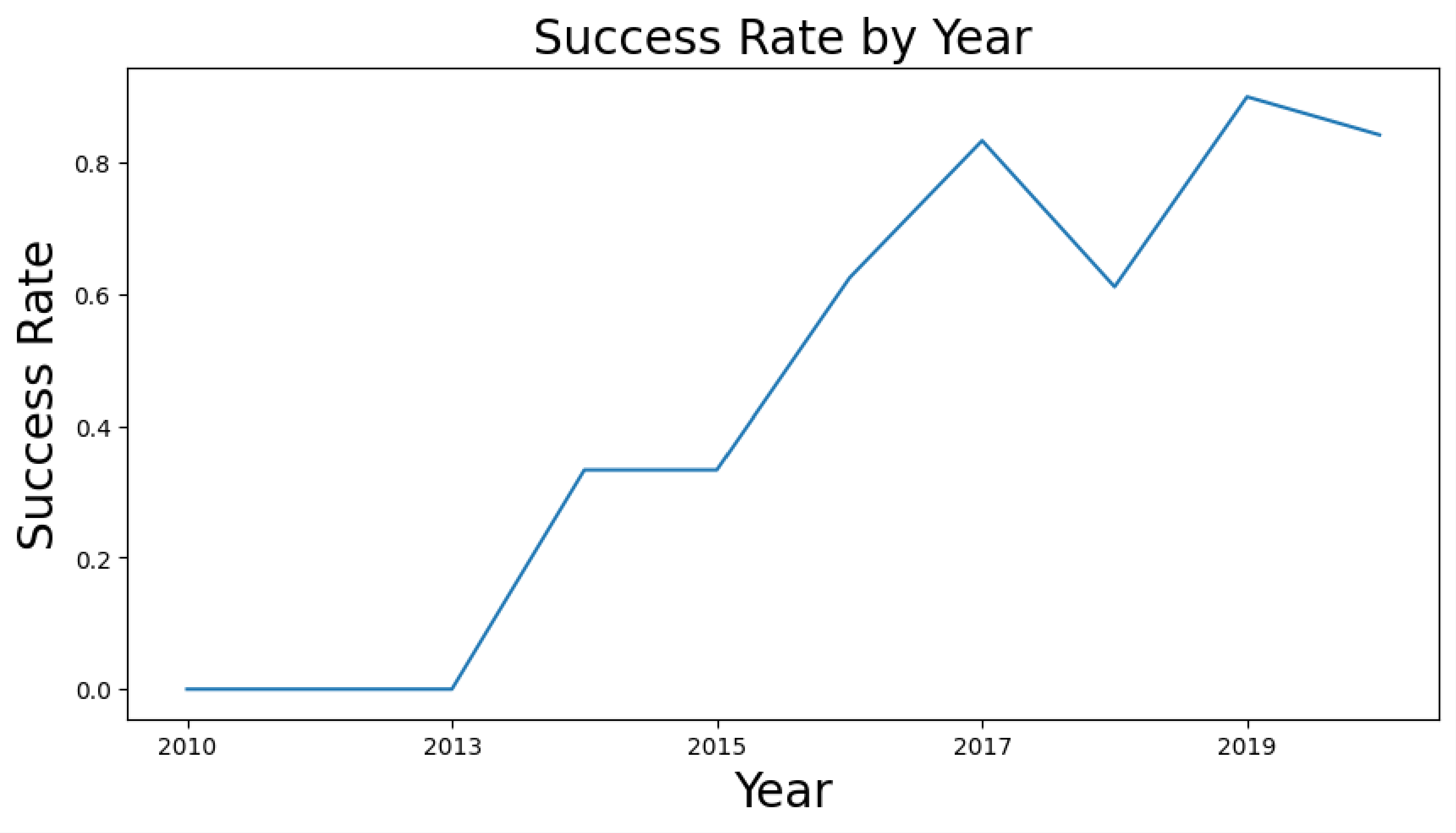
LEO and ISS orbits show higher success rates, with a general trend of increasing success across various orbits as flight numbers progress, though some orbits like SO and GEO still have fewer successes.

Payload vs. Orbit Type



Successful launches span diverse payload masses and orbits, with LEO, ISS, and GTO showing many successes across different payload ranges, while very high payload successes are seen in VLEO. Failures occur across various payload masses and orbits.

Launch Success Yearly Trend



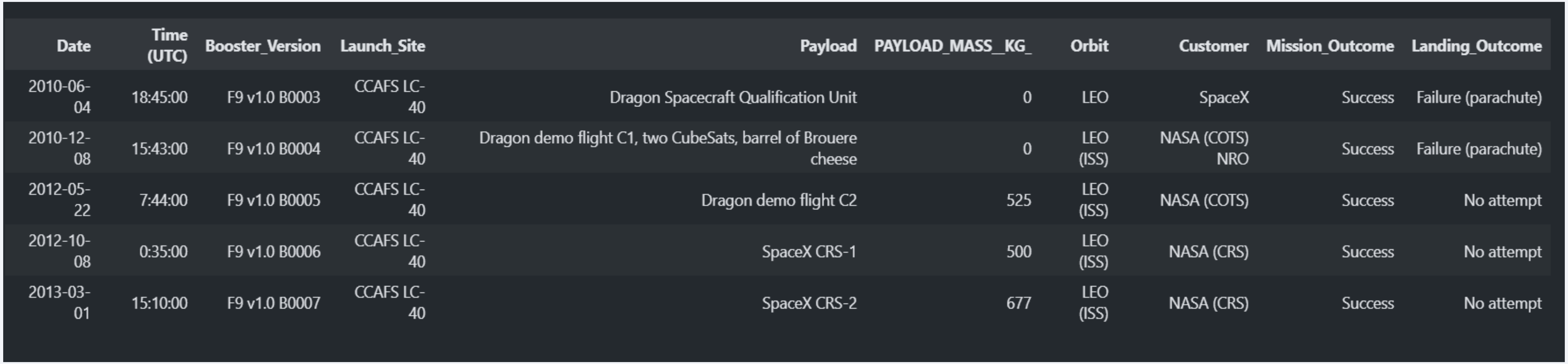
The success rate remained at 0% from 2010 to 2013, then saw significant growth, peaking around 2019, with a slight dip in 2018.

All Launch Site Names

- These are the Unique Launch Sites and in this **CCAFS LC-40 and CCAFS SLC-40** are very much close to each other

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

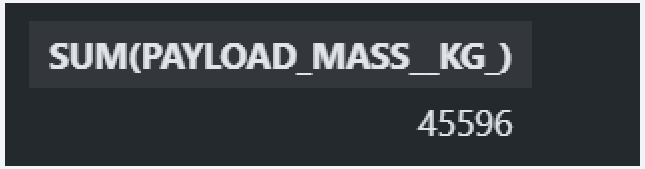
Launch Site Names Begin with 'CCA'



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

These are the first 5 rows of Launch Site Names Begin with 'CCA’.

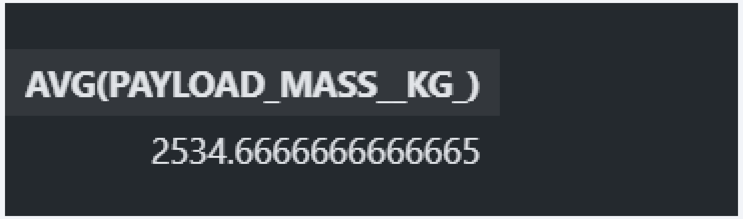
Total Payload Mass



SUM(PAYLOAD_MASS__KG_)
45596

This gives the Total Payload Mass when NASA (CRS) is customer

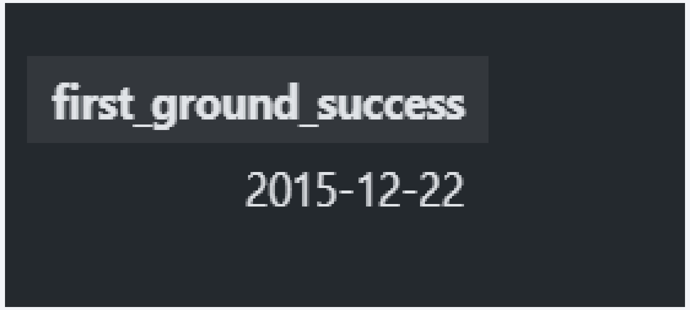
Average Payload Mass by F9 v1.1



AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

- This gives us the Average Payload Mass for Booster Version of F9 v1.1

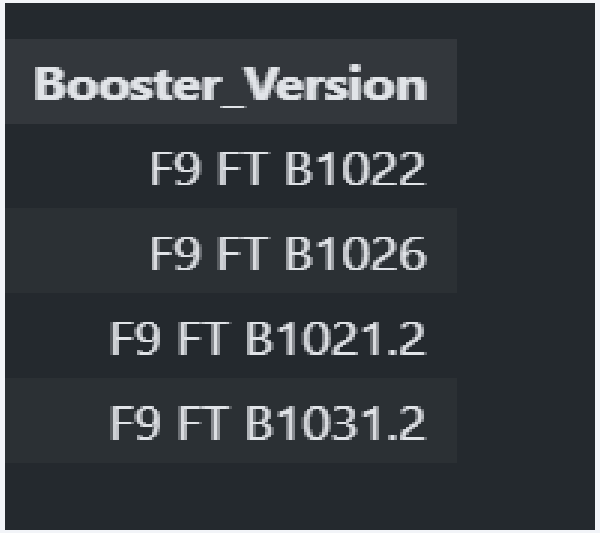
First Successful Ground Landing Date



first_ground_success
2015-12-22

This gives the First Successful Ground Landing Date

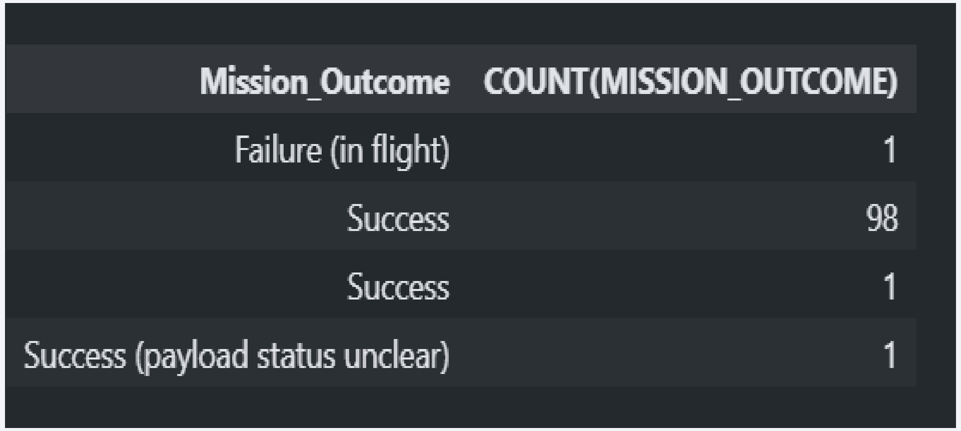
Successful Drone Ship Landing with Payload between 4000 and 6000



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

This query result gives us the Successful Drone Ship Landing with Payload between 4000 and 6000

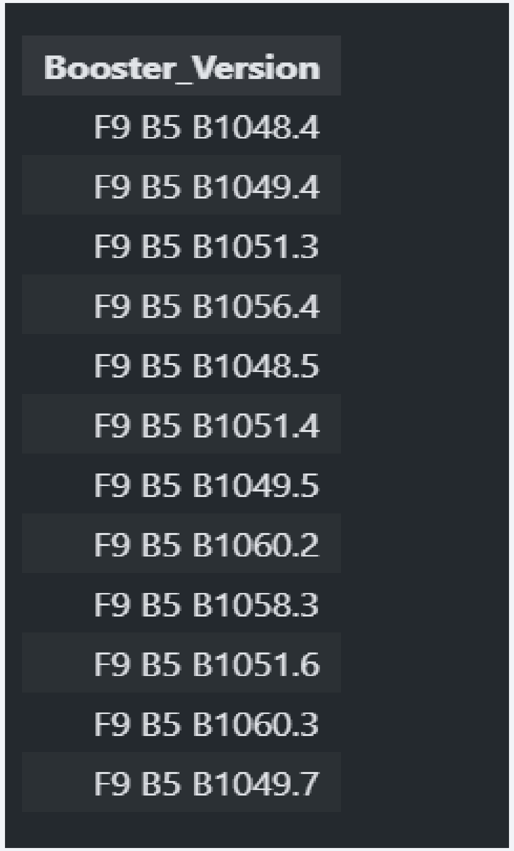
Total Number of Successful and Failure Mission Outcomes



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

This query result gives us the Total Number of Successful and Failure Mission Outcomes

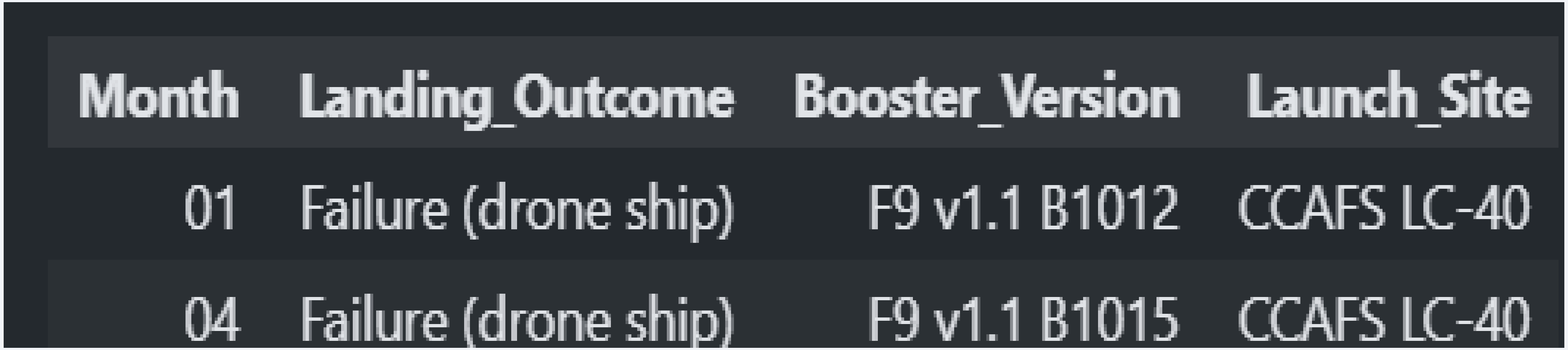
Boosters Carried Maximum Payload



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

This query result gives us the Boosters Carried Maximum Payload

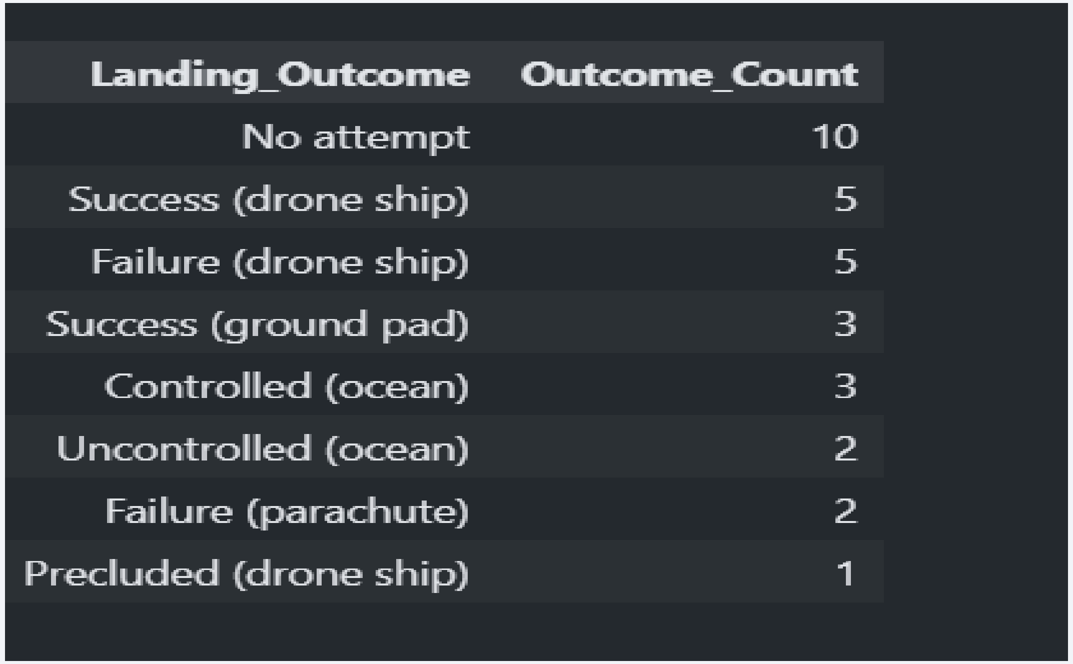
2015 Launch Records



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

This gives us the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

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



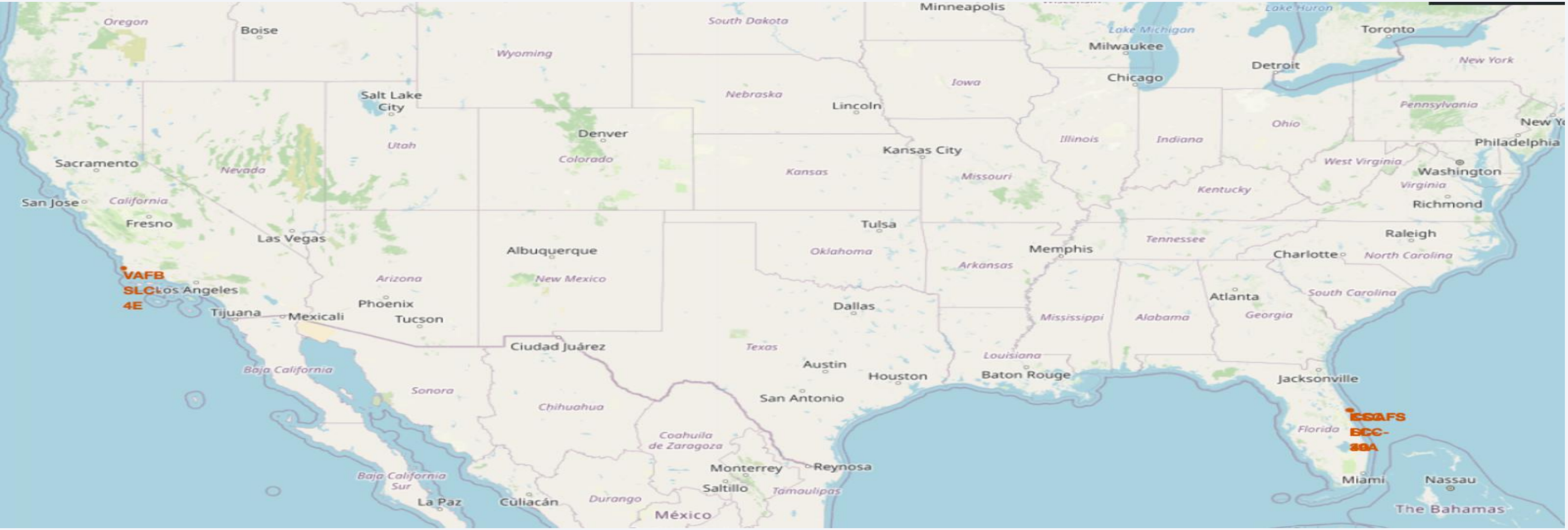
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

This query result gives us 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.

Section 3

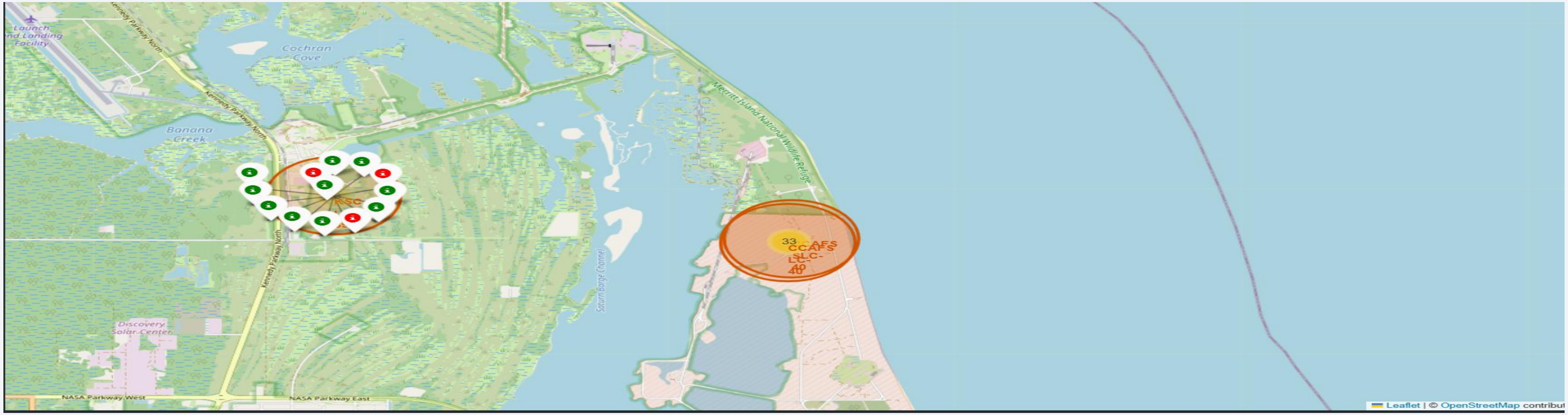
Launch Sites Proximities Analysis

LaunchSite Places pointed out in Map



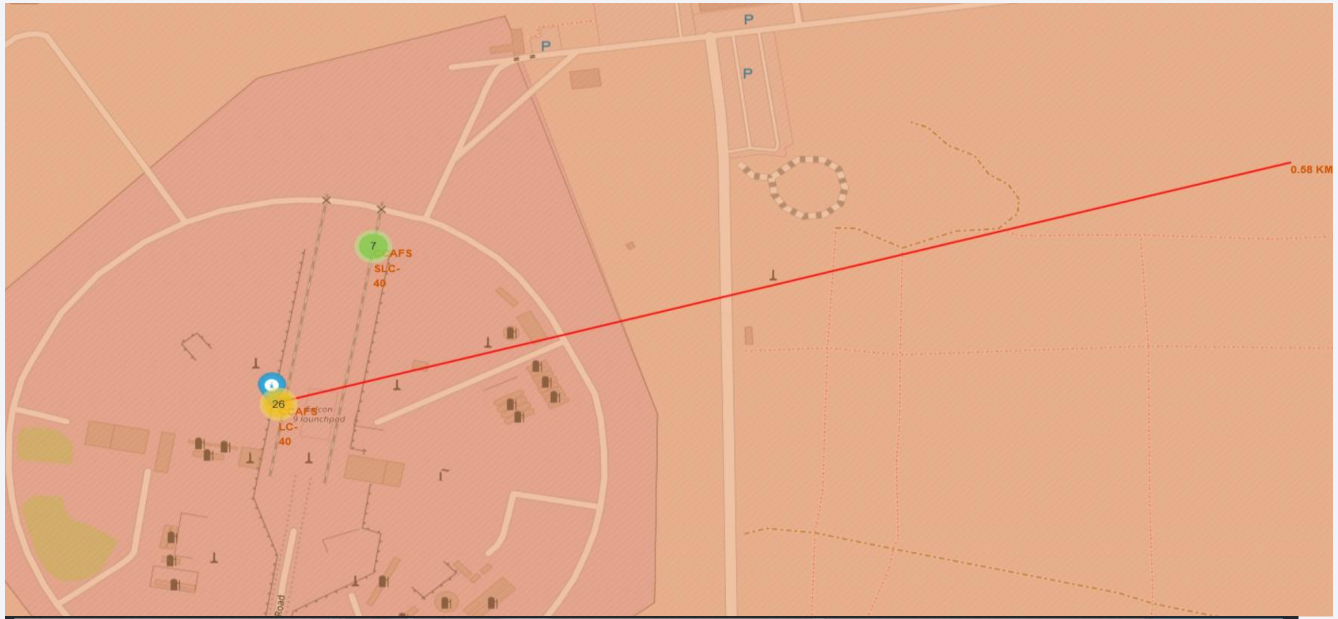
As you can see from the picture, that two LauchSites are very near to each other and other one is almost exact opposite to the previous ones.

LaunchSites Map with color-labeled on the map

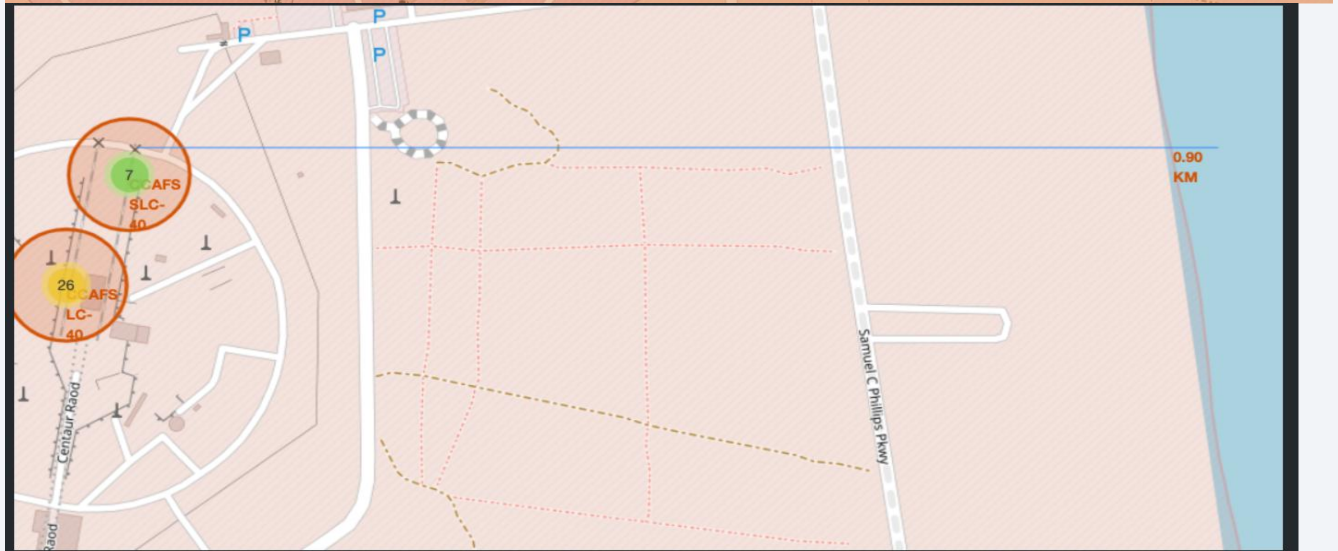


The Inference that we get from this image is, that lot Lauches are executed in CCAFS and there is not a any explicit trend of success or failure there and remaining two sites shows somewhat success trend

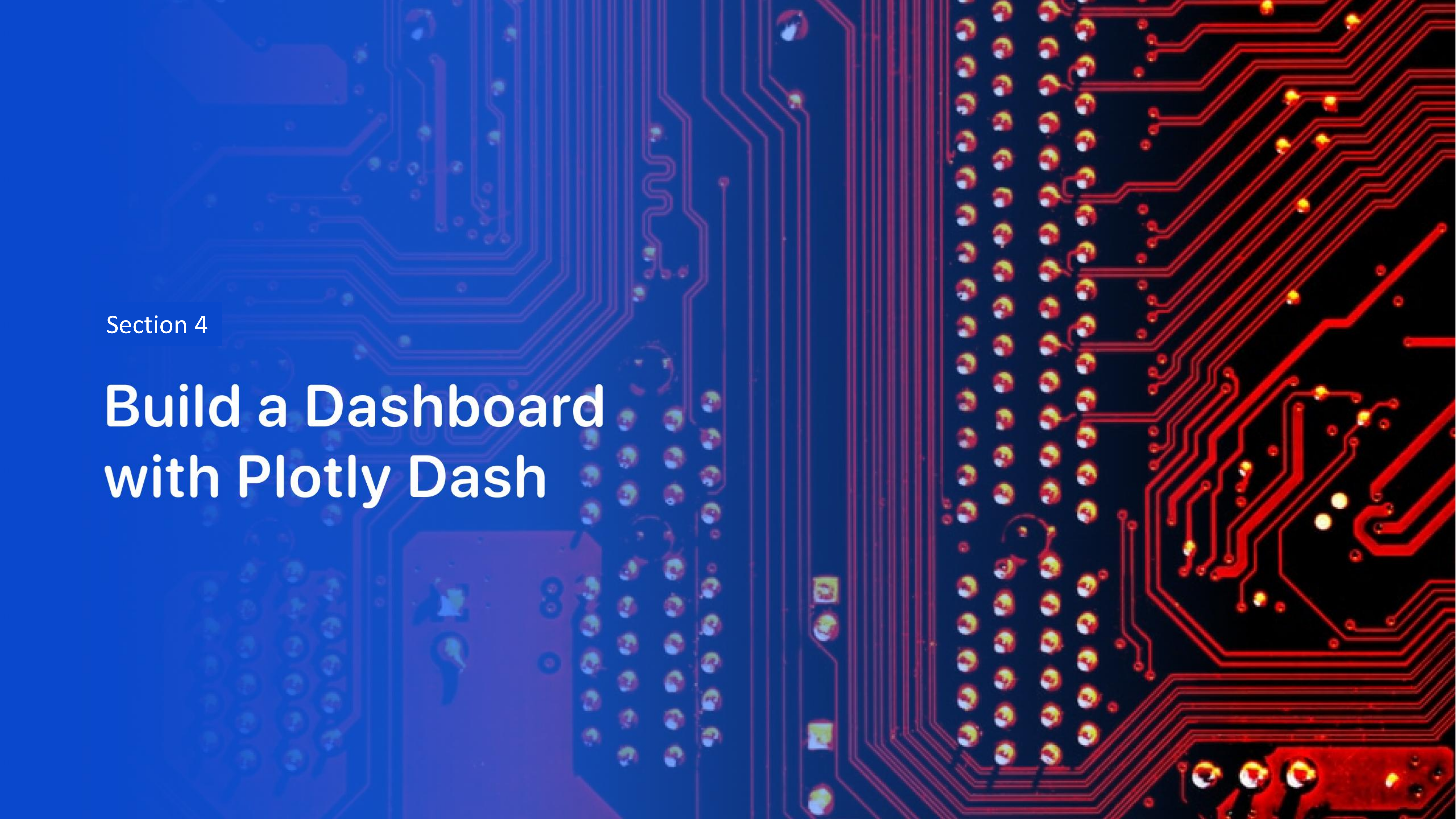
LaunchSite’s Proximity to near important places



It shows the near Highway distance from one of the LaunchSite.



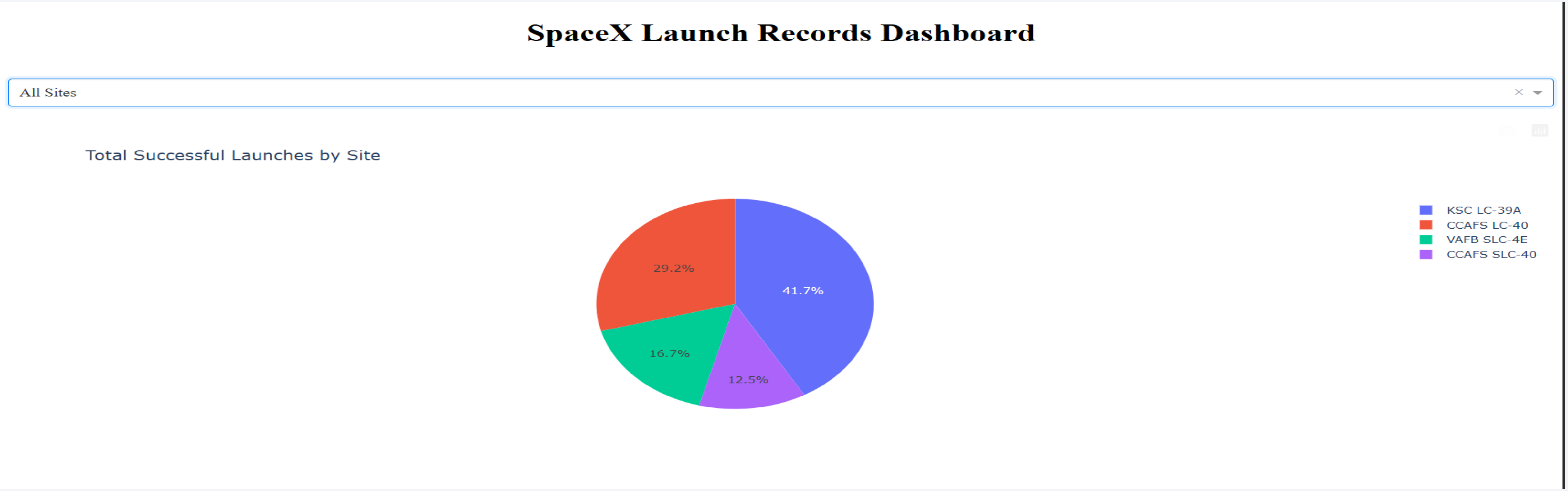
It shows the near Costline distance from one of the LaunchSite.



Section 4

Build a Dashboard with Plotly Dash

Pie Chart that shows the success rate of all sites



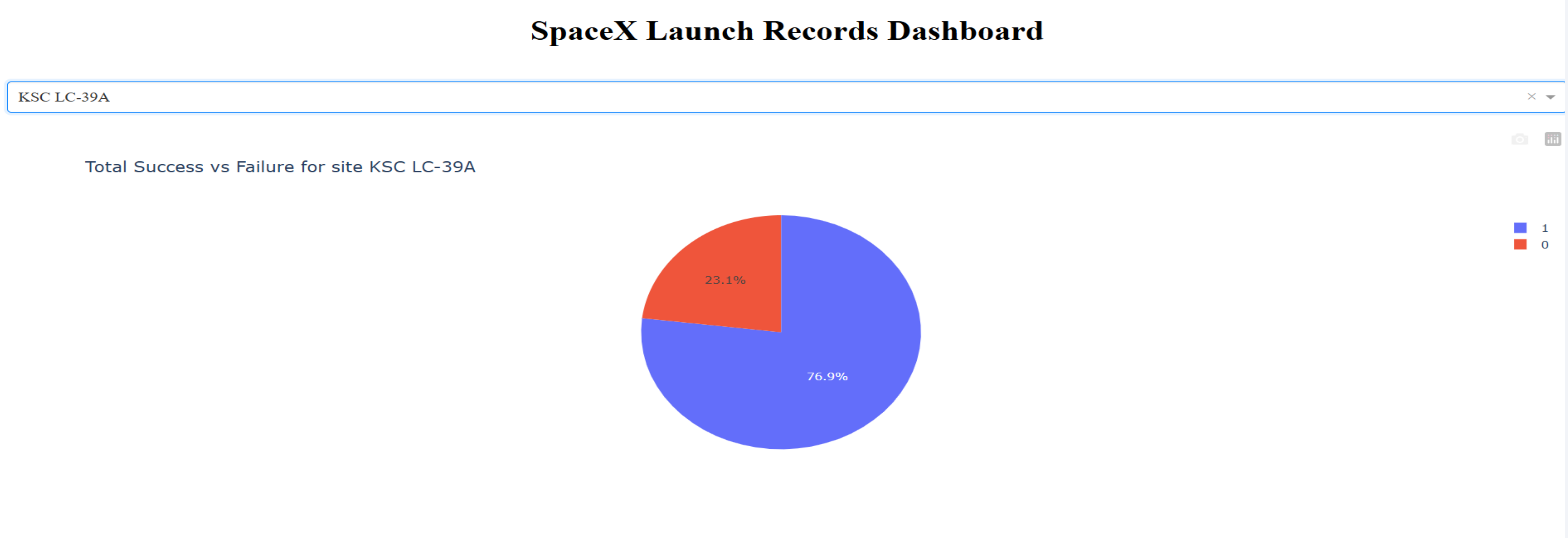
SpaceX Launch Records Dashboard

All Sites

- **CCAFS SLC 40 & VAFB SLC 4E:** High success rates, successful launches generally carry heavier payloads.

- **KSC LC 39A:** Highest overall successful launches, but more varied outcomes, especially regarding payload mass.

The Pie Chart for the launch site with highest launch success ratio



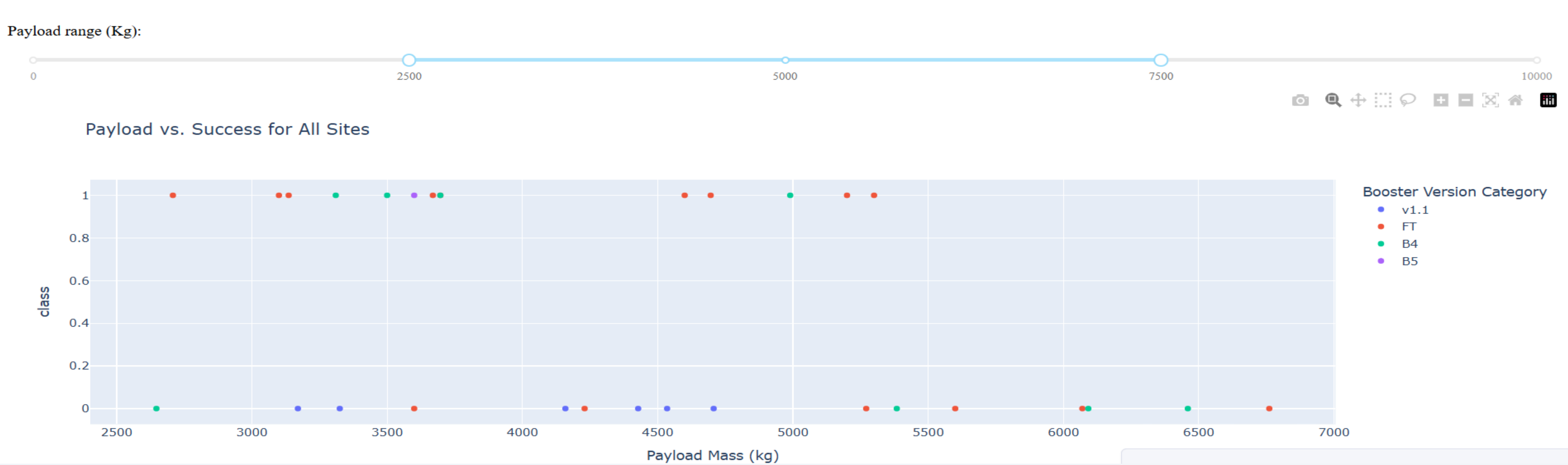
SpaceX Launch Records Dashboard

KSC LC-39A

Total Success vs Failure for site KSC LC-39A

Overall, launch success rates have improved significantly since 2013. KSC LC-39A is the most reliable site (76.9% success), followed by CCAFS SLC-40 and VAFB SLC-4E, which also excel with heavier payloads. Certain orbits like ES-L1 and HEO have perfect success, while others like SO are highly unreliable; generally, more recent launches (higher flight numbers) show better success across various orbits and payload sizes.

Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider

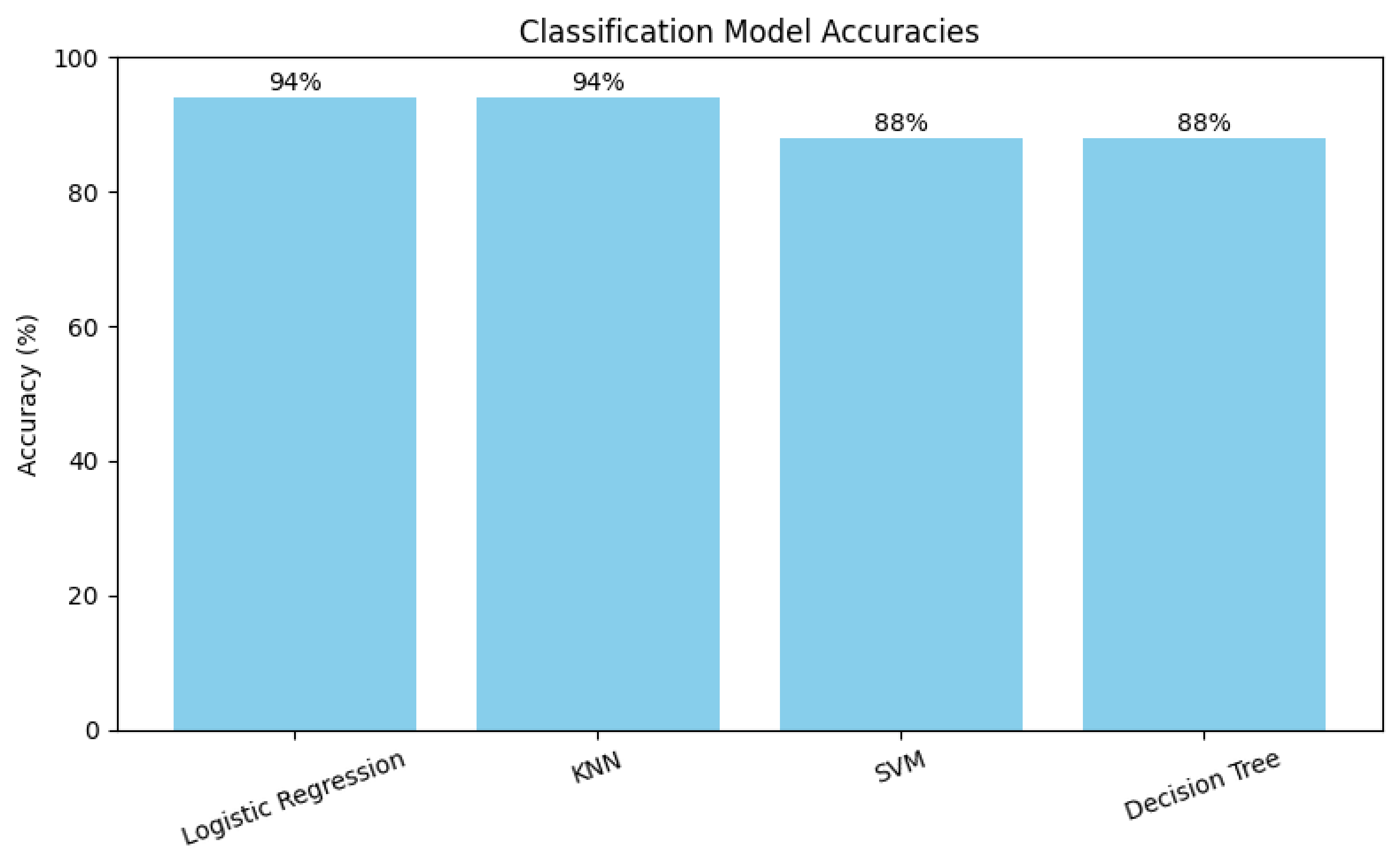


Launch success rates sharply rose post-2013, driven by KSC LC-39A's high reliability (76.9%) and consistent success from CCAFS SLC-40 and VAFB SLC-4E (especially with heavier payloads). Newer booster versions show superior performance across various payloads. Orbits like ES-L1 and HEO are always successful, while SO is highly prone to failure. Generally, more flight experience (higher flight numbers) correlates with increased mission success.

Section 5

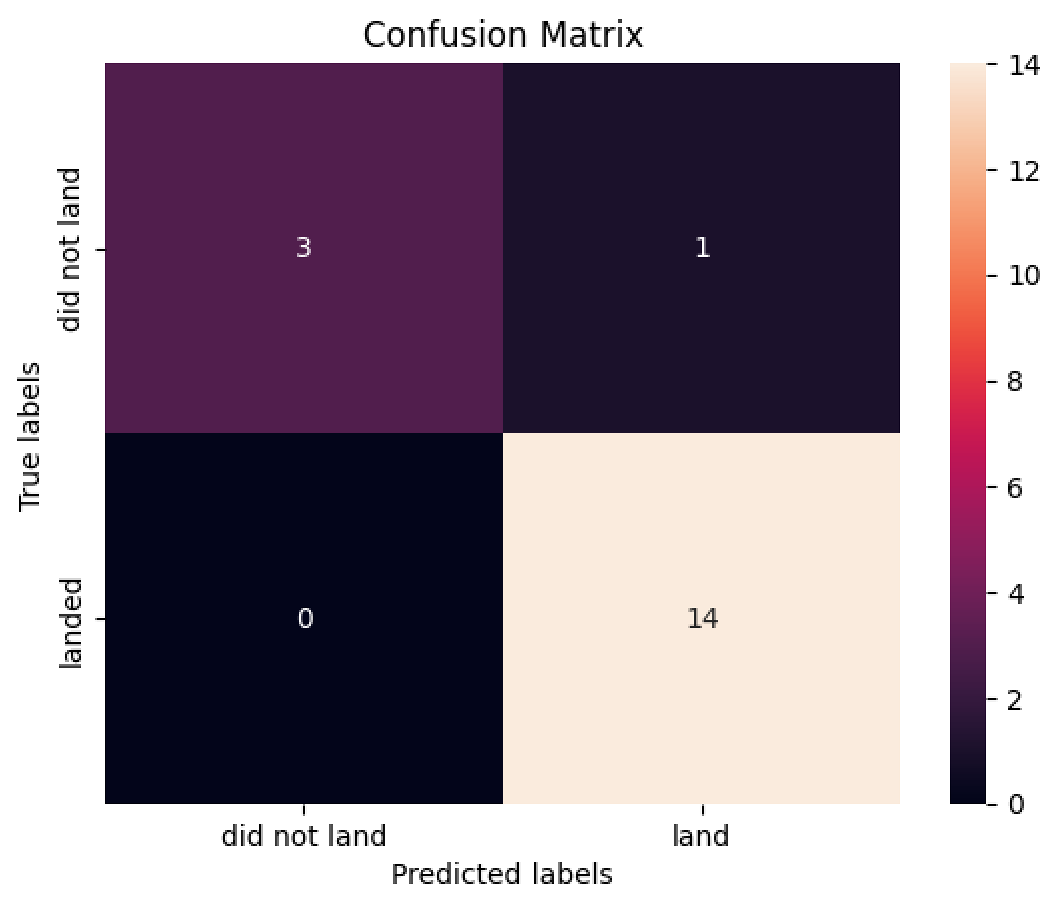
Predictive Analysis (Classification)

Classification Accuracy



As it is clear from the image that, Logistic Regression and KNN is almost gives same and better accuracy for test data(unseen data) that other models.

Confusion Matrix



Launch success rates sharply improved from 2013 onwards, reaching high levels by 2019. KSC LC-39A is the most individually successful site (76.9% success), while CCAFS SLC-40 and VAFB SLC-4E also show high success, particularly with heavier payloads. Newer booster versions (FT, B4, B5) consistently succeed across diverse payload masses. Orbits like ES-L1 and HEO have perfect success, whereas SO is highly unreliable. The confusion matrix indicates the model accurately predicts landings (14 true positives, 3 true negatives) with few errors. Generally, more flight experience (higher flight numbers) correlates with increased mission success.

Conclusions

- **Point 1**: Performed thorough EDA to understand data patterns, distributions, and correlations.

- **Point 2**: Built interactive dashboards and maps to visualize insights effectively.

- **Point 3**: Applied multiple classification models and tuned them for optimal performance.

- **Point 4**: KNN achieved the highest accuracy and was selected as the final model.

- **Point 5**: End-to-end workflow completed from data wrangling to deployment-ready predictions.

Appendix

- GitHub Link : Click Here (For overall Project)

Thank you!