IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nicolas Palacio
Sept 2023

Outline

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

Executive Summary

- Methodology:
  - Data about SapaceX launches was collected using both API and Web scraping techniques. The data was exported and explored using SQL and visualization techniques with Python. To gain a deeper understanding of the data we used both folium and dash to build maps and interactive dashboards. Finally, different machine learning models were trained on the normalized data, evaluated, and compared with one another to find the best performing model.
- Results:
  - Four machine learning models were produced: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. All produced similar results with accuracy rate of about 83.33%. All models over predicted successful landings. More data is needed for better model determination and accuracy.

Introduction

The advent of the Commercial Space Age has transformed the space industry.

- SpaceX has emerged as a key player, known for its competitive pricing and rocket recovery capabilities.
- In response to SpaceX's success, Space Y aims to compete by enhancing its Stage 1 recovery process.
- Problem Statement:
  - Space Y has tasked our team with developing a machine learning model capable of predicting the success of Stage 1 rocket recovery.
- Objective:
  - In this presentation, we will outline our approach, methodology, and the results obtained from training and evaluating machine learning models. We will also discuss the implications of our findings and potential areas for further research.

Methodology

Methodology

Summary

- Data collection methodology:
  - API and Web Scraping Techniques
- Perform data wrangling
  - Python for standardization and normalization
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Logistic Regression, SVM, Decision Tree, KNN

Data Collection

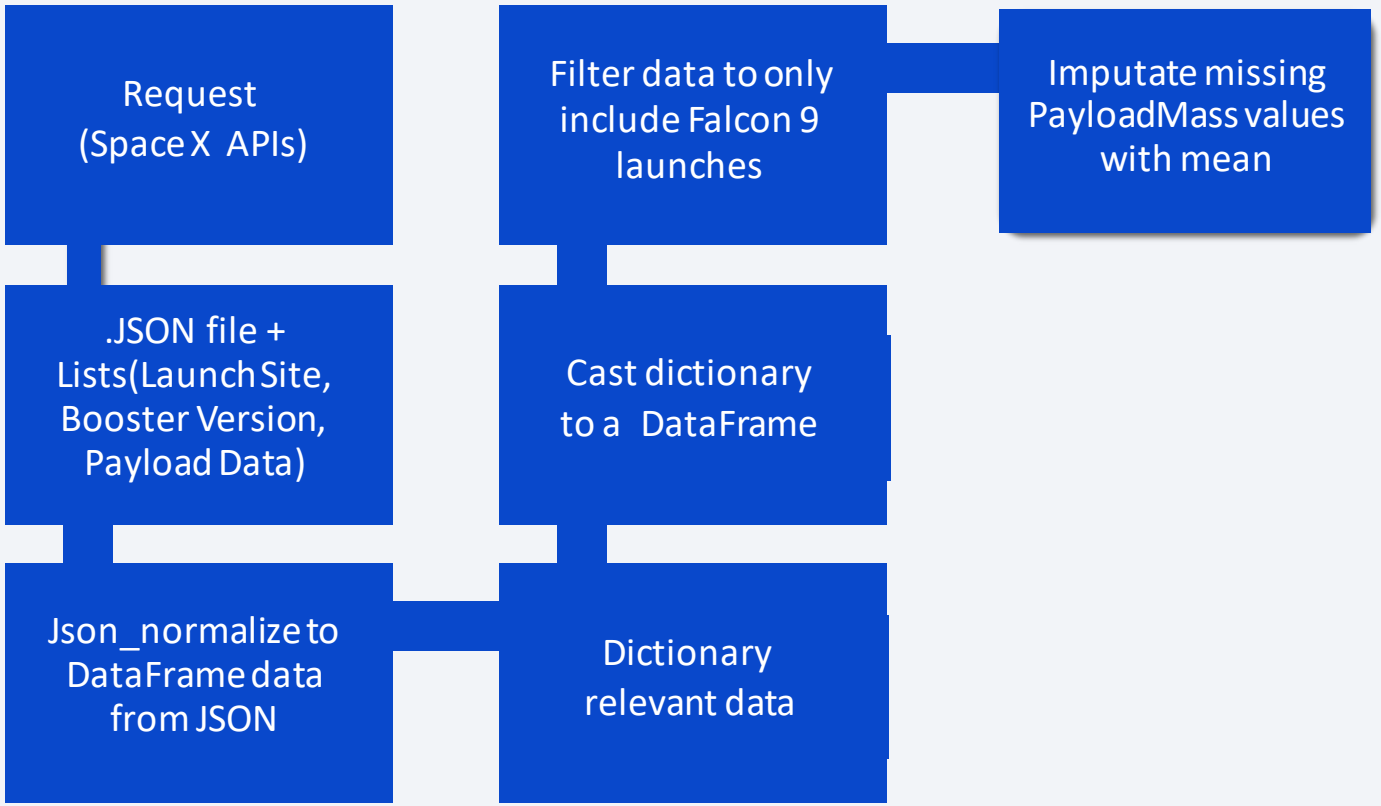
The data collection process was a two-fold approach, combining API requests from the SpaceX public API with web scraping techniques to extract information from a table in SpaceX's Wikipedia entry.

On the next slide, we will present a flowchart illustrating the process of data collection from the API, and following that, we will showcase a flowchart depicting the data collection through web scraping.

Data Collection – SpaceX API

- Data collection with SpaceX REST calls flowchart

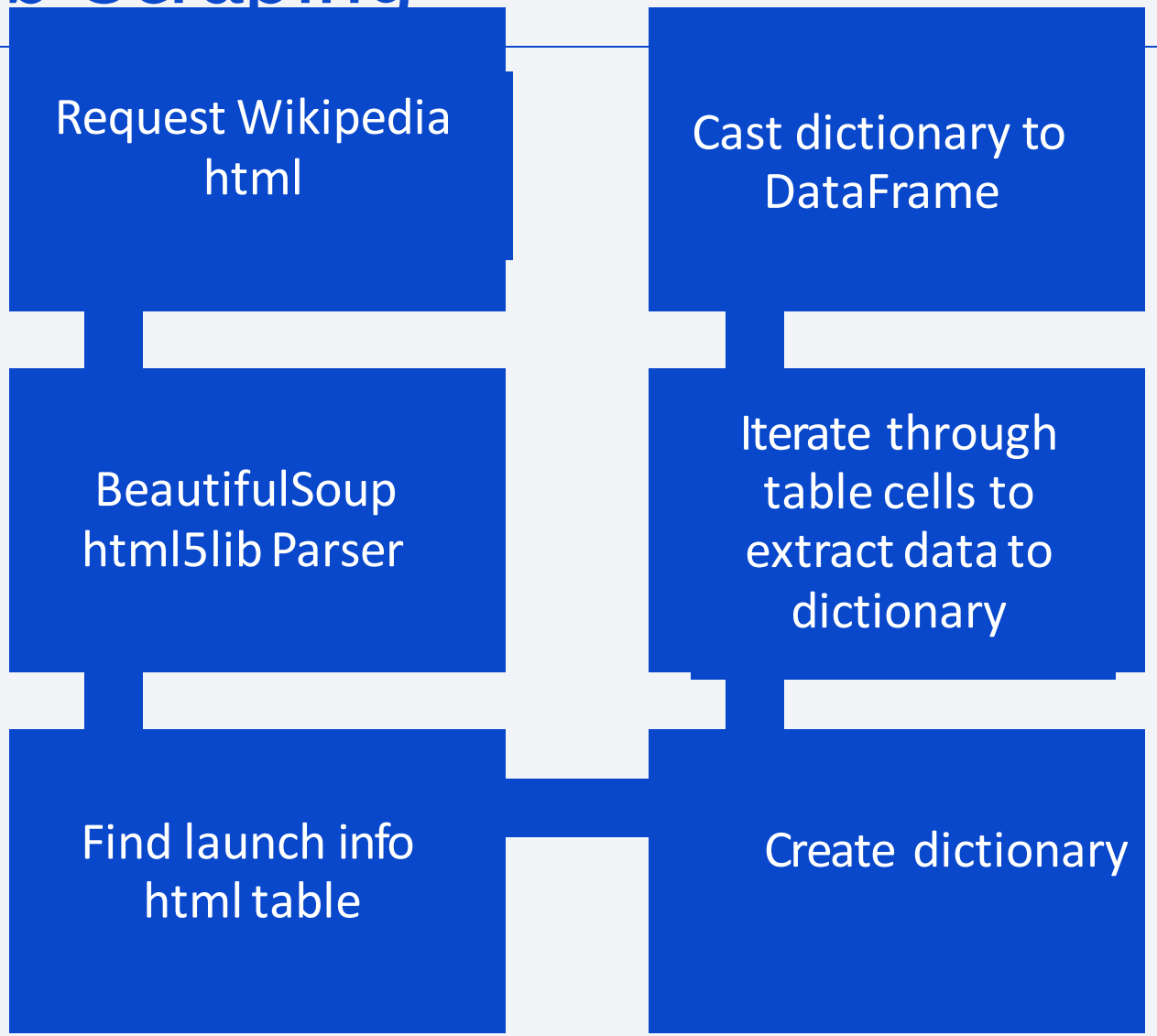
- Github: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Web Scraping

- Web scraping process flowchart

- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Generate a training label based on the landing outcomes, where "successful" is represented as 1, and "failure" as 0.
- The "Outcome" column comprises two elements: "Mission Outcome" and "Landing Location."
- Introduce a new training label column, denoted as "class," with a value of 1 if "Mission Outcome" is deemed True, and 0 otherwise. The value mapping is as follows:
- For "True ASDS," "True RTLS," and "True Ocean," set the value to 1.
- For "None None," "False ASDS," "None ASDS," "False Ocean," and "False RTLS," set the value to 0.

- GitHub : https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_1_L3_labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

- Conducted Exploratory Data Analysis (EDA) on the following variables: Flight Number, Payload Mass, Launch Site, Orbit, Class, and Year.
- Utilized various plots, including scatter plots, line charts, and bar plots, to explore relationships between these variables. These visualizations included:
- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload Mass vs. Launch Site
- Orbit vs. Success Rate
- Flight Number vs. Orbit
- Payload vs. Orbit
- Yearly Trend of Success Rates
- The objective was to identify potential relationships among these variables to determine their suitability for training the machine learning model.
- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- Performed SQL queries using Python integration to extract valuable insights from the dataset.
- Executed queries to gain a deeper understanding of the data, including retrieving information about launch site names, mission outcomes, different payload sizes of customers, booster versions, and landing outcomes.
- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Folium maps are used to pinpoint Launch Sites, both successful and unsuccessful landings, as well as demonstrate the proximity of key locations such as Railway, Highway, Coast, and City.

- These maps help us gain insights into the reasons behind the selection of specific launch site locations and visualize the geographical distribution of successful landings in relation to these key landmarks.
- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- The dashboard comprises a pie chart and a scatter plot.
- The pie chart allows users to explore the distribution of successful landings among all launch sites collectively or individually by selecting specific launch sites.
- The scatter plot takes two inputs: either data from all sites or data from an individual site, and allows users to adjust the payload mass using a slider ranging from 0 to 10,000 kg.
- The pie chart serves as a visual representation of the success rate at launch sites, while the scatter plot provides insights into how success rates differ across launch sites, payload masses, and booster version categories.
- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- **Model Selection:** We explored several classification algorithms, including Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K-Nearest Neighbors.
- **Hyperparameter Tuning:** For each model, we performed hyperparameter tuning using techniques like GridSearchCV to find the optimal combination of hyperparameters.
- **Model Training:** We trained each model using our preprocessed dataset, and we utilized cross-validation techniques to ensure robustness and avoid overfitting.
- **Evaluation Metrics:** We assessed model performance using various evaluation metrics such as accuracy, precision, recall, F1-score, and ROC curves. These metrics helped us understand how well each model was performing.
- **Model Comparison:** We compared the performance of all models based on evaluation metrics to identify the best candidate.

- GitHub: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

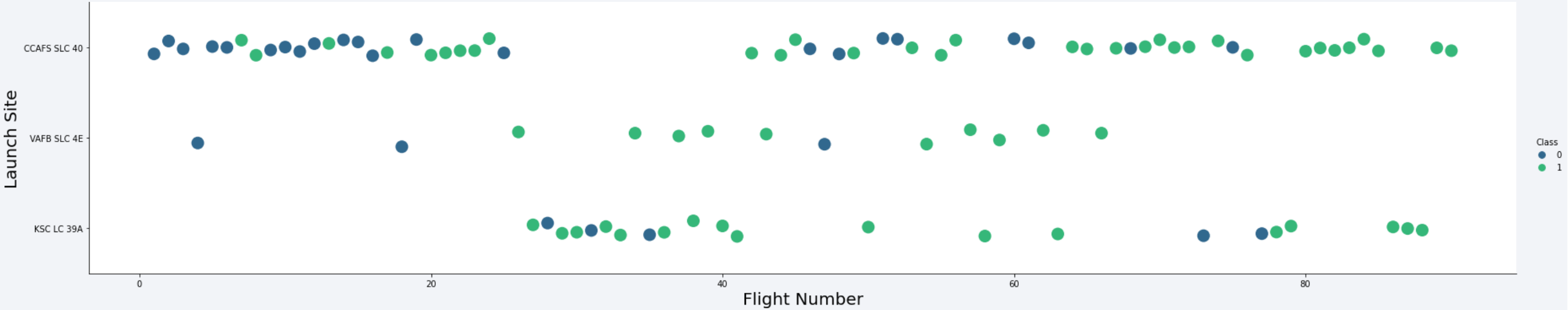
Results

- **Exploratory Data Analysis (EDA):** EDA provided valuable insights into the dataset. Notable findings include:
  - Payload mass and launch site had varying impacts on mission success.
  - Certain launch sites exhibited higher success rates.
  - Yearly trends showed an increasing number of successful missions over time.
- **Dashboard:** Our interactive dashboard featured a pie chart showing the distribution of successful landings across launch sites and a scatter plot allowing exploration of success rates based on payload mass and booster version category.
- **Predictive Analysis:** We developed and evaluated four classification models—Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K-Nearest Neighbors—achieving an accuracy rate of approximately 83.33%. All models tended to overpredict successful landings, suggesting a need for additional data to improve accuracy.

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site



Green indicates successful launch; Purple indicates unsuccessful launch.

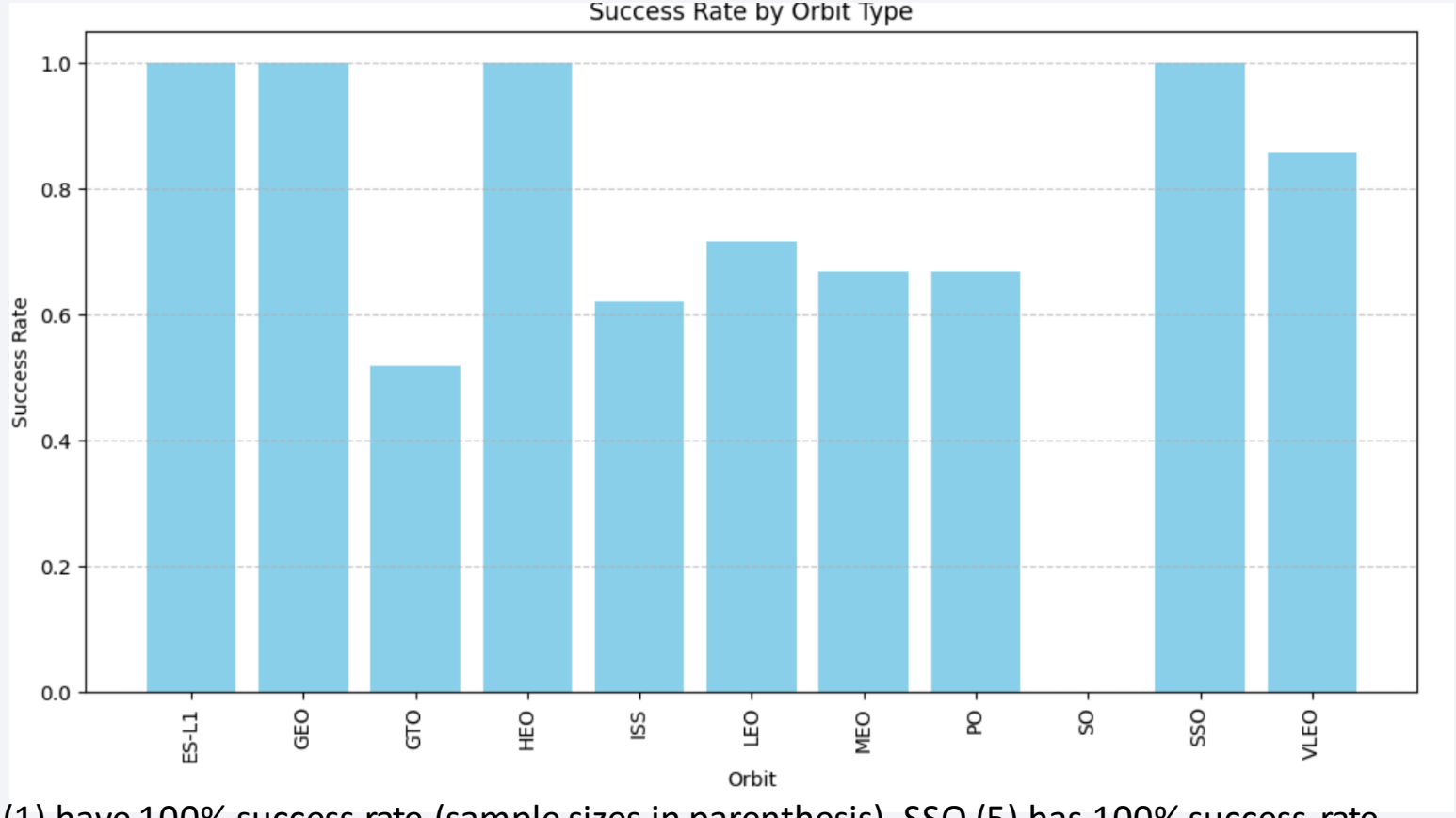
Graphic suggests an increase in success rate over time (indicated in Flight Number). Likely a big breakthrough around flight 20 which significantly increased success rate. CCAFS appears to be the main launch site as it has the most volume.

Payload vs. Launch Site

Green indicates successful launch; Purple indicates unsuccessful launch.

Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type



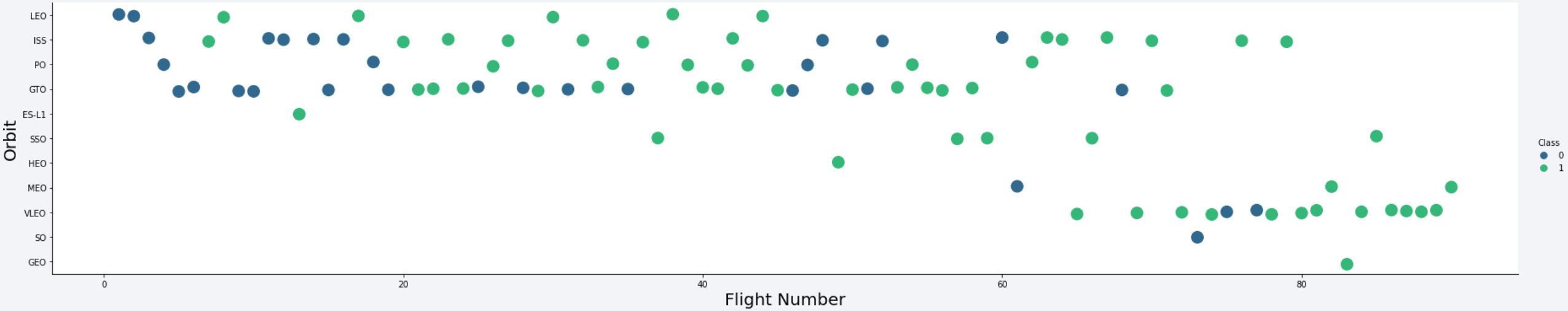
ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate

VLEO (14) has decent success rate and attempts

SO (1) has 0% success rate

GTO (27) has the around 50% success rate but largest sample

Flight Number vs. Orbit Type



Green indicates successful launch; Purple indicates unsuccessful launch.

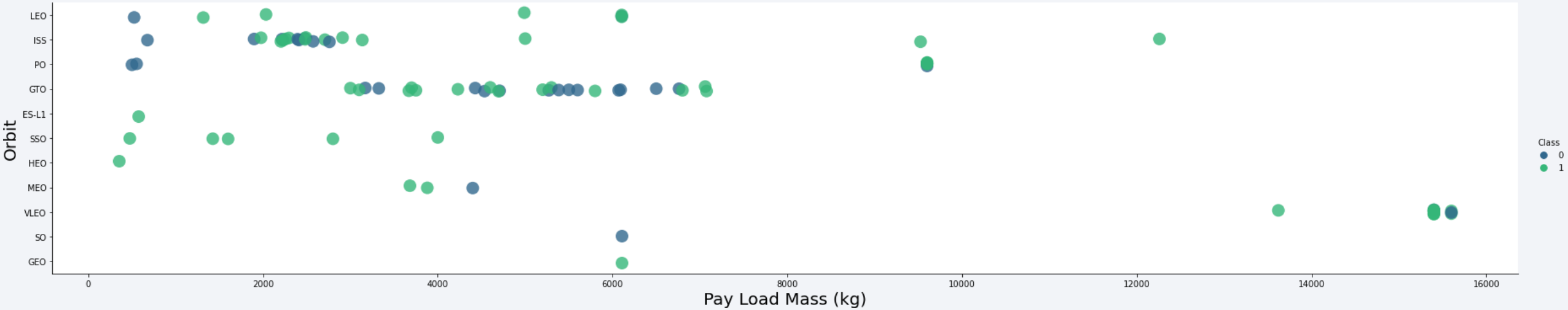
Launch Orbit preferences changed over Flight Number.

Launch Outcome seems to correlate with this preference.

SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches

SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

Payload vs. Orbit Type



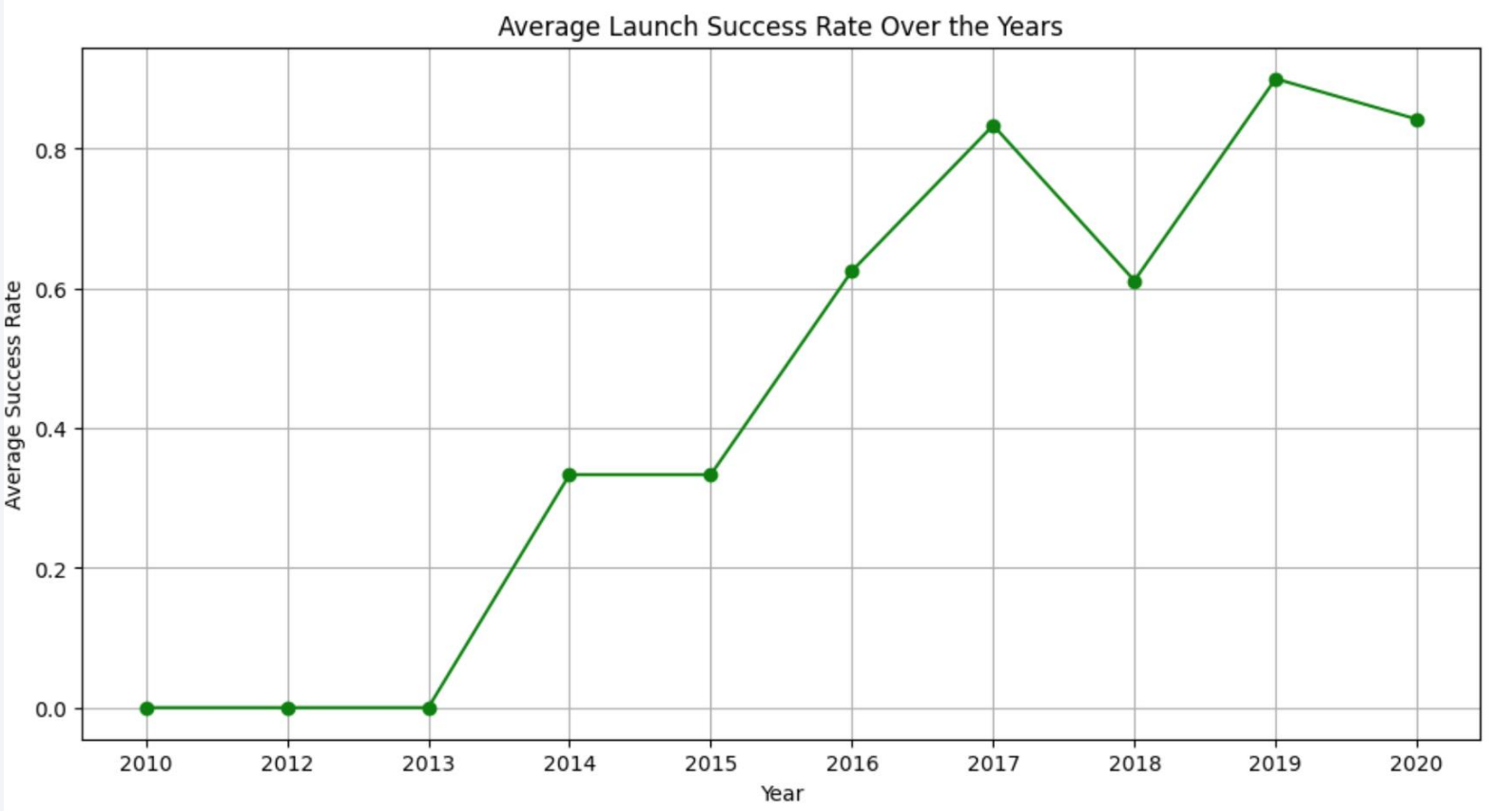
Green indicates successful launch; Purple indicates unsuccessful launch.

Payload mass seems to correlate with orbit

LEO and SSO seem to have relatively low payload mass

The other most successful orbit VLEO only has payload mass values in the higher end of the range

Launch Success Yearly Trend



Success generally increases over time since 2013 with a slight dip in 2018

Success in recent years at around 80%

All Launch Site Names

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

In [11]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

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

Out[11]:

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

Query unique launch site names from database.

Launch Site Names Begin with 'CCA'

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

In [12]:

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

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

Out[12]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Lar
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Fai
2010-08-12	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	Fai
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	

First five entries in database with Launch Site name beginning with CCA.

Total Payload Mass

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

In [17]:
```
%%sql
SELECT SUM("PAYLOAD_MASS__KG_")
FROM SPACEXTABLE
WHERE "Customer" LIKE "NASA (CRS)%";
```

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

Out[17]:

SUM("PAYLOAD_MASS__KG_")
48213

This query sums the total payload mass in kg where NASA was the customer.

CRS stands for Commercial Resupply Services which indicates that these payloads were sent to the International Space Station (ISS).

Average Payload Mass by F9 v1.1

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

In [18]:
```
%%sql
SELECT AVG("PAYLOAD_MASS__KG_")
FROM SPACEXTABLE
WHERE "Booster_Version" LIKE "F9 v1.1";
```

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

Out[18]:

AVG("PAYLOAD_MASS__KG_")
2928.4

This query calculates the average payload mass or launches which used booster version F9 v1.1

Average payload mass of F9 1.1 is on the low end of our payload mass range

First Successful Ground Landing Date

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

Hint:Use min function

In [19]:

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

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

Out[19]:

MIN("Date")
2015-12-22

This query calculates the average payload mass or launches which used booster version F9 v1.1

Average payload mass of F9 1.1 is on the low end of our payload mass range

Successful Drone Ship Landing with Payload between 4000 and 6000

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

In [20]:

```
%%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.
```

Out[20]:

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

This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 noninclusively.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [23]:
```
%%sql
SELECT "Mission_Outcome", COUNT(*) AS "Count"
FROM SPACEXTABLE
GROUP BY "Mission_Outcome";
```

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

Out[23]:

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

This query returns a count of each mission outcome.

SpaceX appears to achieve its mission outcome nearly 99% of the time.

This means that most of the landing failures are intended.

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [24]:
```
%%sql
SELECT DISTINCT "Booster_Version"
FROM SPACEXTABLE
WHERE "PAYLOAD_MASS__KG_" = (
    SELECT MAX("PAYLOAD_MASS__KG_")
    FROM SPACEXTABLE
);
```

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

Out[24]:

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 returns a count of each mission outcome.

SpaceX appears to achieve its mission outcome nearly 99% of the time.

This means that most of the landing failures are intended.

Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

2015 Launch Records

```
%%sql
SELECT MONTHNAME(DATE) AS MONTH, landing__outcome, booster_version, PAYLOAD_MASS__KG_, launch_site
FROM SPACEXDATASET
WHERE landing__outcome = 'Failure (drone ship)' AND YEAR(DATE) = 2015;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.app
Done.
```

MONTH	landing__outcome	booster_version	payload_mass__kg_	launch_site
January	Failure (drone ship)	F9 v1.1 B1012	2395	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	1898	CCAFS LC-40

This query returns the Month, Landing Outcome, Booster Version, Payload Mass (kg), and Launch site of 2015 launches where stage 1 failed to land on a drone ship.

There were two such occurrences.

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

In [30]:

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

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

Out[30]:

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

This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.

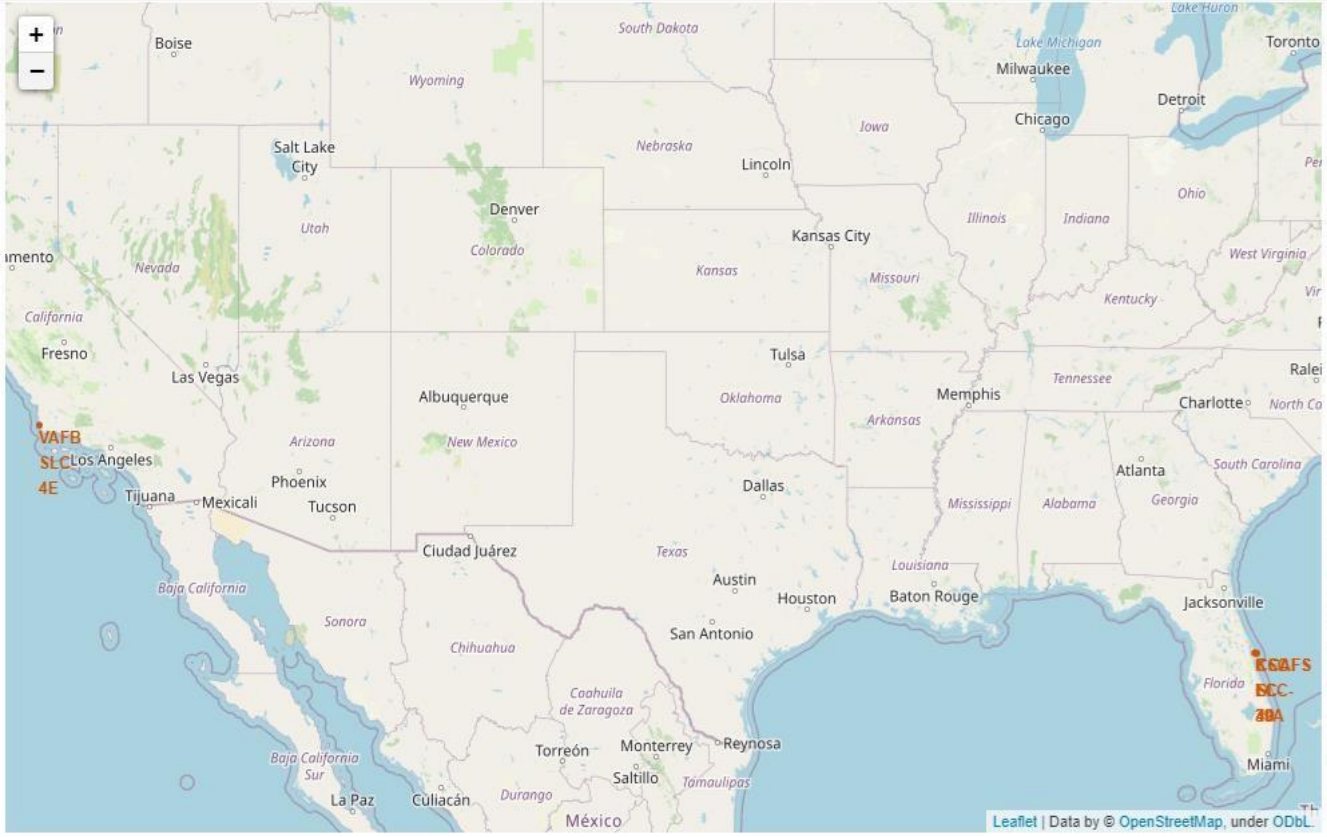
There are two types of successful landing outcomes: drone ship and ground pad landings.

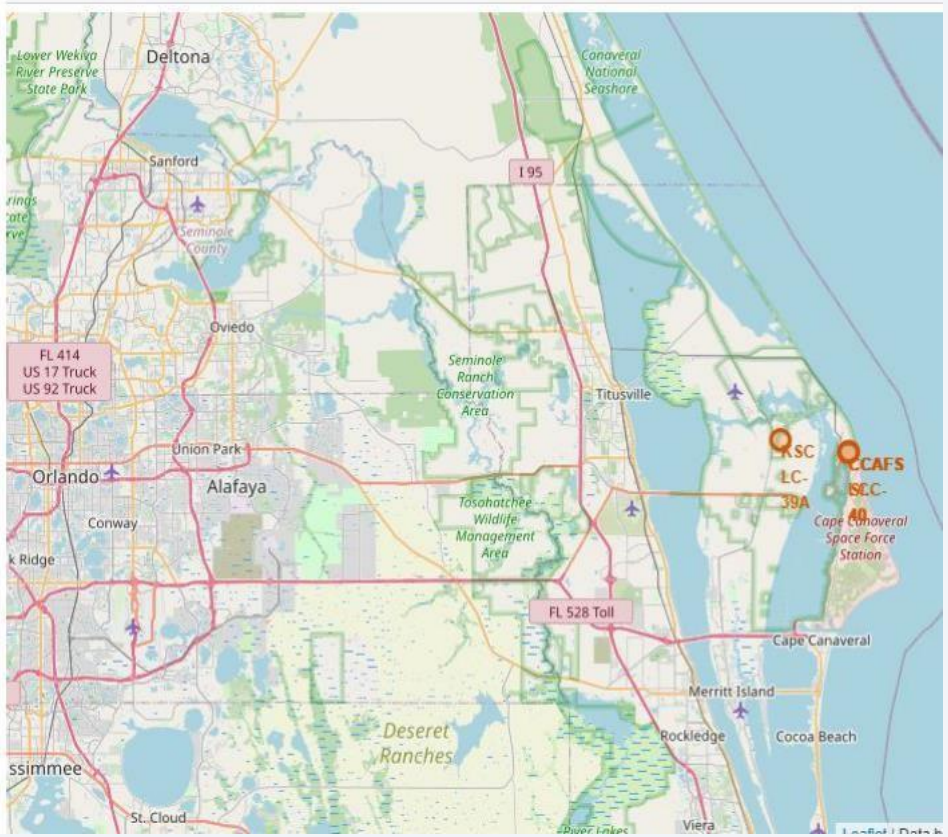
There were 8 successful landings in total during this time period

Section 3

Launch Sites Proximities Analysis

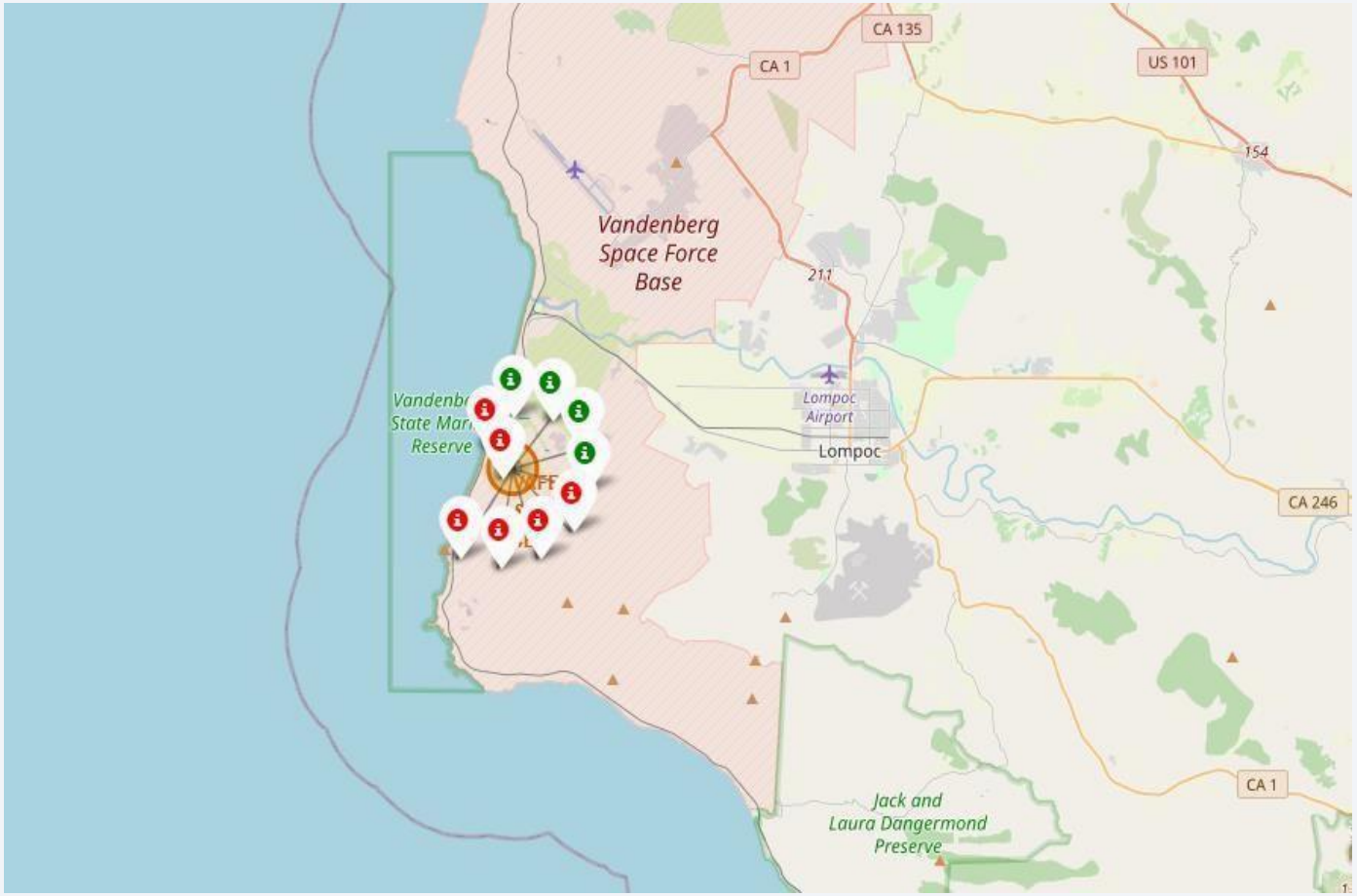
Launch site locations





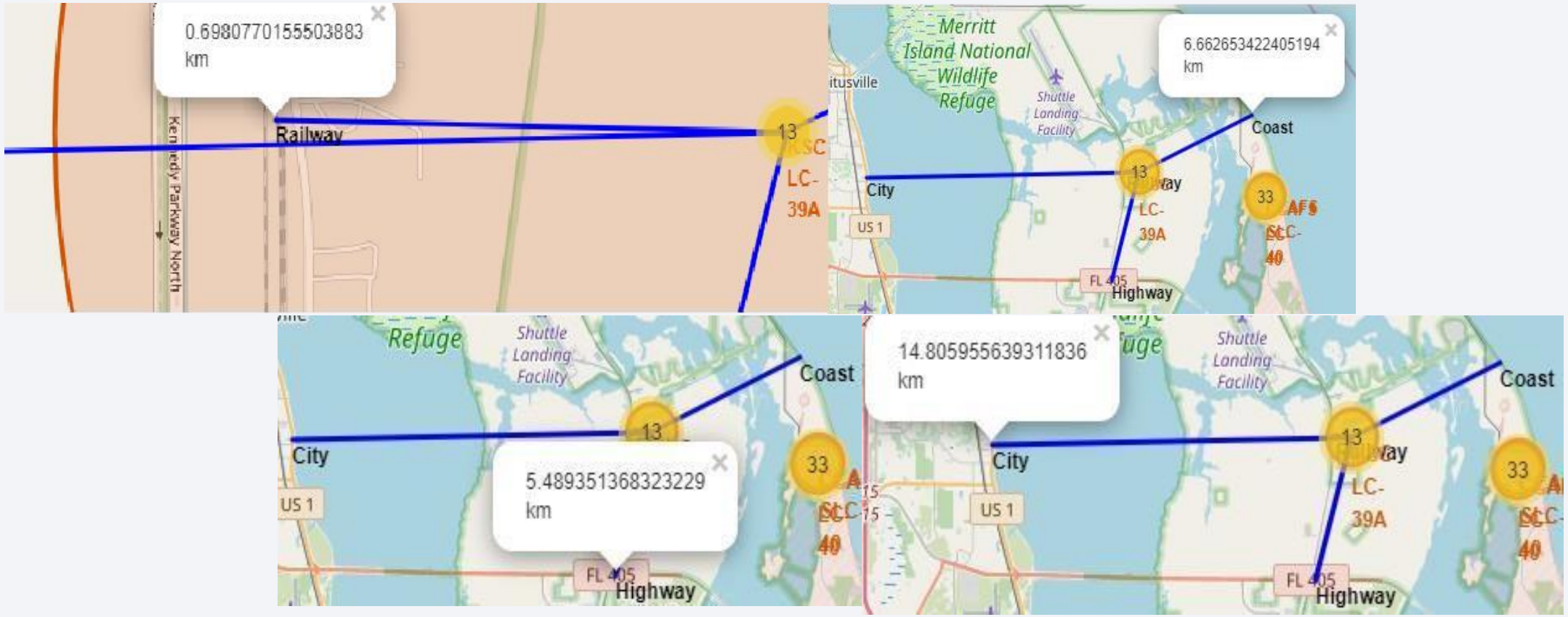
The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other. All launch sites are near the ocean.

Color coded Markers

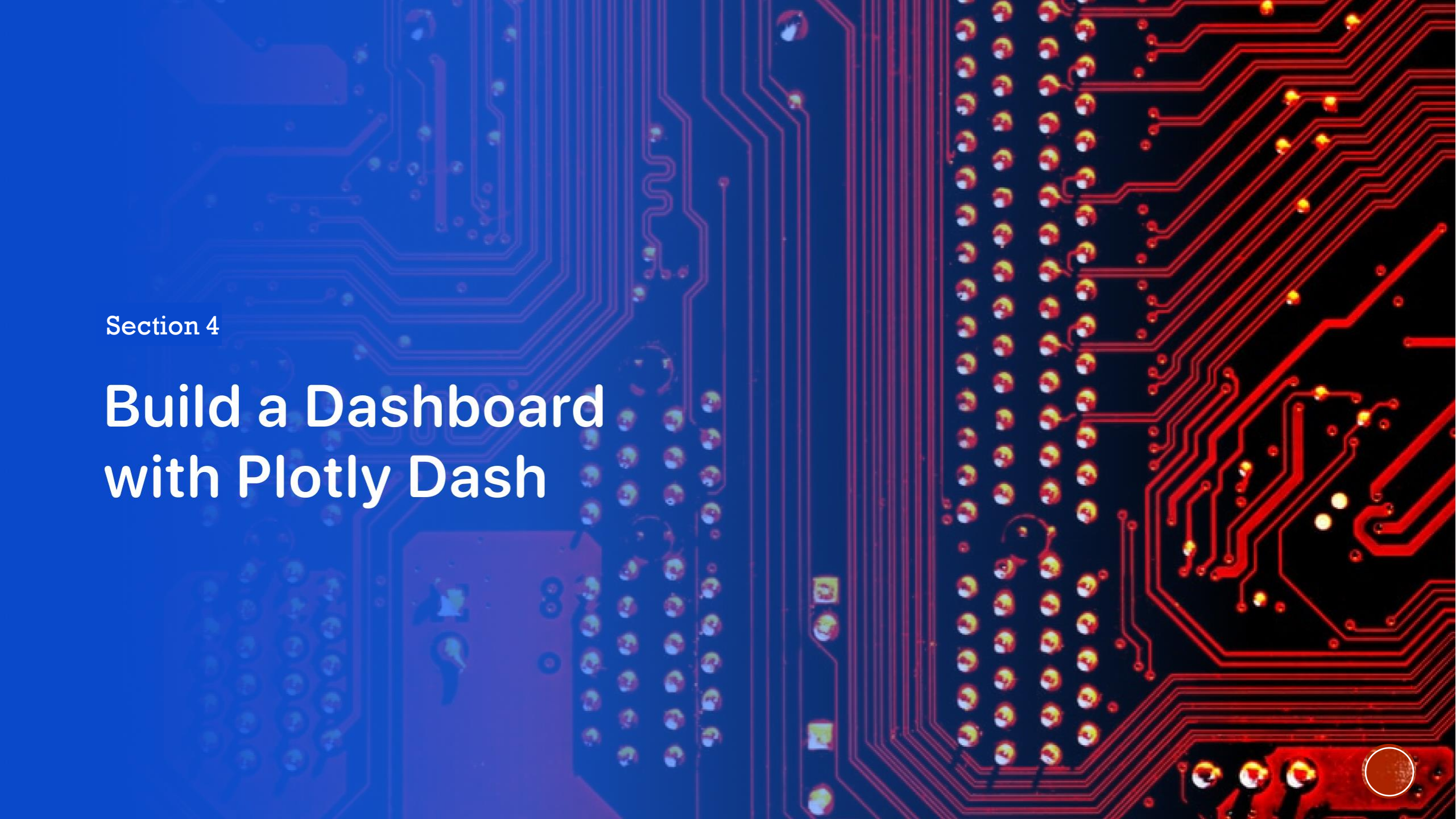


Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). In this example VAFB SLC-4E shows 4 successful landings and 6 failed landings.

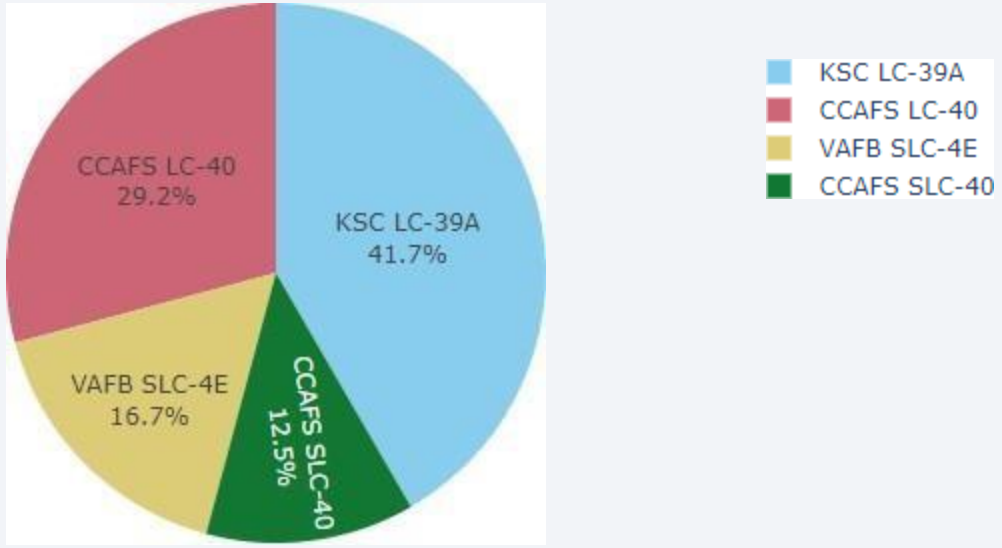
Marker Proximity



Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.


Section 4
Build a Dashboard with Plotly Dash

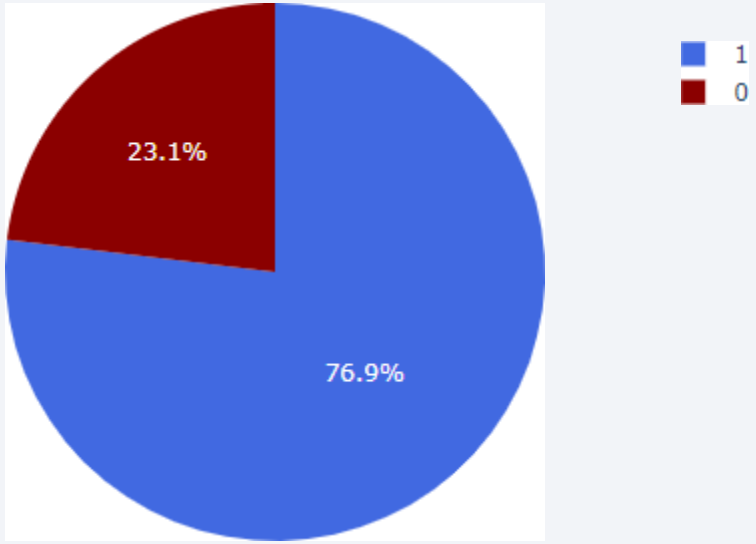
Successful Launches Across Launch Sites



This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.

Highest Success Rate Launch Site

KSC LC-39A Success Rate (blue=success)



KSC LC-39A has the highest success rate with 10 successful landings and 3 failed landings.

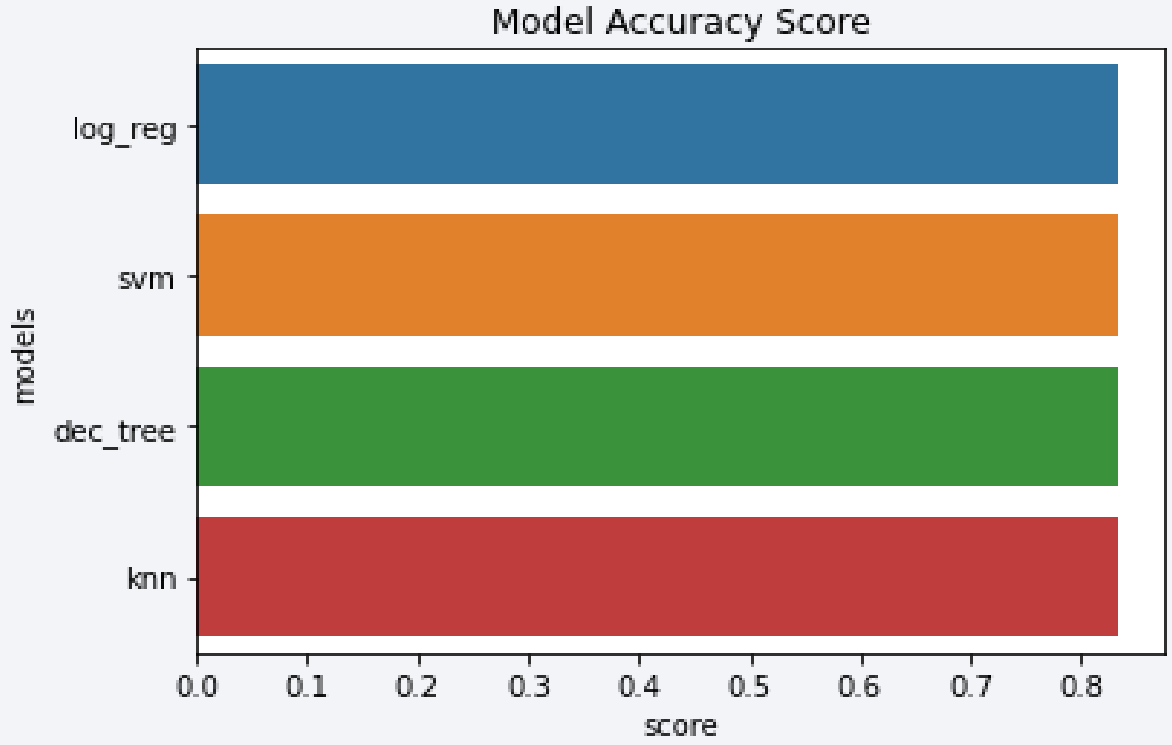
Payload Mass vs. Success vs. Booster Version Category

Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

Section 5

Predictive Analysis (Classification)

Classification Accuracy



All models had virtually the same accuracy on the test set at 83.33% accuracy.

It should be noted that test size is small at only sample size of 18.

This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.

We likely need more data to determine the best model.

Confusion Matrix

Since all models performed the same for the test set, the confusion matrix is the same across all models. The models predicted 12 successful landings when the true label was successful landing.

The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.
The models predicted 3 successful landings when the true label was unsuccessful landings (false positives). Our models over predict successful landings.

Conclusions

All four machine learning models achieved a consistent accuracy rate of approximately 83.33%.

Notably, these models tended to overpredict successful landings, suggesting a need for additional data to enhance model accuracy.

Our analysis yielded valuable insights into SpaceX mission outcomes and the factors influencing rocket landings.

In summary, our exploration and analysis of SpaceX mission data uncovered crucial trends and relationships contributing to mission success.

Our task: Develop a machine learning model for Space Y to compete with SpaceX.

The model's objective: Predict the successful landing of Stage 1, potentially saving ~$100 million USD.

SpaceY, under the leadership of Allon Mask, can use this model to predict, with relatively high accuracy, the success of a Stage 1 landing before launch, aiding in launch decision-making.

For improved model accuracy, we recommend collecting more data to refine the machine learning model selection.

Appendix

- Github repository
  URL: https://github.com/Nicopalameji/SpaceX_Launch-ML-Models/tree/main

- Special Thanks to All Instructors:
- https://www.coursera.org/professional-certificates/ibm-data-science?#instructors

Thank you!