IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ryan Ezekiel WenLongJie Zhang

25 AUG 2025

Outline

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

Executive Summary

- Summary of methodologies
  1. Data collection through API requests and web scraping
  2. Exploratory data analysis with Python visualizations and SQL queries
  3. Interactive visualization with Folium
  4. Dashboard using Plotly Dash
  5. Predictive analysis using machine learning (scikit-learn)
- Summary of all results
  1. Success has improved over time
  2. Logistic regression and a decision tree classifier were found to be equally as effective at predicting launch outcomes

Introduction

SpaceX has been a major disruptor in the aerospace industry. One of their innovations has been reusing their first stage boosters, which are usually discarded after launch. If their rockets land successfully, it’s not just a successful mission, it means money saved. Using data from both SpaceX’s REST API and publicly scraped Wikipedia data, machine learning models were built and tested to see if launch outcomes could be predicted accurately.

- Questions:

1. What factors effect a successful launch?
2. What machine learning model most accurately predicts launch outcomes?
3. How does landing success change over time?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API
  - Werb Scraping Wikipedia
- Perform data wrangling
  - Launch types were transformed into a class column indicating success/failure
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Evaluated the accuracy of multiple cross-validated classification models

Data Collection

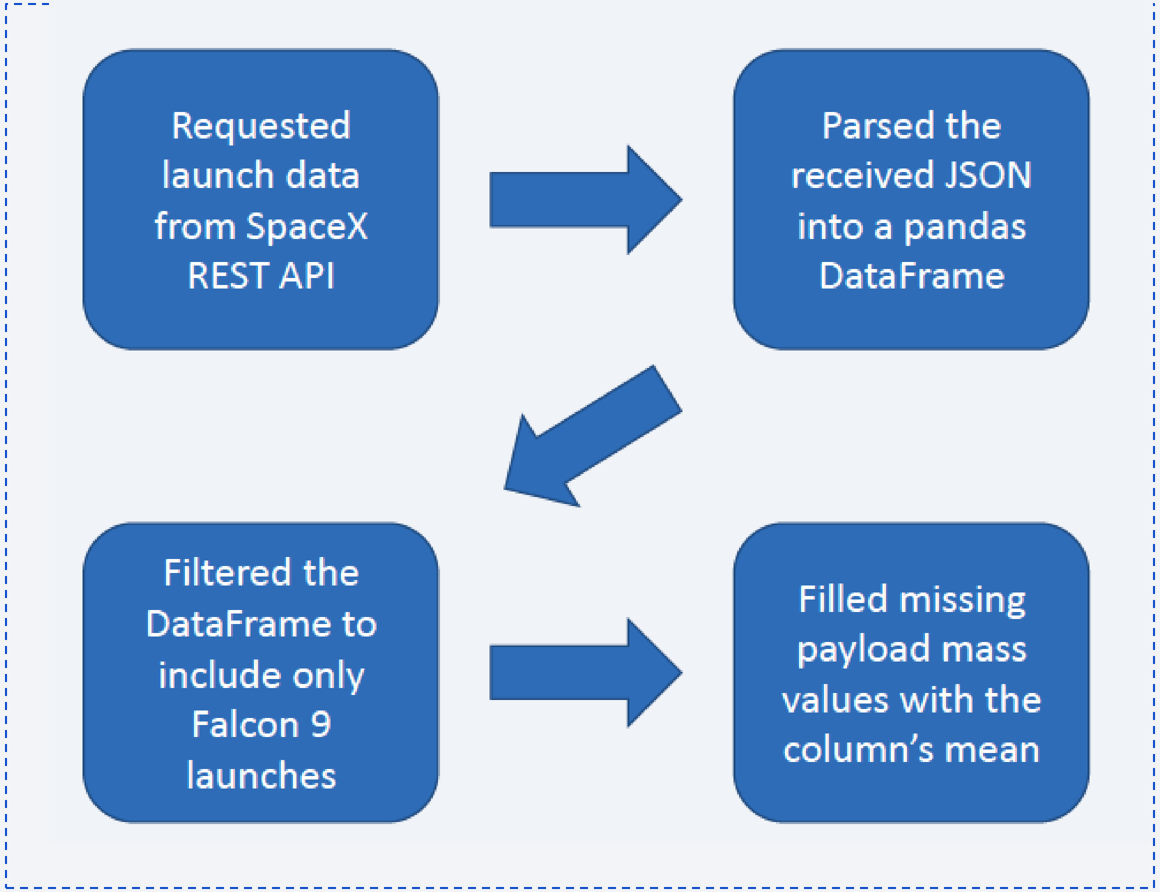
Data was collected from SpaceX’s REST API and scraping Wikipedia for additional launch data. This was done to obtain additional data not available in either source alone.

Some of these features are:

- -Payload mass
- -Whether grid fins were present or not
- -Number of reuses
- -Longitude and latitude

Data Collection – SpaceX API

- Data was requested from the SpaceX REST API
- The JSON response was parsed into a pandas DataFrame
- The DataFrame was filtered for Falcon 9 data only
- The payload mass column’s missing values were imputed with the column’s mean
- The data was saved for later use



Requested launch data from SpaceX REST API

Parsed the received JSON into a pandas DataFrame

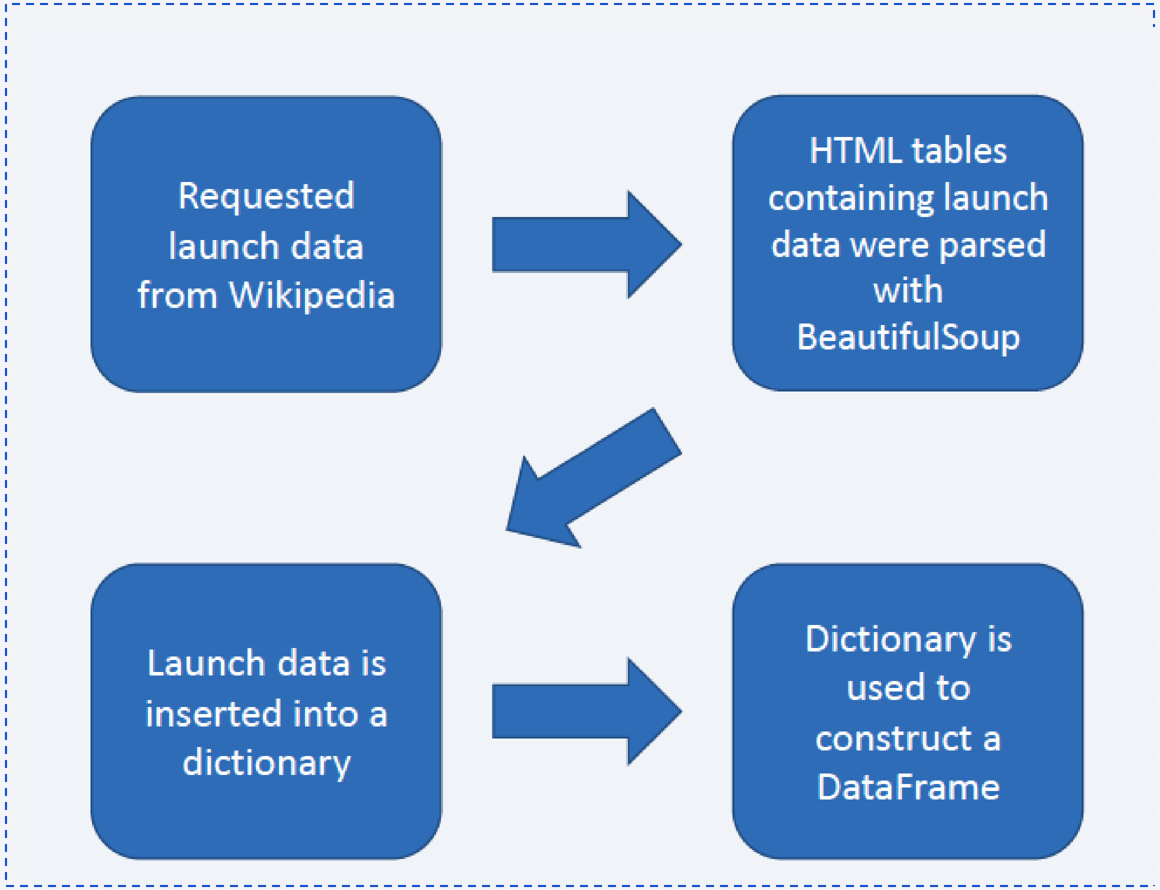
Filtered the DataFrame to include only Falcon 9 launches

Filled missing payload mass values with the column’s mean

SpaceX API calls notebook: https://github.com/ryanwljzhang/ibmdatascience/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- An HTTP request was sent to retrieve the content of a Wikipedia page containing SpaceX launch data
- Using BeautifulSoup, the HTML object was parsed
- Table headers and values were inserted into a dictionary
- The dictionary was used to construct a pandas DataFrame



Requested launch data from Wikipedia

HTML tables containing launch data were parsed with BeautifulSoup

Launch data is inserted into a dictionary

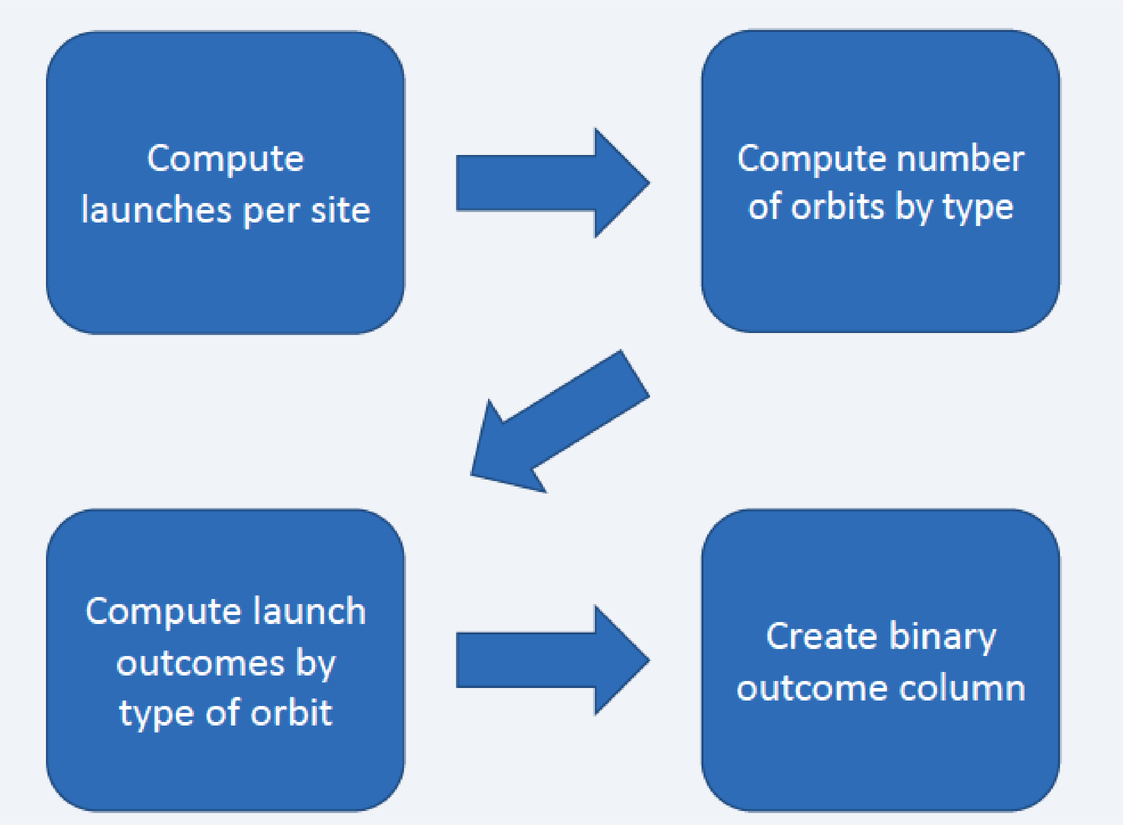
Dictionary is used to construct a DataFrame

Web scraping notebook: https://github.com/ryanwljzhang/ibmdatascience/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

Multiple labels were present for describing launches, which were simplified into a binary column of successes/failures

Data wrangling notebook: https://github.com/ryanwljzhang/ibmdatascience/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



Compute launches per site

Compute number of orbits by type

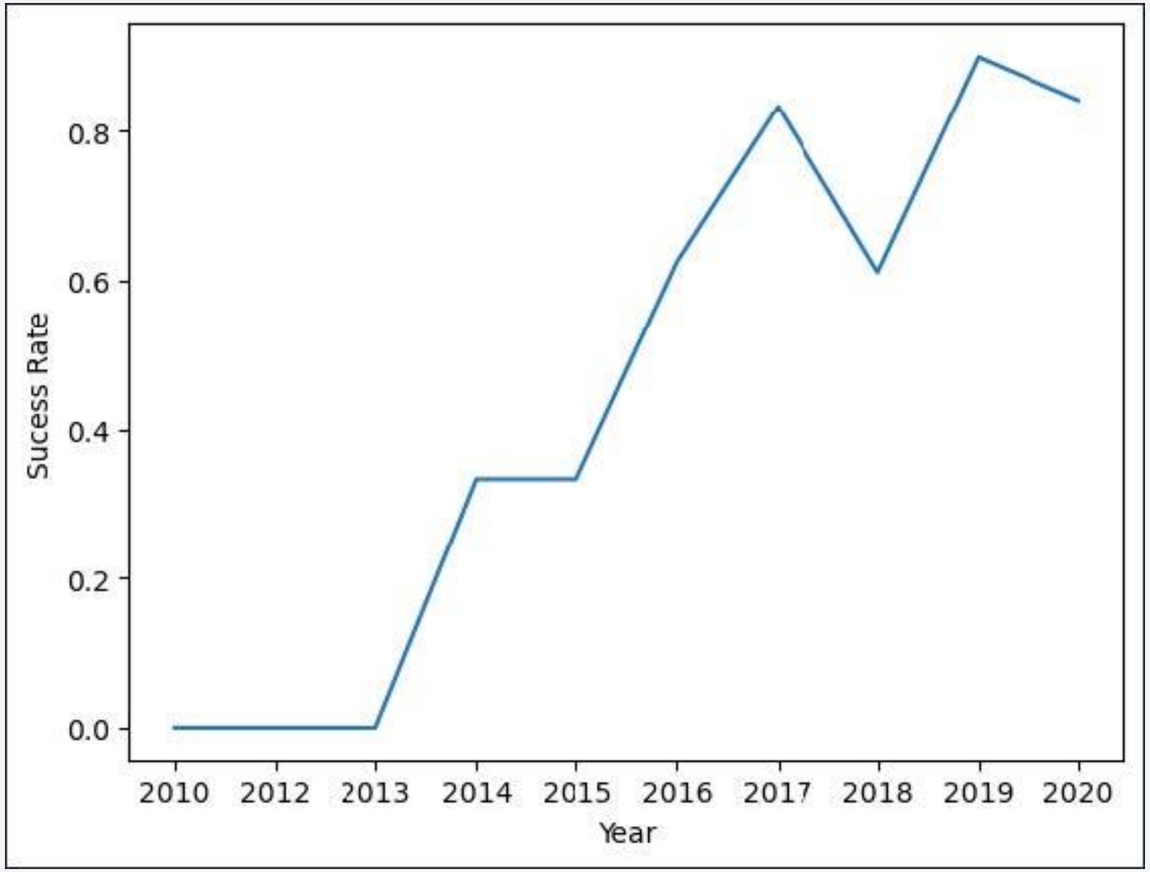
Compute launch outcomes by type of orbit

Create binary outcome column

EDA with Data Visualization

- Plotted various graphs:
- Scatterplots: payload mass, launch site, orbit, payload mass vs. launch site, payload mass vs. orbit
- Bar graph: orbit vs. success rate
- Line graph: year vs. success rate

EDA data visualization: https://github.com/ryanwljzhang/ibmdatascience/blob/main/edadataviz.ipynb



EDA with SQL

- Queried unique launch site names
- Displayed 5 records where the launch site's name began with 'CCA'
- Summed the total mass of all payloads
- Calculated the average mass carried by F9 v1.1 boosters
- Displayed the date of the first successful ground pad landing
- Displayed names of boosters that have succeeded in drone ship with payload mass between 4000 and 6000 kg
- Listed total number of successful and failed mission outcomes
- Listed all booster versions that have carried the maximum payload mass
- Listed the month, landing outcome, booster version, and launch site for drone ship failures in 2015
- Ranked landing outcomes between 6/4/2010 and 3/20/2017 in descending order
- SQL notebook: https://github.com/ryanwljzhang/ibmdatascience/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Placed circles with labels to highlight the various launch sites
- laced a marker cluster containing markers for each launch (green for success red for fail) to visually demonstrate the success rate of each launch site
- Lines going from Cape Canaveral Space Launch Complex 40 to the nearest coastline, railroad, highway, and city with markers for their distance to highlight landmark distance trends across launch sites

Map HTML: https://github.com/ryanwljzhang/ibmdatascience/blob/main/map.html

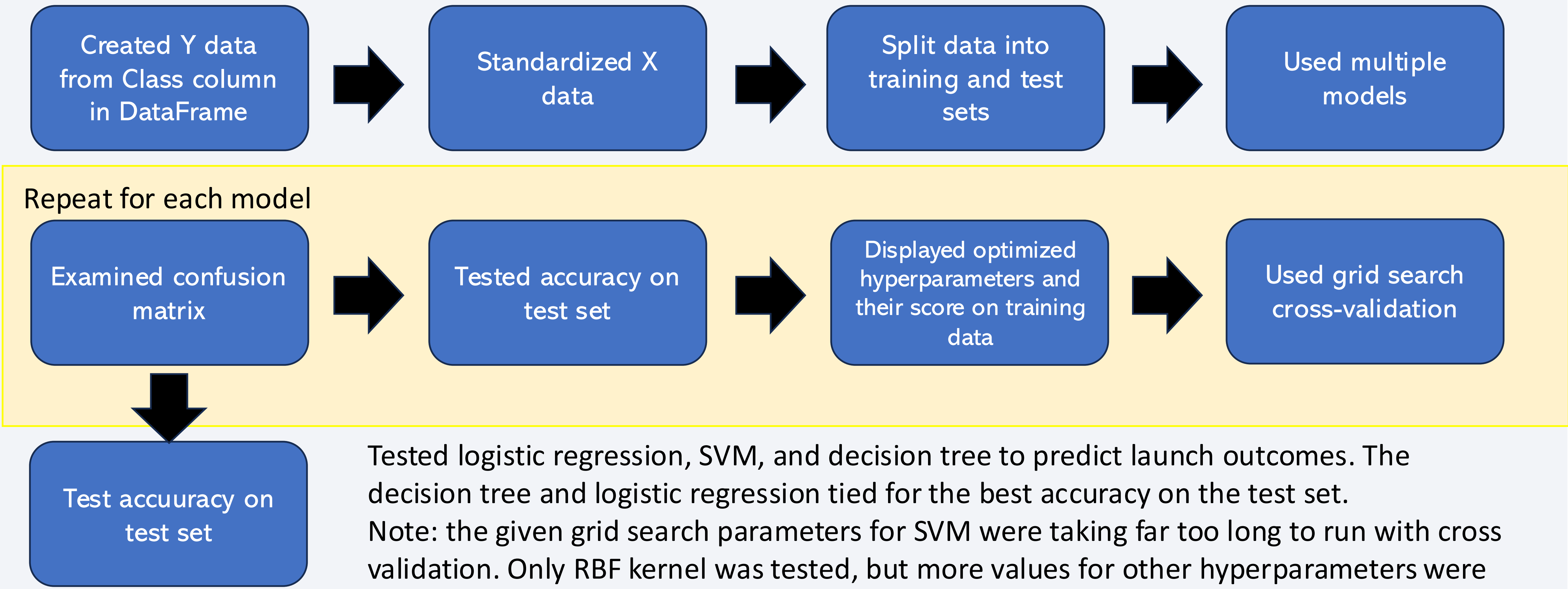
Folium notebook: https://github.com/ryanwljzhang/ibmdatascience/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Dropdown menu to select data for all sites or a specific one
- Pie chart
  - For all launch sites: displays portion of total successes that occurred at each site
  - For single sites: displays successes and failures for that site
- Scatterplot graphing payload mass against success for selected sites, with points colored based on booster version
- Slider to change range of payload masses displayed in the scatterplot

Dash lab: https://github.com/ryanwljzhang/ibmdatascience/blob/main/spacex-dash-app.py

Predictive Analysis (Classification)



Created Y data from Class column in DataFrame

Standardized X data

Split data into training and test sets

Used multiple models

Repeat for each model

Used grid search cross-validation

Displayed optimized hyperparameters and their score on training data

Tested accuracy on test set

Examined confusion matrix

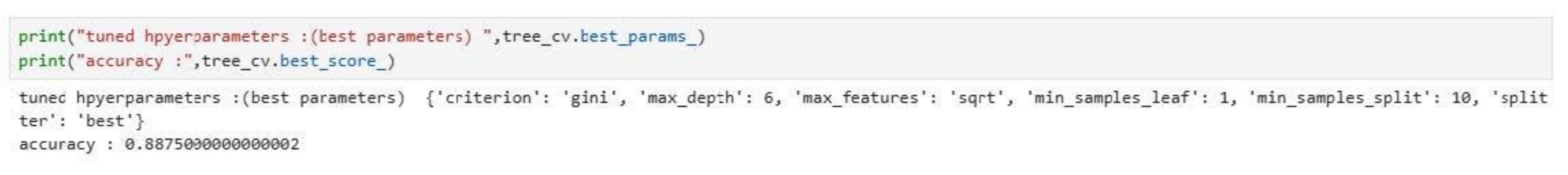
Test accuuracy on test set

Tested logistic regression, SVM, and decision tree to predict launch outcomes. The decision tree and logistic regression tied for the best accuracy on the test set.

Note: the given grid search parameters for SVM were taking far too long to run with cross validation. Only RBF kernel was tested, but more values for other hyperparameters were tested to try to compensate for this.

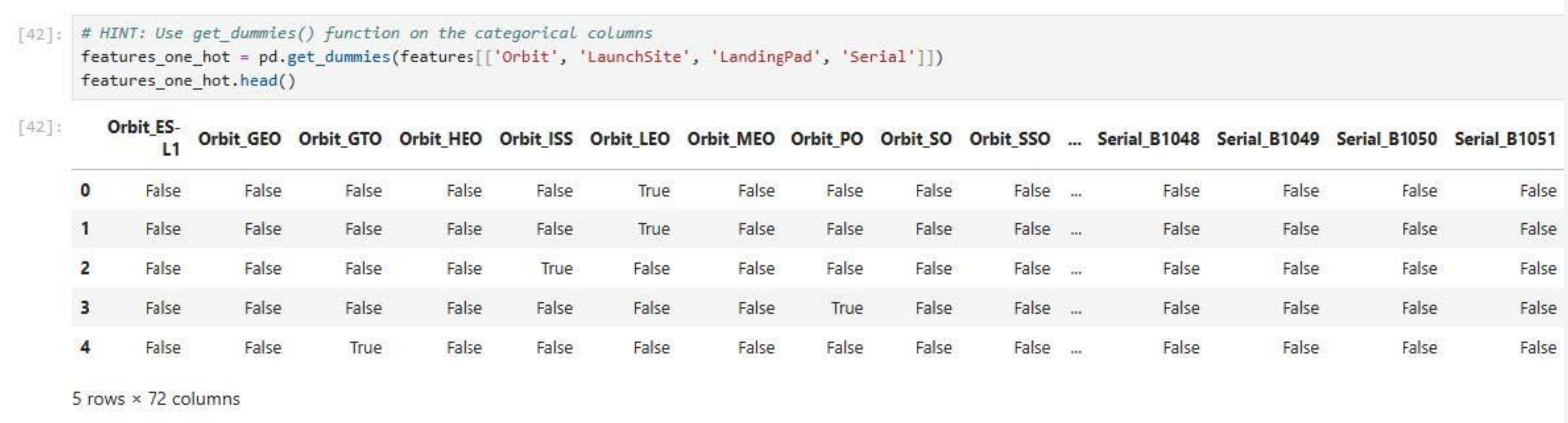
Machine learning notebook:
https://github.com/ryanwljzhang/ibmdatascience/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results



```
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 10, 'split
ter': 'best'}
accuracy : 0.8875000000000002
```



```
[42]: # HINT: Use get_dummies() function on the categorical columns
      features_one_hot = pd.get_dummies(features[['Orbit', 'LaunchSite', 'LandingPad', 'Serial']])
      features_one_hot.head()
```

[42]:

	Orbit_ES-L1	Orbit_GEO	Orbit_GTO	Orbit_HEO	Orbit_ISS	Orbit_LEO	Orbit_MEO	Orbit_PO	Orbit_SO	Orbit_SSO	...	Serial_B1048	Serial_B1049	Serial_B1050	Serial_B1051
0	False	False	False	False	False	True	False	False	False	False	...	False	False	False	False
1	False	False	False	False	False	True	False	False	False	False	...	False	False	False	False
2	False	False	False	False	True	False	False	False	False	False	...	False	False	False	False
3	False	False	False	False	False	False	False	True	False	False	...	False	False	False	False
4	False	False	True	False	False	False	False	False	False	False	...	False	False	False	False

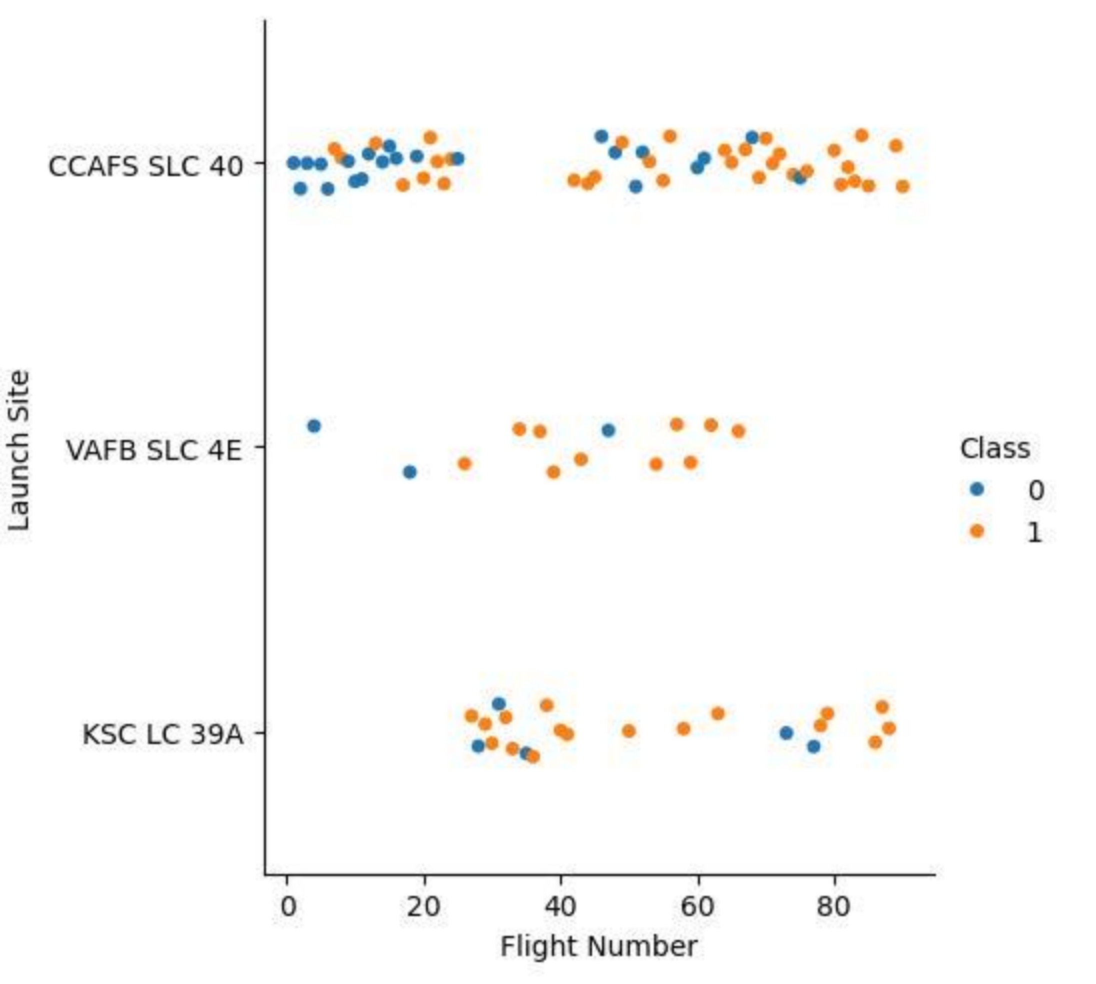
5 rows × 72 columns

Section 2

Insights drawn from EDA

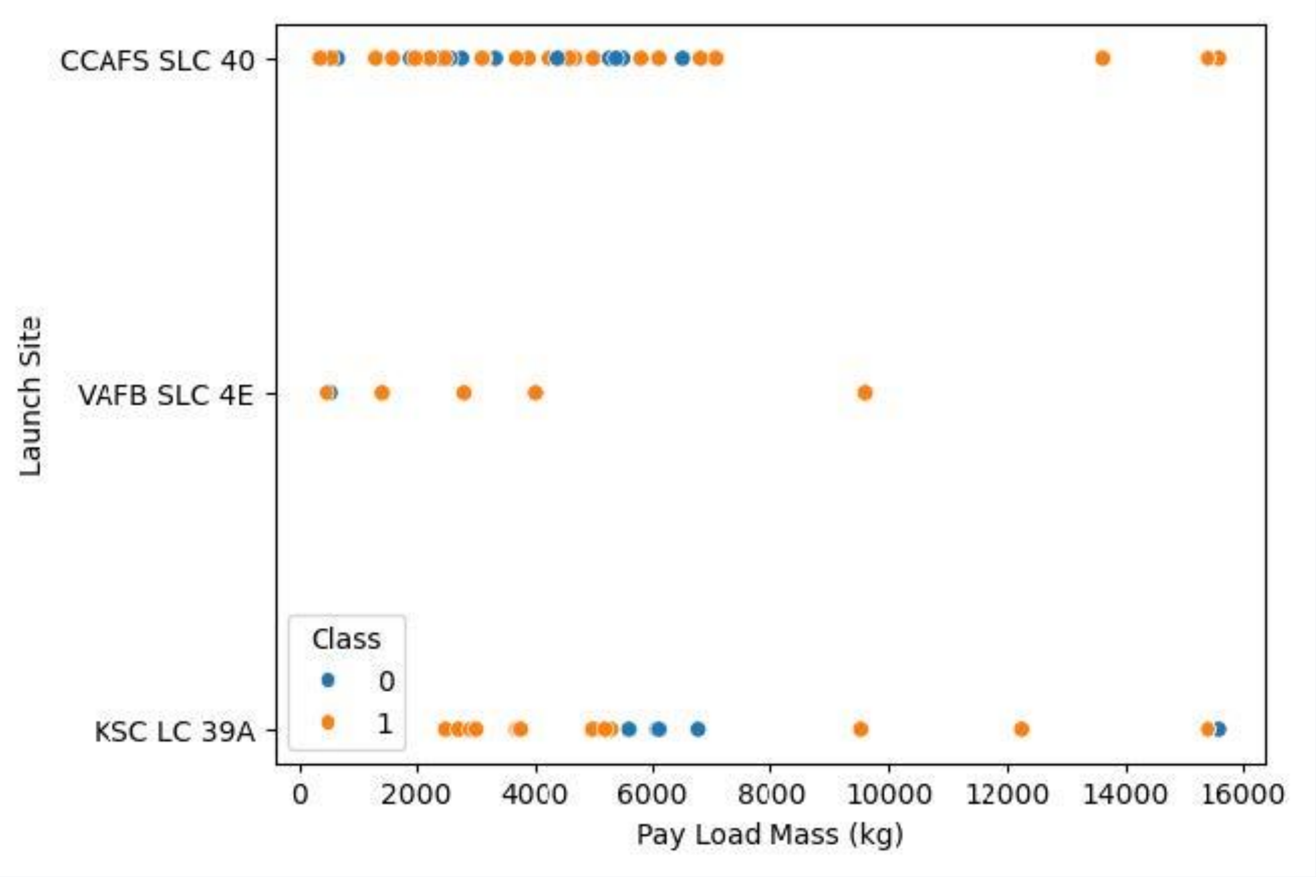
Flight Number vs. Launch Site

- Scatterplot showing how successes changed over time and based on site.



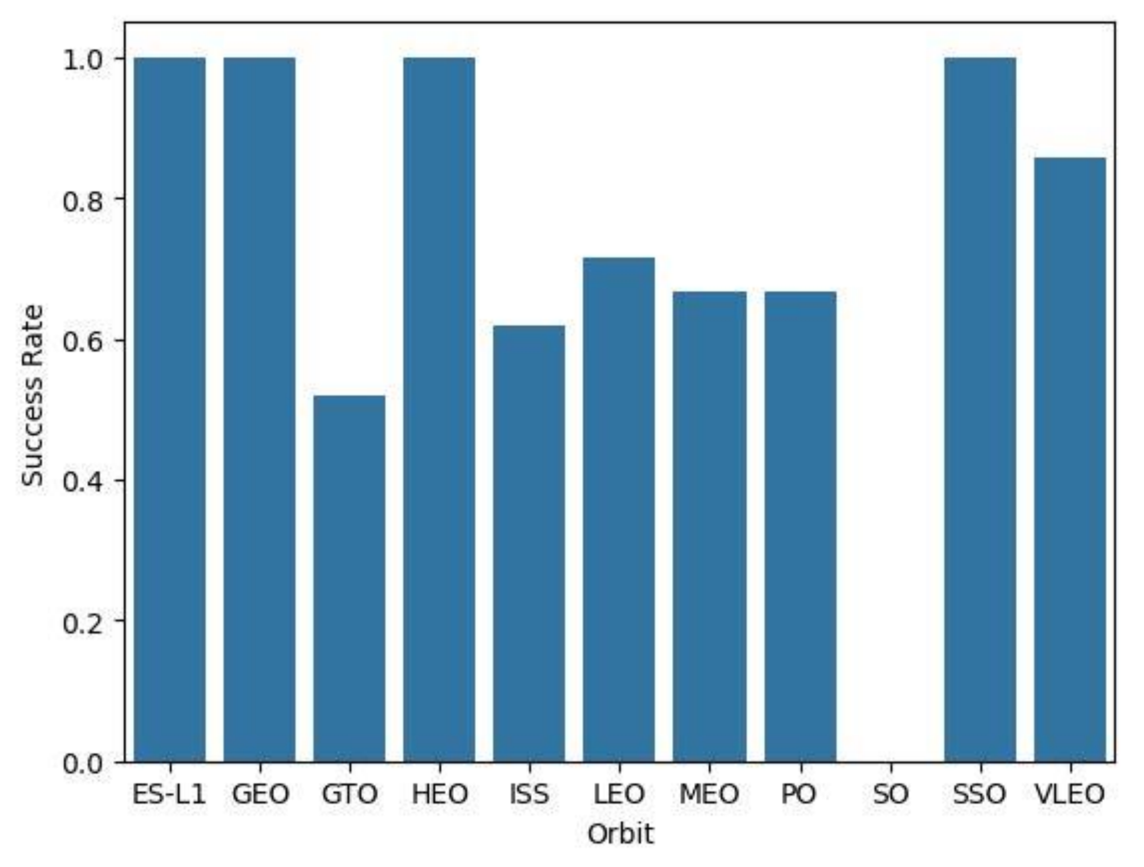
Payload vs. Launch Site

- A scatter plot ofPayload vs. Launch Site.



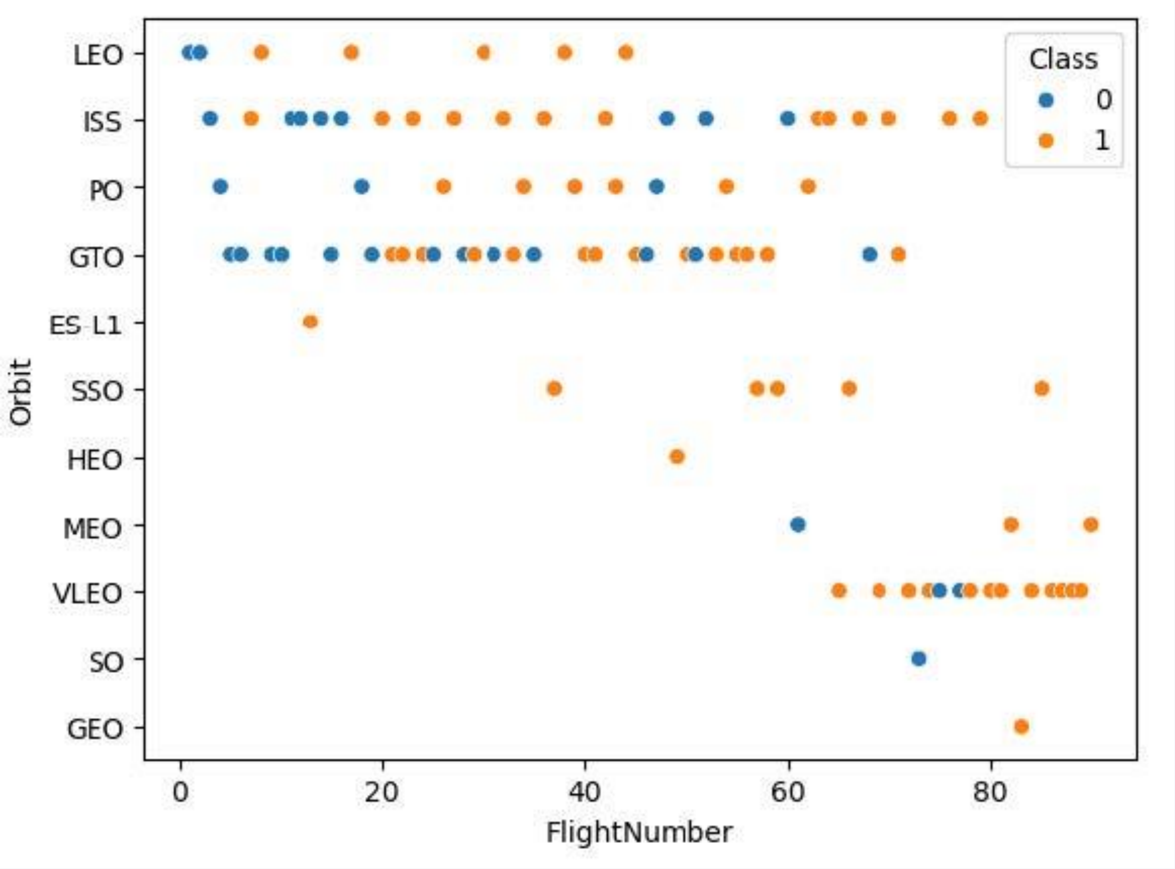
Success Rate vs. Orbit Type

Bar chart for the success rate of each orbit type.



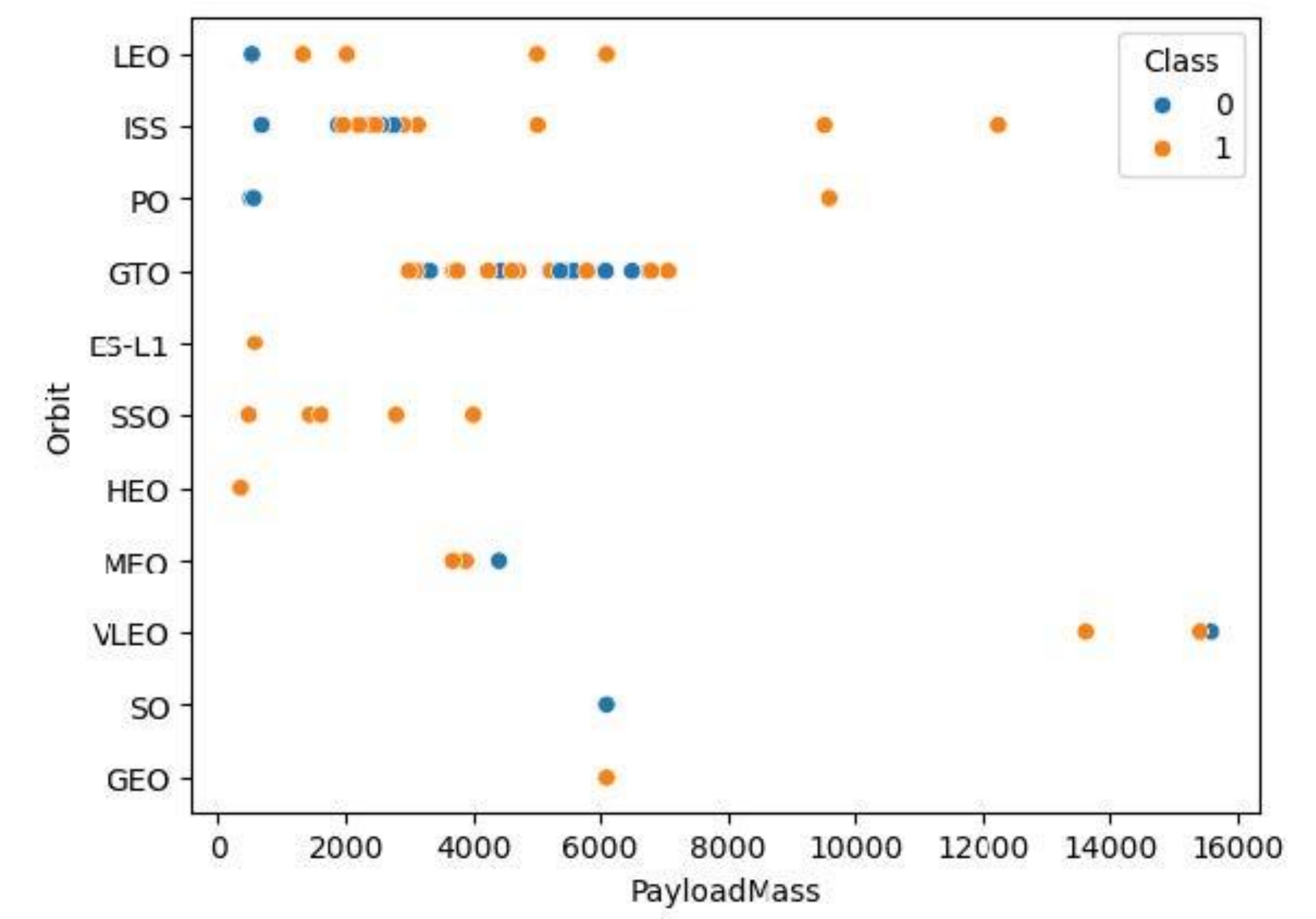
Flight Number vs. Orbit Type

Scatterplot of Flight number vs. Orbit type.



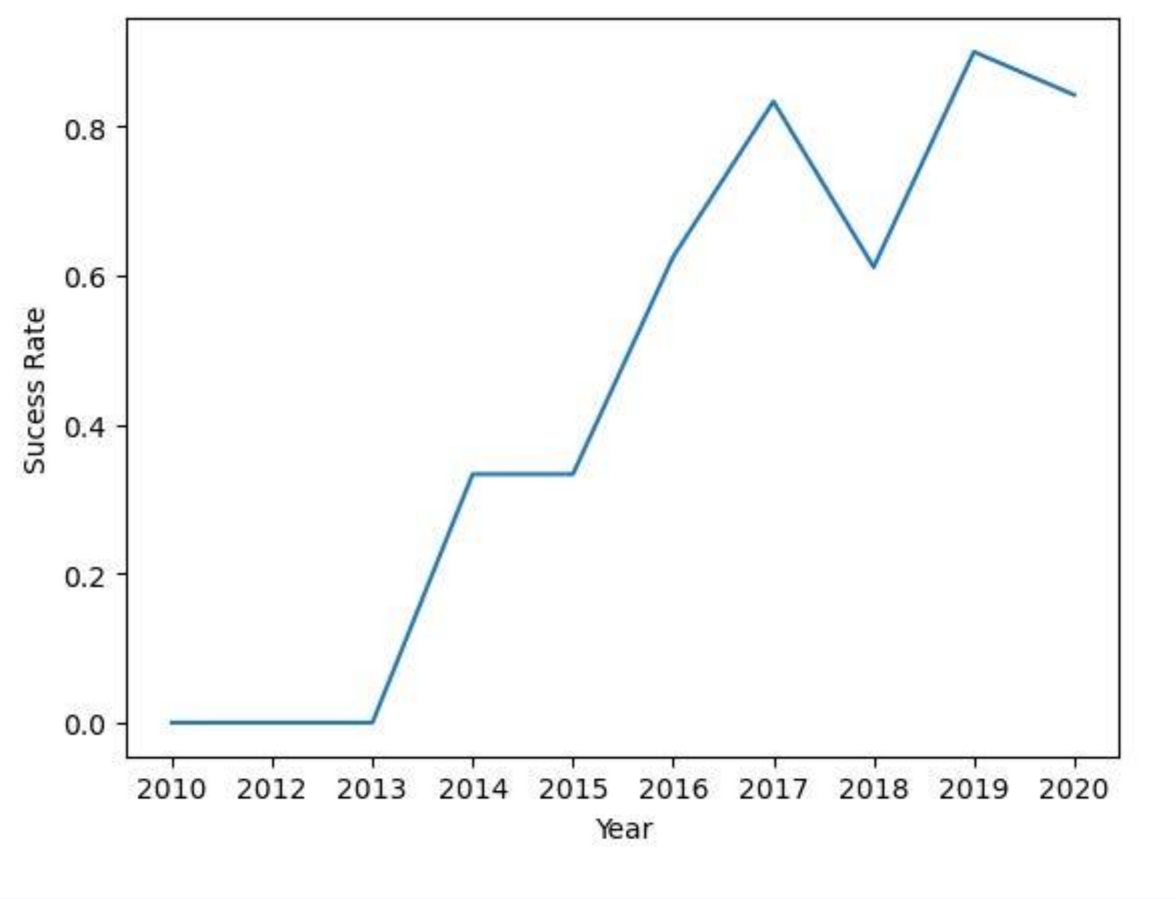
Payload vs. Orbit Type

- Scatterplot of payload vs. orbit type.



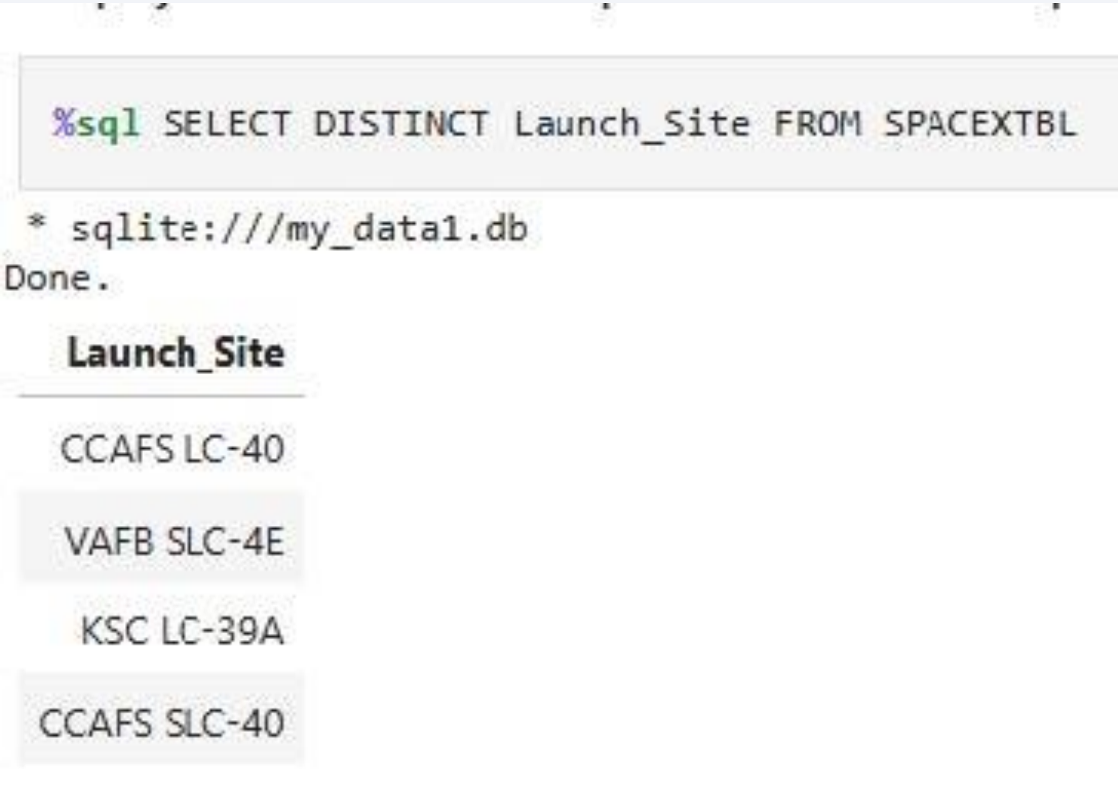
Launch Success Yearly Trend

- Line plot showing success rate vs. year.



All Launch Site Names

- Find the names of the unique launch sites



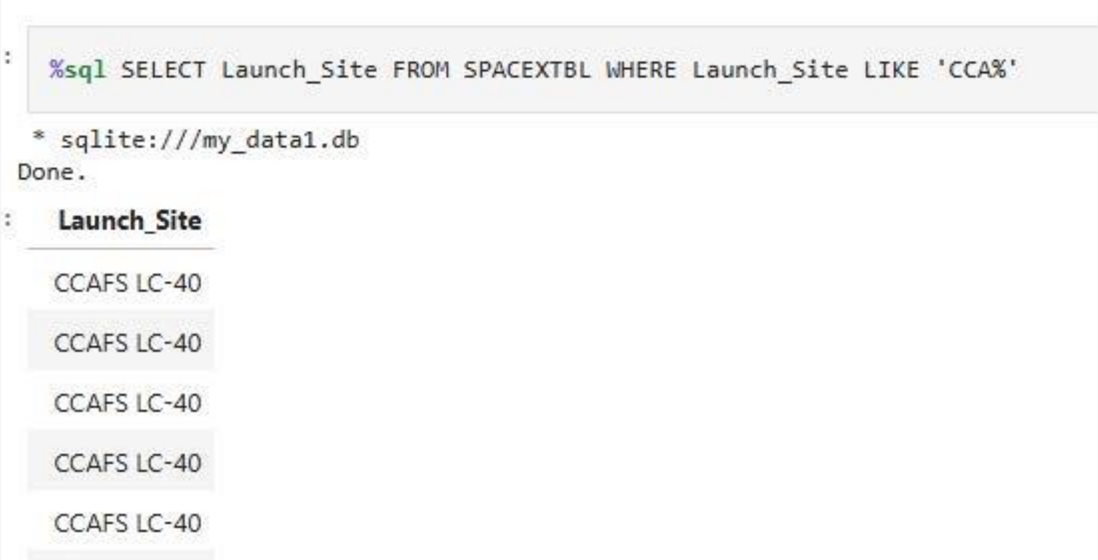
```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

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

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`



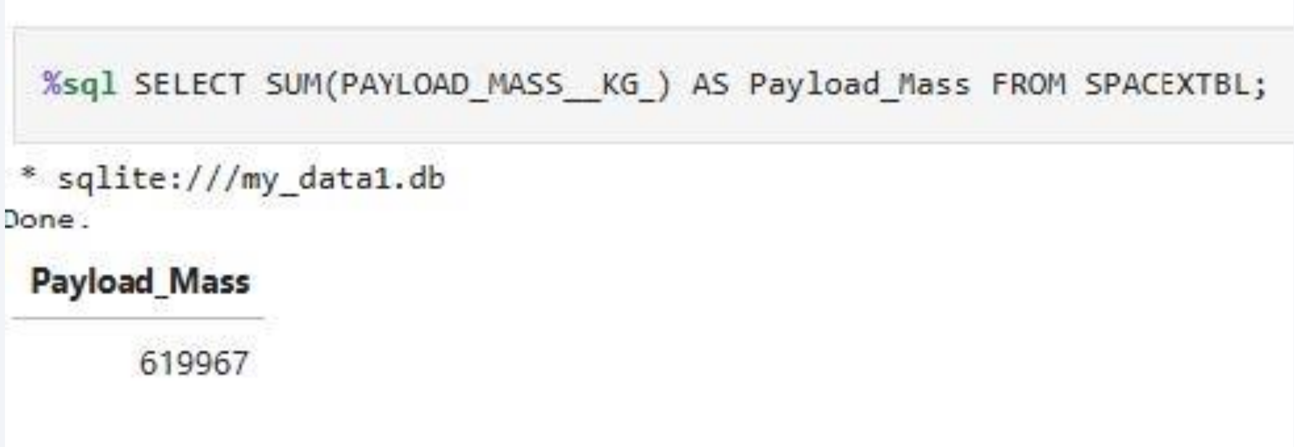
```
%sql SELECT Launch_Site FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%'
```

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

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

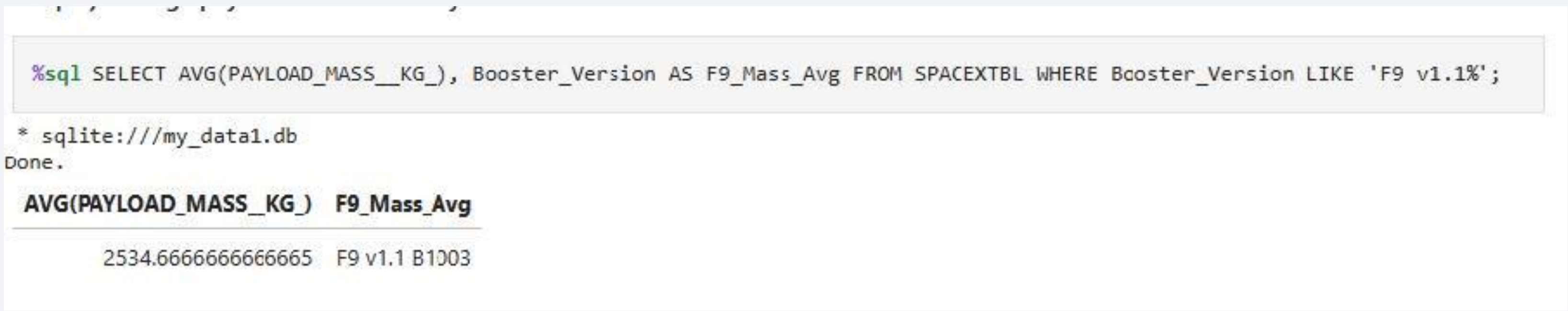


```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS Payload_Mass FROM SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
Payload_Mass
      619967
```

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1



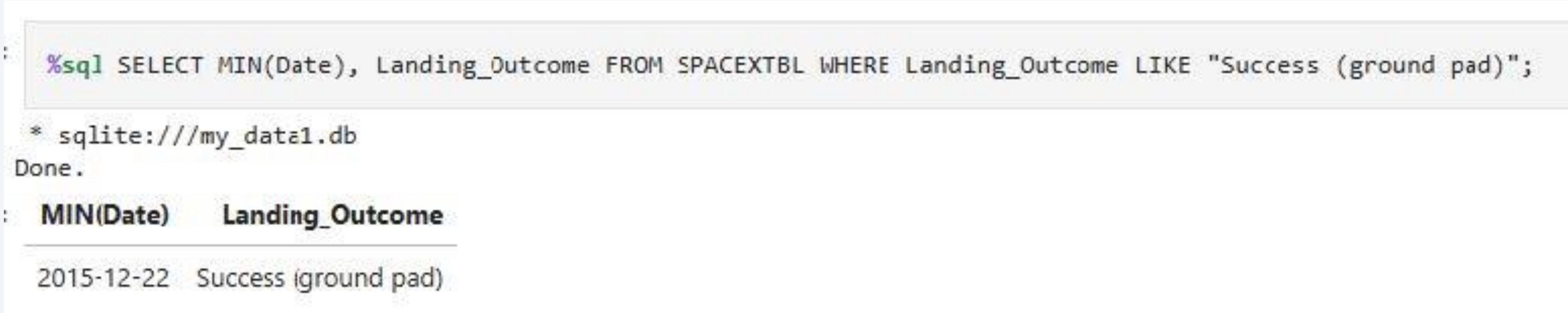
```
%sql SELECT AVG(PAYLOAD_MASS__KG_), Booster_Version AS F9_Mass_Avg FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.1%';
```

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

AVG(PAYLOAD_MASS__KG_)	F9_Mass_Avg
2534.6666666666665	F9 v1.1 B1003

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad



```
%sql SELECT MIN(Date), Landing_Outcome FROM SPACEXTBL WHERE Landing_Outcome LIKE "Success (ground pad)";
```

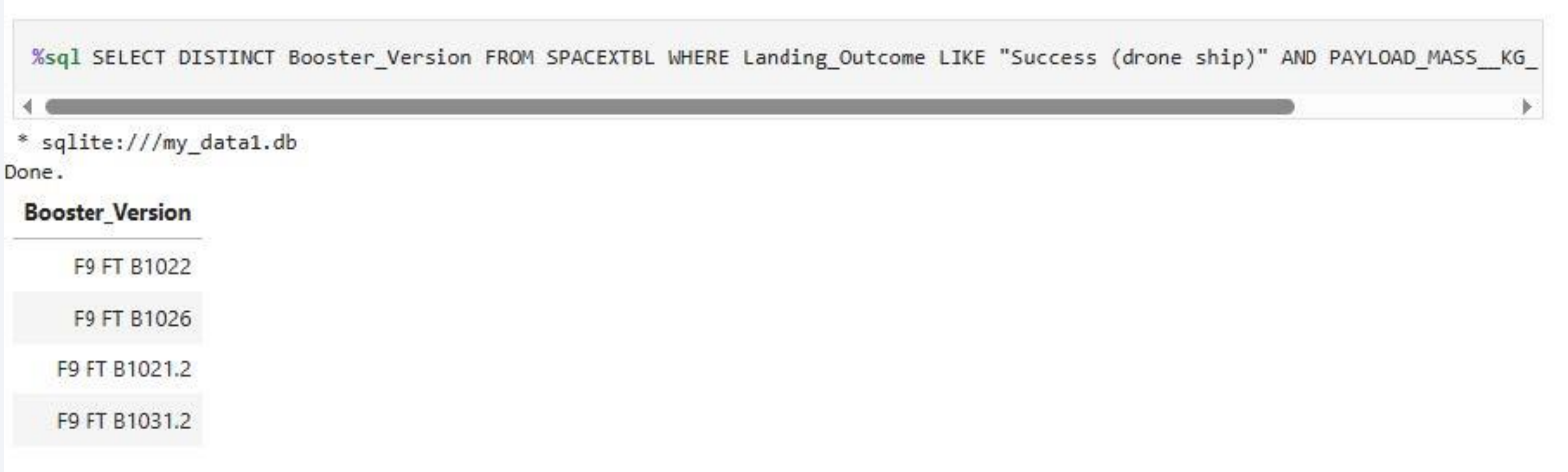
```
 * sqlite:///my_data1.db
Done.
```

MIN(Date)	Landing_Outcome
2015-12-22	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

%sqlSELECT DISTINCT Booster_VersionFROM SPACEXTBL WHERE Landing_OutcomeLIKE "Success (drone ship)" AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000



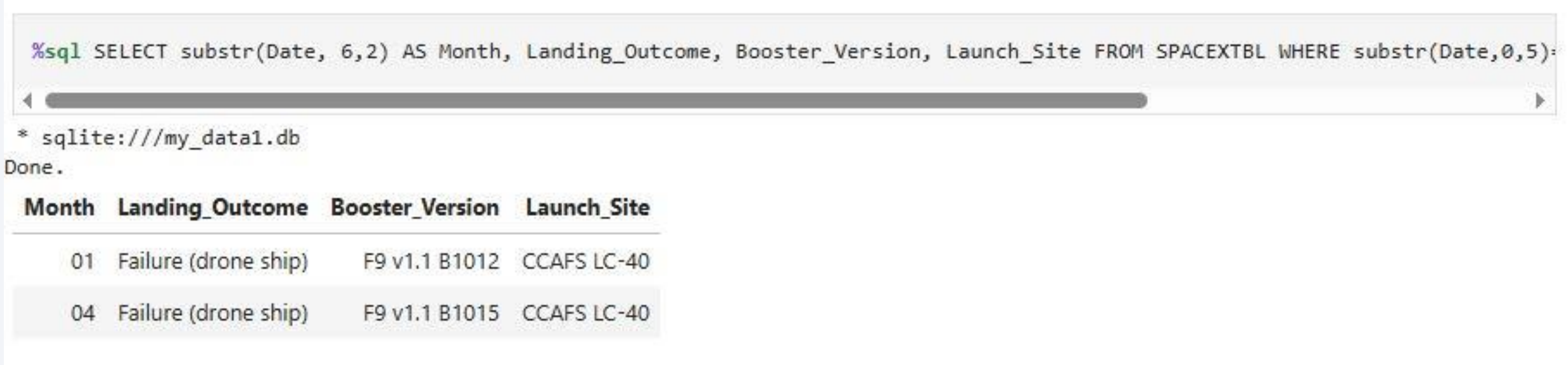
```
%sql SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome LIKE "Success (drone ship)" AND PAYLOAD_MASS__KG_
```

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

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- %sqlSELECT DISTINCT Booster_VersionFROM SPACEXTBL WHERE PAYLOAD_MASS__KG_=(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);



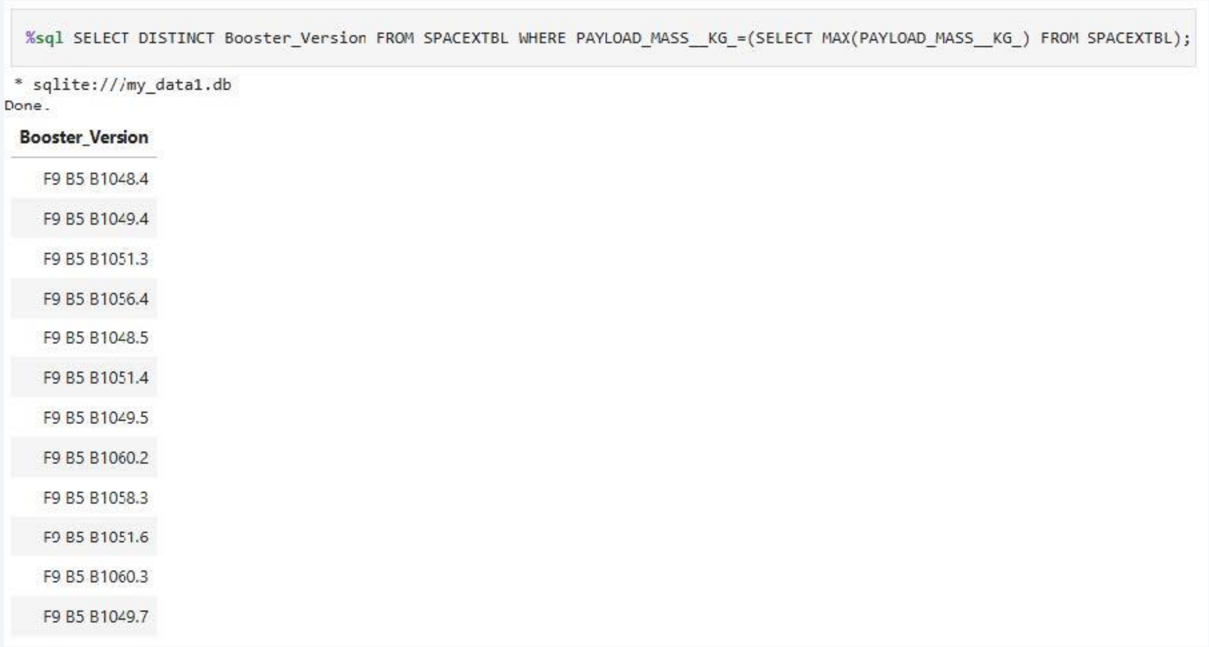
```
%sql SELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL WHERE substr(Date,0,5)=
```

```
 * 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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- %sqlSELECT DISTINCT Booster_VersionFROM SPACEXTBL WHERE PAYLOAD_MASS__KG_=(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);



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

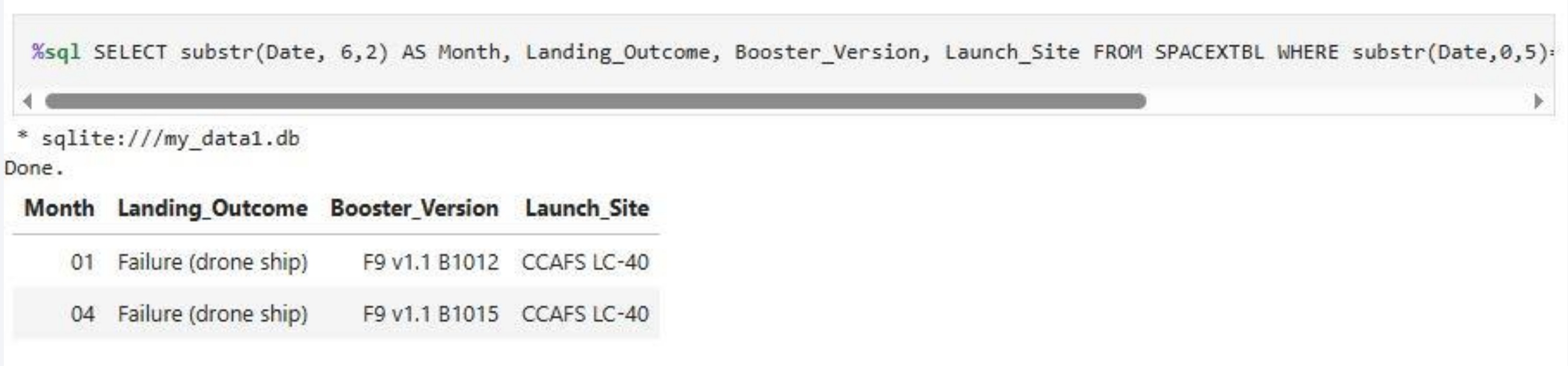
```
 * sqlite:///my_data1.db
Done.
```

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- %sqlSELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_SiteFROM SPACEXTBL WHERE substr(Date,0,5)='2015' AND Landing_OutcomeLIKE 'Failure%'



```
%sql SELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL WHERE substr(Date,0,5)
```

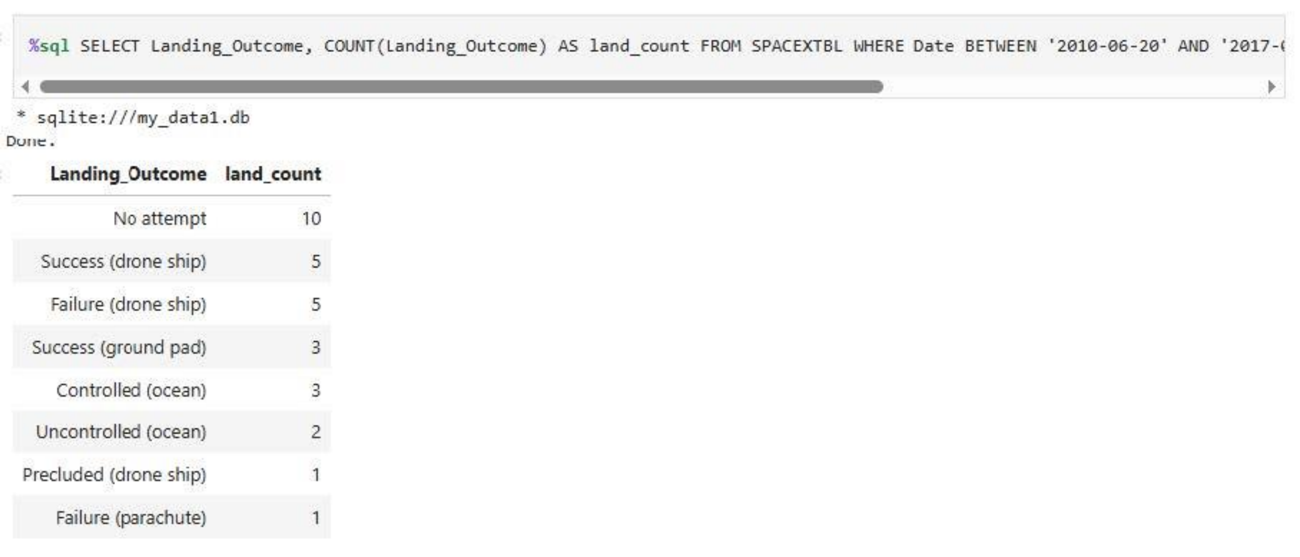
```
* sqlite:///my_data1.db
Done.
```

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 date 2010-06-04 and 2017-03-20, in descending order

- %sqlSELECT Landing_Outcome, COUNT(Landing_Outcome) AS land_countFROM SPACEXTBL WHERE Date BETWEEN '2010-06-20' AND '2017-03-20' GROUP BY Landing_OutcomeORDER BY land_countDESC



```
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS land_count FROM SPACEXTBL WHERE Date BETWEEN '2010-06-20' AND '2017-
```

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

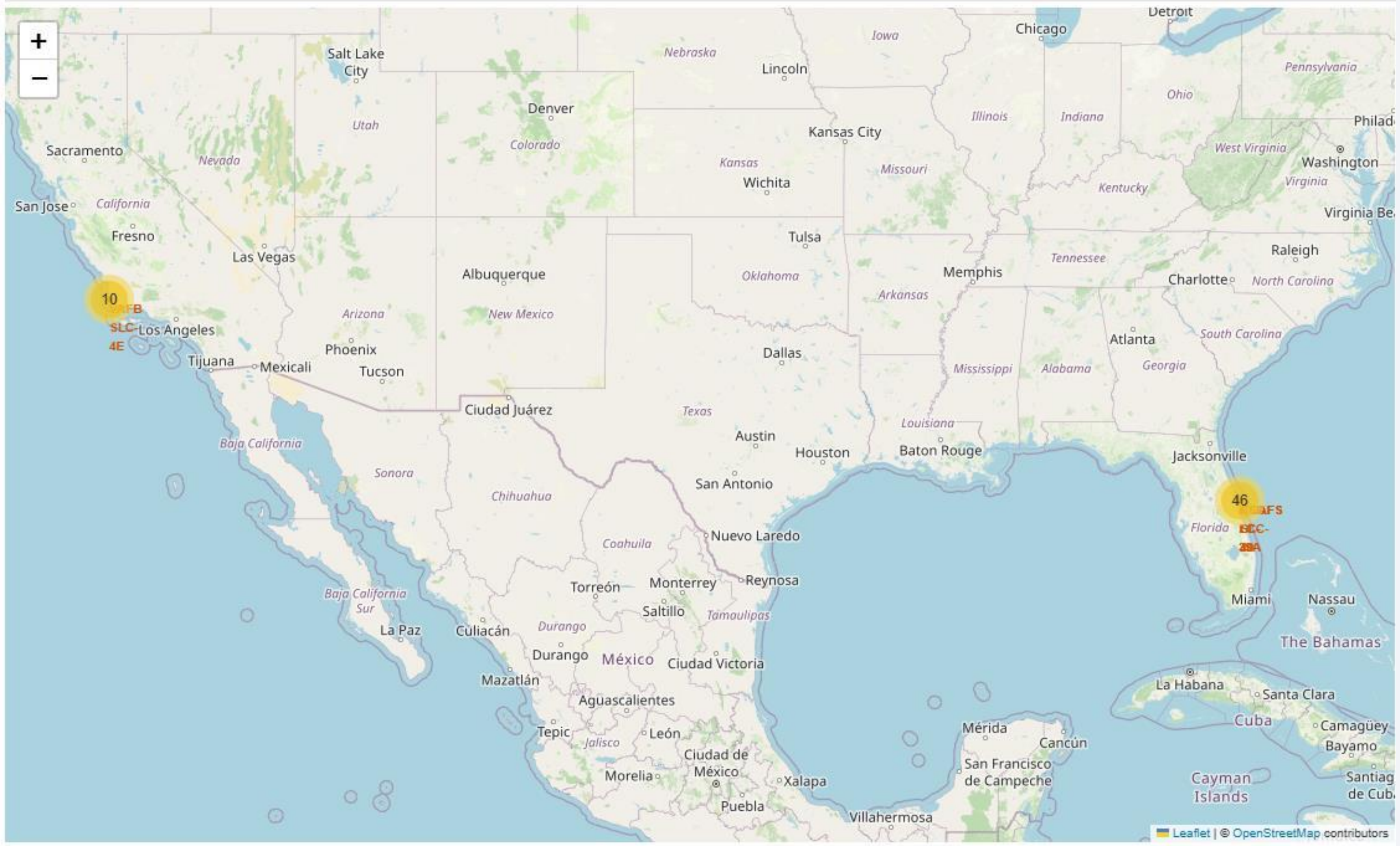
Landing_Outcome	land_count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

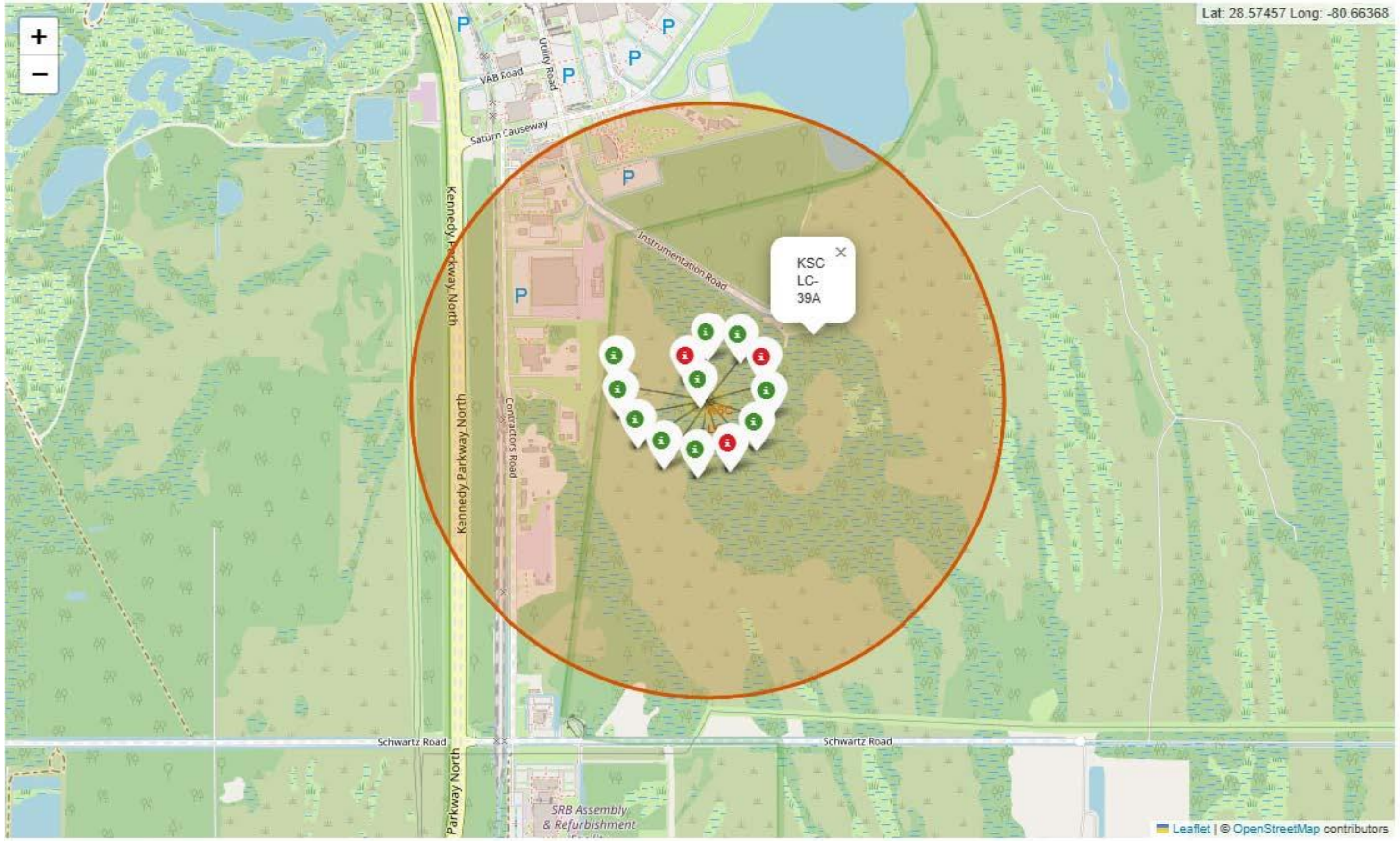
Launch Site Locations

- Evidently, launch sites are all located in coastal states.



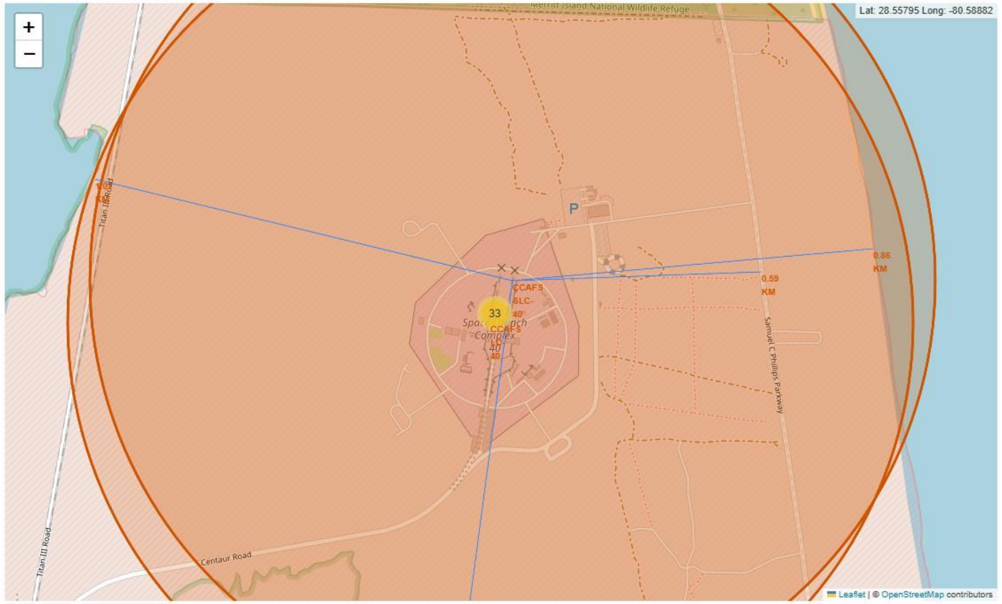
Most Successful Launch Site

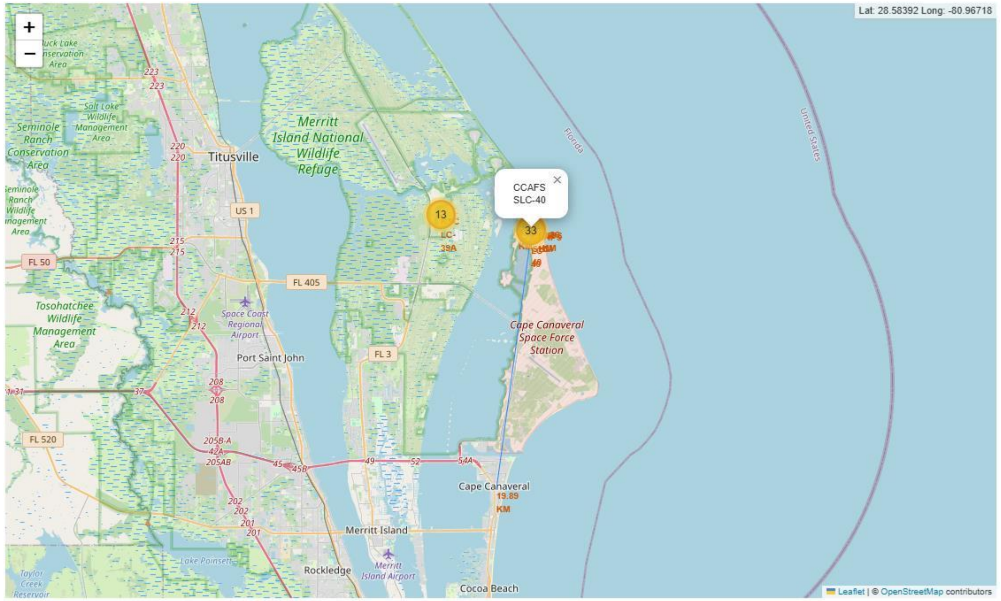
- The site with the highest success rate is KSC LC-39A.



Landmark Distance from Launch Site

- Analyzing the distance of landmarks from launch sites revealed trends.Railroads, highways, and the coastline are usually quite close to launch sites, while major cities are much further away. For example, a railroad, highway, and the coastline are within a kilometer from the launch site. The closest city is almost 20 kilometers away.


Lat: 28.55795 Long: -80.58882
Merritt Island National Wildlife Refuge
Titan III Road
0.86 KM
0.59 KM
33
CCAFS SLC-40
CCAFS LC-40
Space Launch Complex 40
Samuel C Phillips Parkway
Centaur Road
Titan III Road
Leaflet | © OpenStreetMap contributors


Lat: 28.58392 Long: -80.96718
Titusville
Merritt Island National Wildlife Refuge
Seminole Ranch Conservation Area
Salt Lake Wildlife Management Area
Tosohatchee Wildlife Management Area
Space Coast Regional Airport
Port Saint John
US 1
FL 50
FL 405
FL 3
FL 520
13
LC-39A
CCAFS SLC-40
33
Cape Canaveral Space Force Station
Cape Canaveral
19.89 KM
Merritt Island
Rockledge
Merritt Island Airport
Cocoa Beach
Lake Poinsett
Florida
United States
Leaflet | © OpenStreetMap contributors

<Folium Map Screenshot 2>

- Replace <Folium map screenshot 2> title with an appropriate title

- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

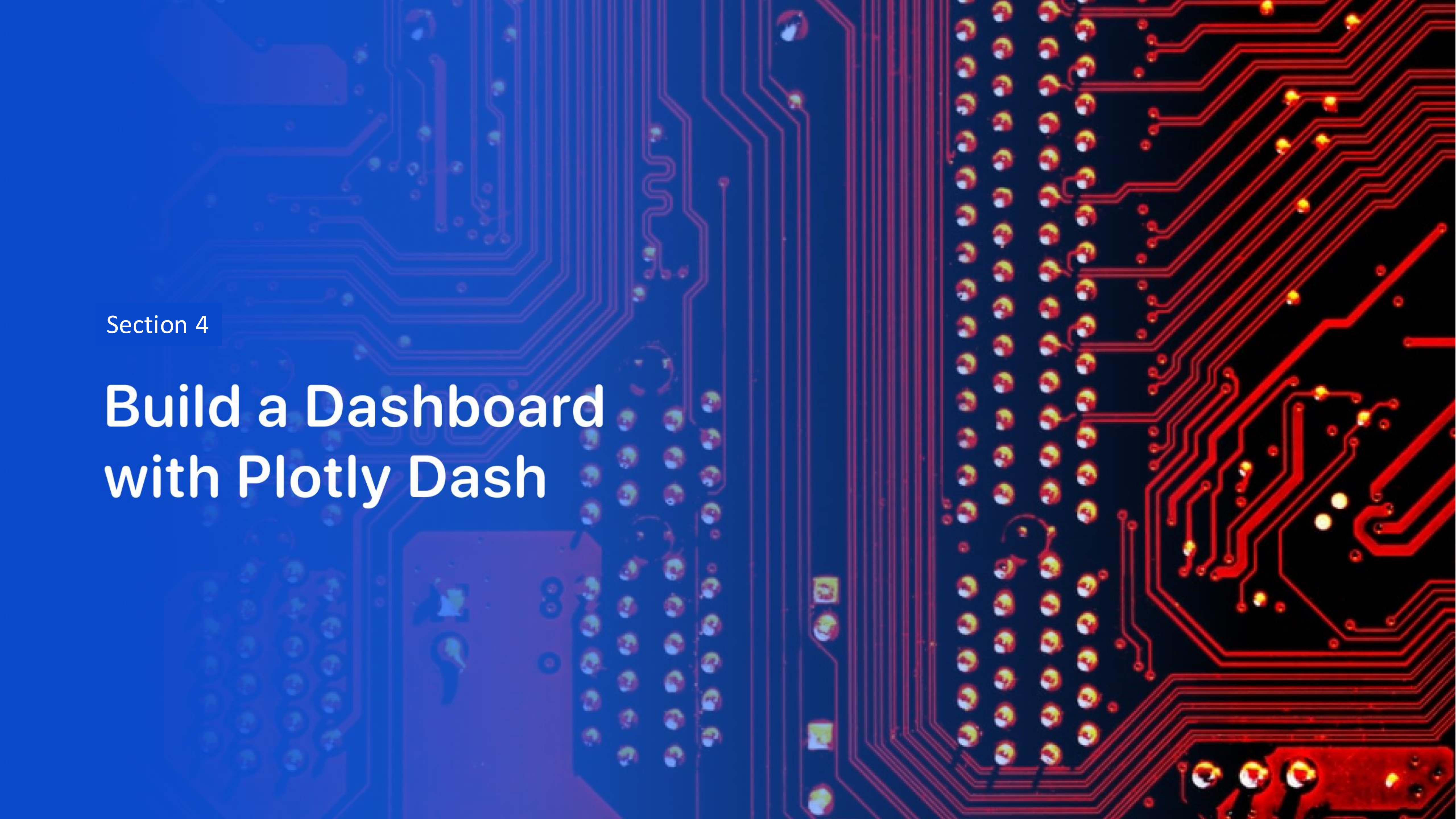
- Explain the important elements and findings on the screenshot

<Folium Map Screenshot 3>

- Replace <Folium map screenshot 3> title with an appropriate title

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

- Explain the important elements and findings on the screenshot

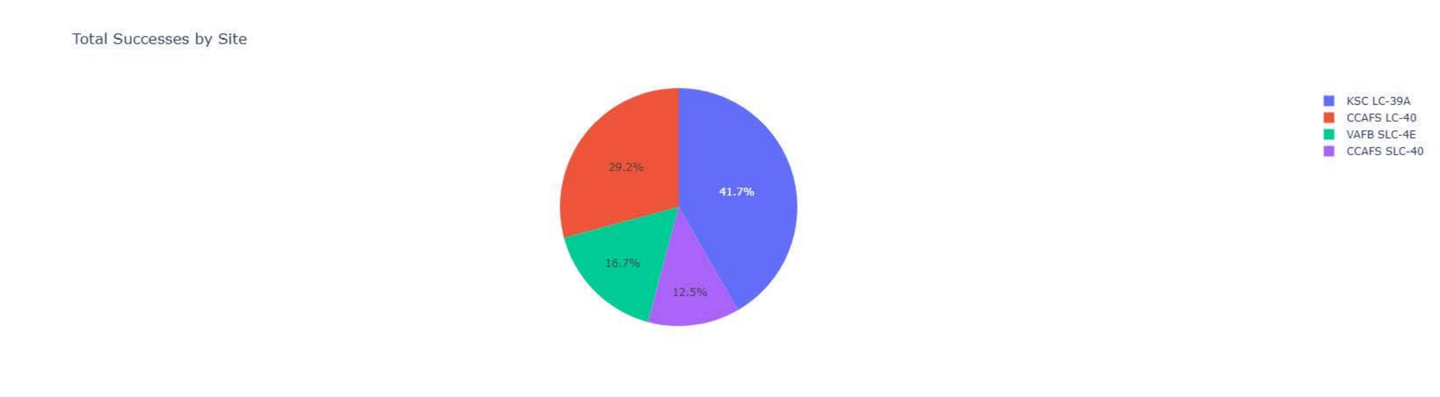


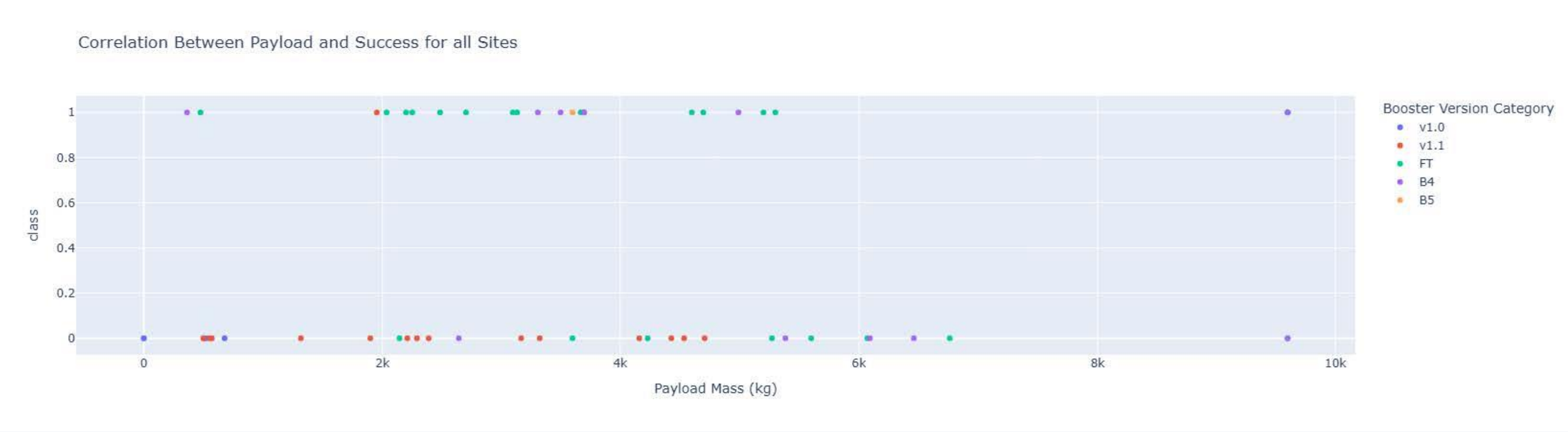
Section 4

Build a Dashboard with Plotly Dash

Finding across all sites

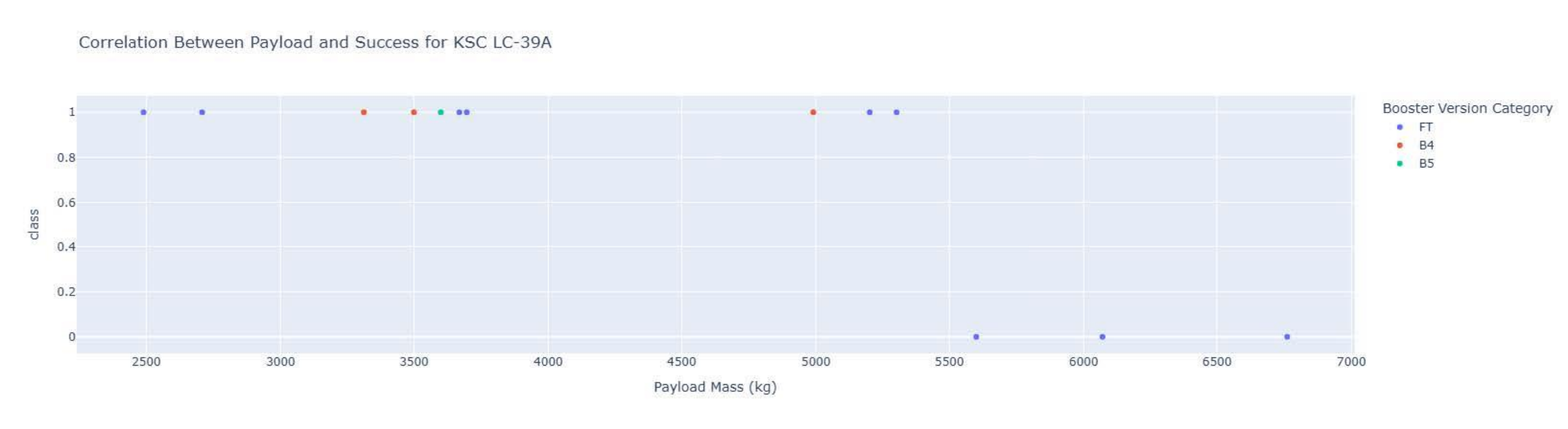
- The site with the highest success rate is KSC LC-39A, while the site with the lowest success rate is CCAFS SLC-40. FT boosters seem to do well, but v1.1 boosters seem to perform poorly.

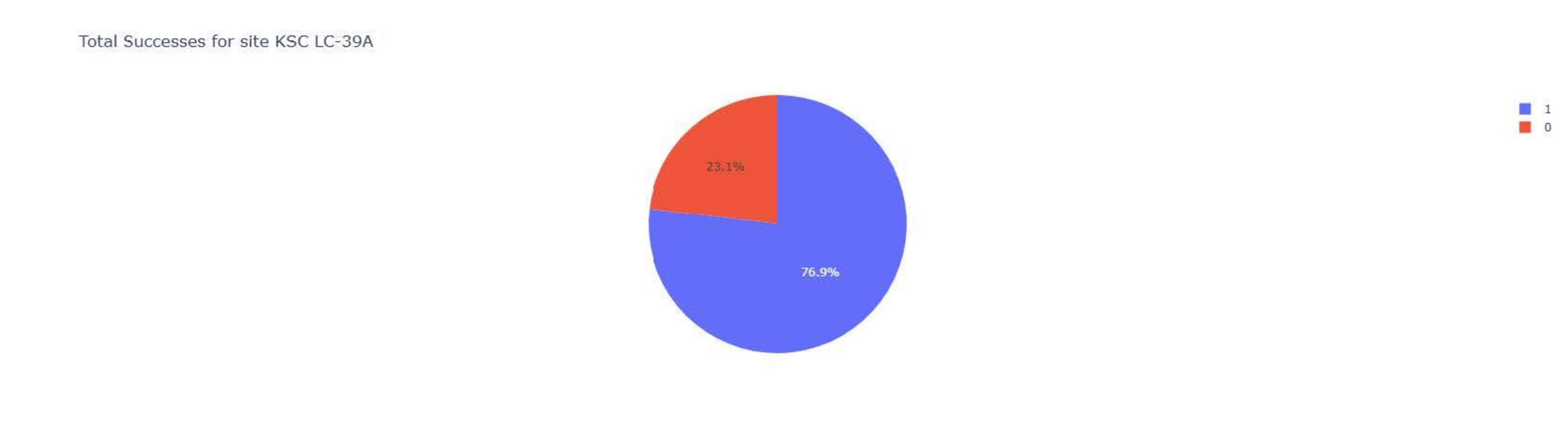




Highest Success Rate Site

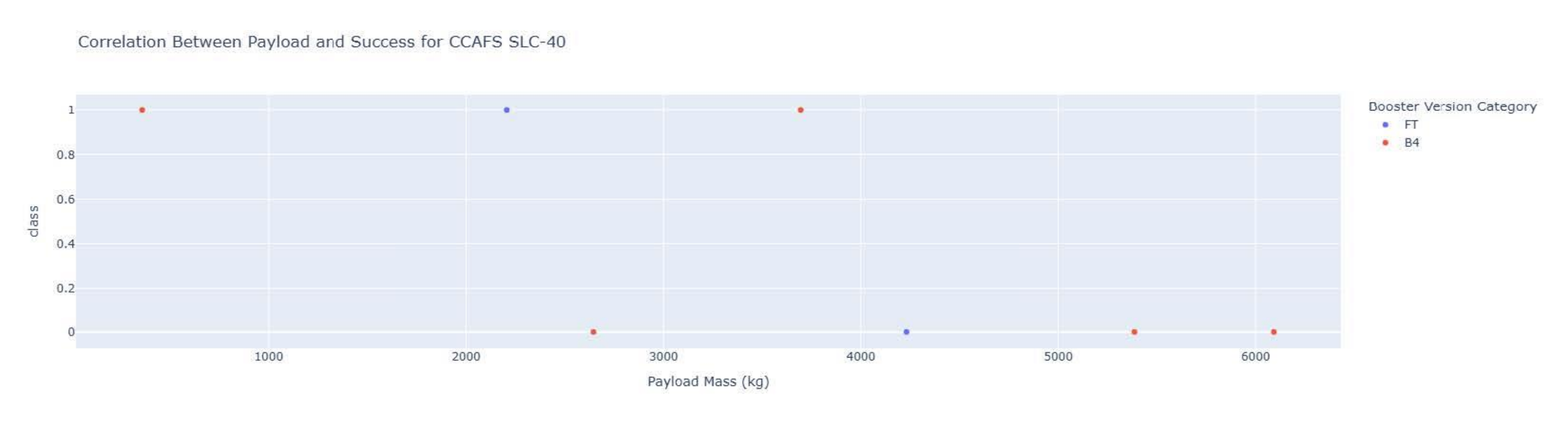
- The most successful site, KSC LC-39A, had successes ~77% of the time. After beginning to launch rockets with higher payloads and only using FT boosters, launches began to fail.

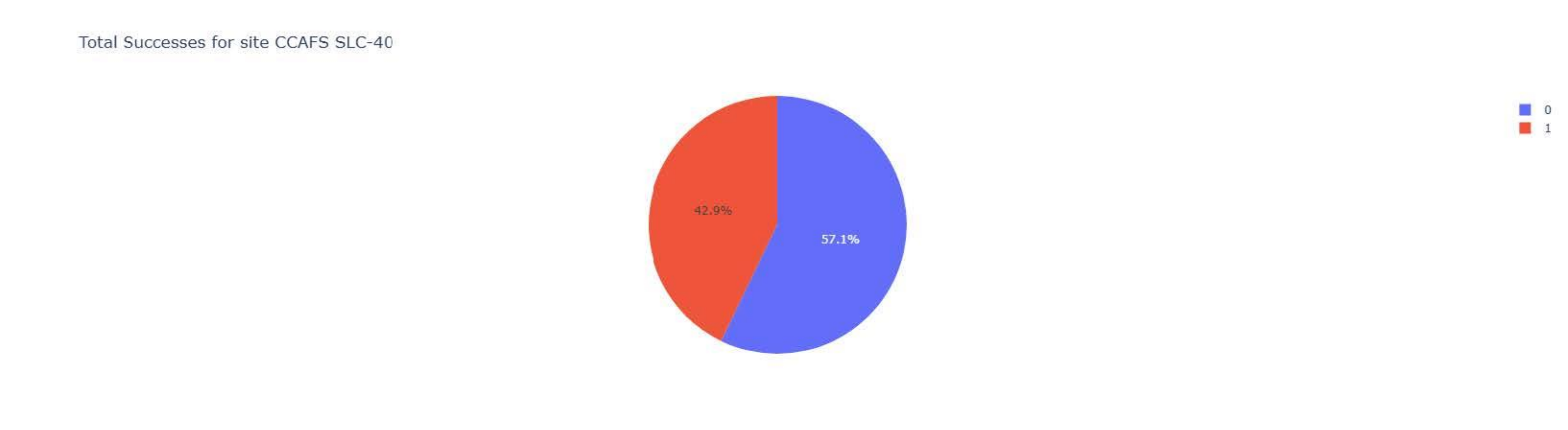




Lowest Success Rate Site

- The site with the lowest success rate, CCAFS SLC-40, failed ~57% of the time. Booster type, payload mass, and time don't seem to be correlated with success.



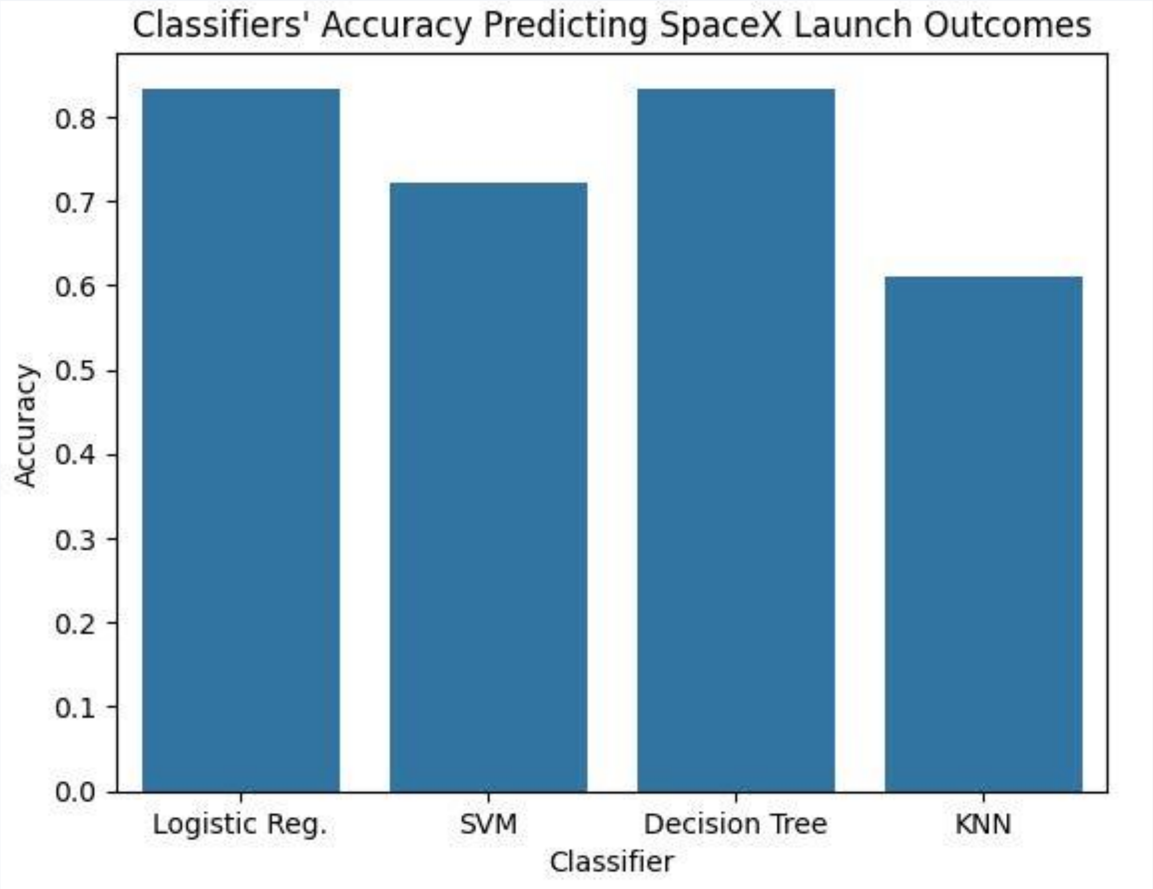


Section 5

Predictive Analysis (Classification)

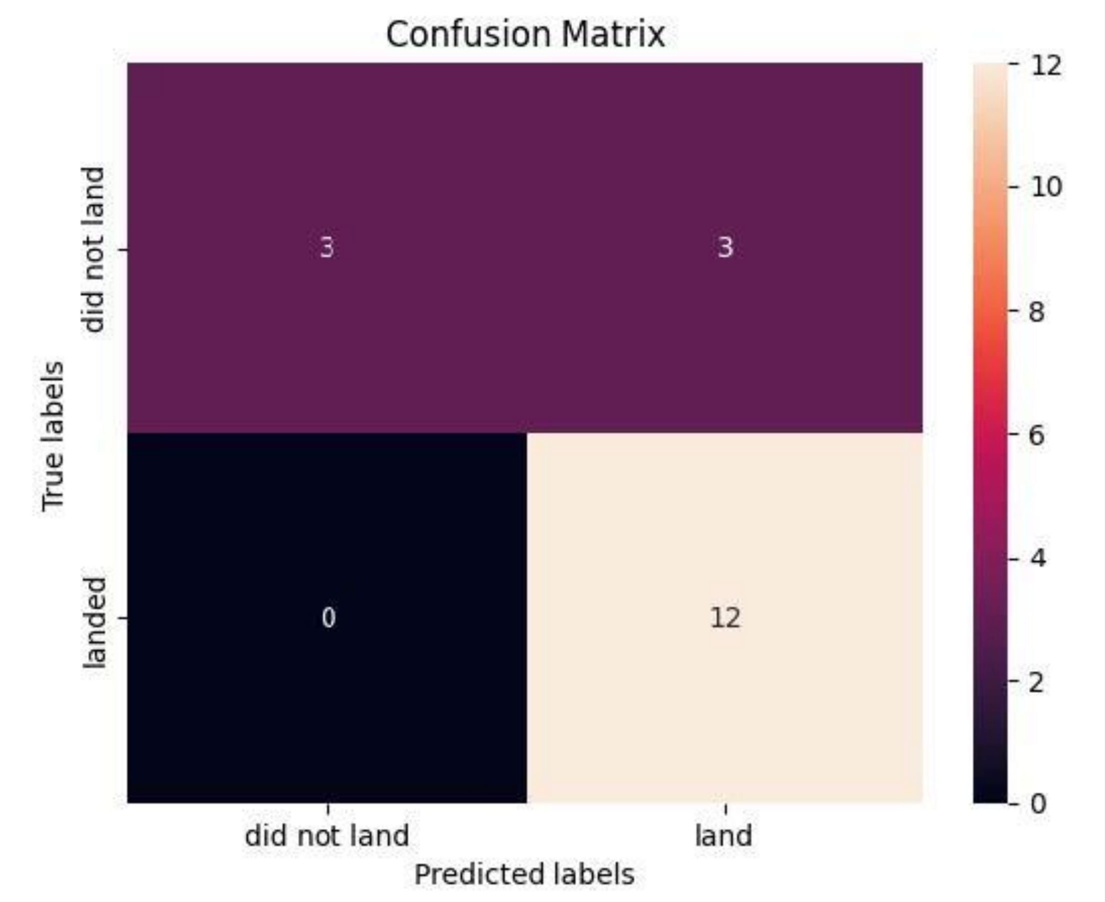
Classification Accuracy

- Though decision tree had higher accuracy during cross-validation, it had the same accuracy as logistic regression when evaluated against the test dataset.



Confusion Matrix

- Both the logistic regression and decision tree models had identical confusion matrices, only lacking in recall. These aren’t necessarily good models for this situation, but with experimentation a new model could sacrifice accuracy for higher recall. This would make it more overly cautious, but it would be much better at identifying launches that will fail.



Conclusions

- KSC LC-39A has the highest success rate out of all sites.
- Either decision tree or logistic regression are the best algorithms to use to predict success.
- Launch sites have close proximity to the coast and are far from major cities.
- Across all sites, time has increased success rate slightly.
- Most launches with payload mass above 5000 kg fail.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!