IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ben Lo
12 21 2023

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context

Commercial space age is here. With millions of investment, SpaceX has tested parameters optimal for successful landing. As a startup company SpaceY intended to compete with SpaceX, we will mine the public dataset from spaceX to retrieve the optimal parameters and predictive models to accelerate spaceship design. In this project, we will be focusing on the successful landing of the Falcon 9 model in the first stage.

- Problems you want to find answers
  - What are the optimal parameters (spaceship launch sites, booster version, payload) that have the best success?
  - Can we leverage these parameters to build machine learning model to predict mission success?

Section 1

Methodology

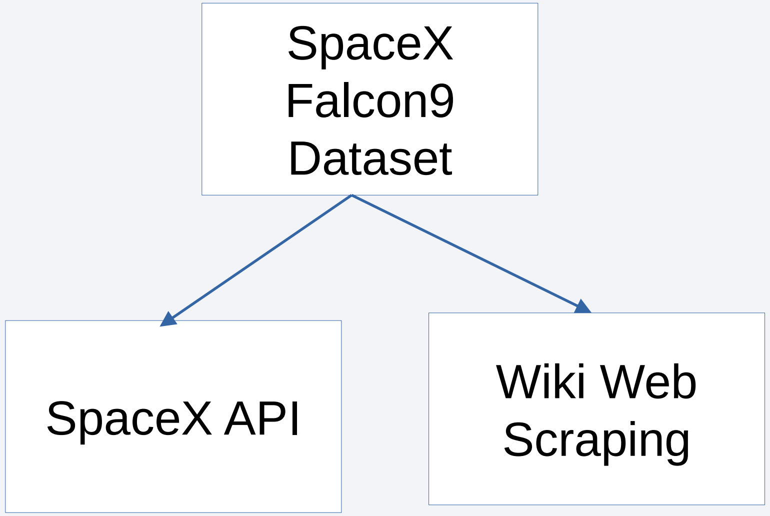
Methodology

Executive Summary

- Data collection methodology:
  - The data will be collected from spaceX API and web scraping form Wiki.
- Perform data wrangling
  - The missing data will be imputed using mean value.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - We will evaluate several ML models (log-reg, decision tree, SVM, kNN) and their

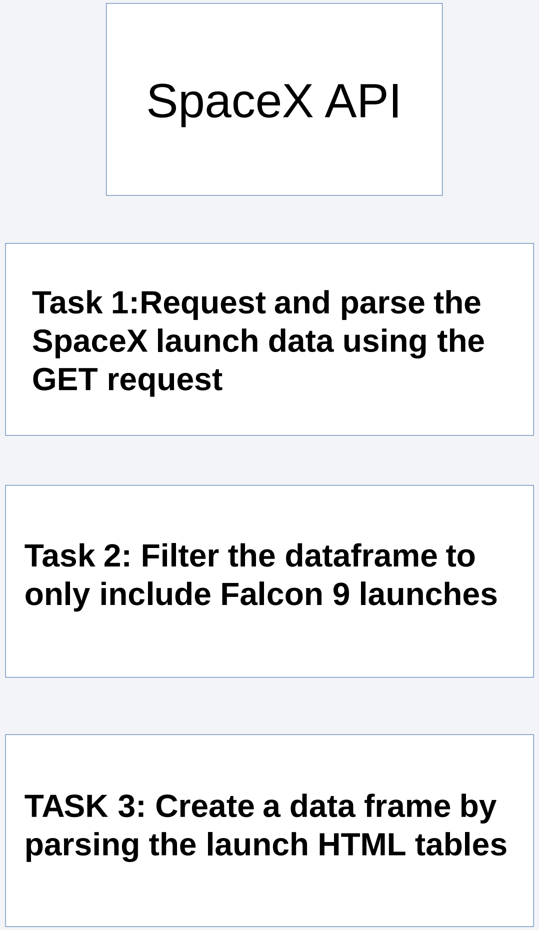
Data Collection

Data collection were conducted using SPACEX API and web scraping from the Wiki page.



Data Collection – SpaceX API

- Data collection workflow with SpaceX REST calls.



SpaceX API

Task 1:Request and parse the SpaceX launch data using the GET request

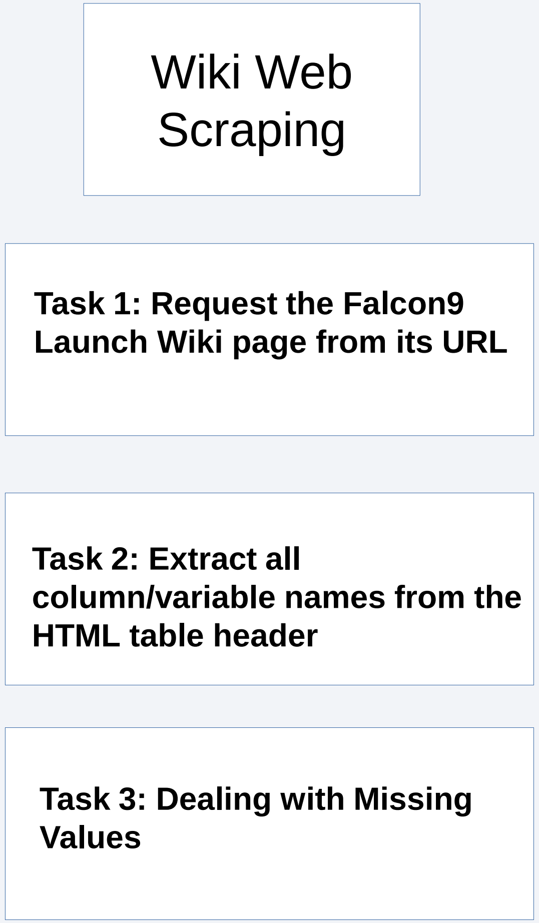
Task 2: Filter the dataframe to only include Falcon 9 launches

TASK 3: Create a data frame by parsing the launch HTML tables

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection – Web Scraping

- Data collection workflow with Wiki web scraping.



Wiki Web Scraping

Task 1: Request the Falcon9 Launch Wiki page from its URL

Task 2: Extract all column/variable names from the HTML table header

Task 3: Dealing with Missing Values

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad.True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship.

we will mainly convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

Flight number vs Payload mass: We can plot out the FlightNumber vs. PayloadMassand overlay the outcome of the launch. We see that as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; it seems the more massive the payload, the less likely the first stage will return.

Flight number vs Launch site: We see that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.

Payload mass vs Launch site: for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success rate vs Orbit type: orbit types like ESL-1, GEO, HEO and SSO have better success rate.

Flight number vs Orbit type: we see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload mass vs Orbit type: With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

launch success yearly trend: we observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

1. %sql select * from SPACEXTABLE; Display the names of the unique launch sites in the space mission.
2. %sql select * from SPACEXTABLE where Launch_Site Like 'CCA%'; Display 5 records where launch sites begin with the string 'CCA'
3. %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer Like '%NASA%'; Display the total payload mass carried by boosters launched by NASA (CRS).
4. %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'; Display average payload mass carried by booster version F9 v1.1
5. %sql select min(Date) from SPACEXTABLE where Landing_Outcome like '%Success%'; List the date when the first succesful landing outcome in ground pad was acheived.
6. %sql select Booster_Version from SPACEXTABLE where Landing_Outcome like '%Success%' and PAYLOAD_MASS__KG_ 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.
7. %sql select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome; List the total number of successful and failure mission outcomes
8. %sql select distinct(Booster_Version) from SPACEXTABLE where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTABLE); List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
9. %sql select substr(Date, 6,2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where substr(Date,0,5)='2015' and Landing_Outcome like '%Failure%';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.
10. %sql select Landing_Outcome, count(*) as total from SPACEXTABLE where Date between "2010-06-04" and "2017-03-20" group by Landing_Outcome order by total desc; 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.

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Task 1: Mark all launch sites on a map: Create and add folium.Circle and folium.Marker for each launch site on the site map.
Task 2: Mark the success/failed launches for each site on the map: for each launch result in spacex_df data frame, add a folium.Marker to marker_cluster
TASK 3: Calculate the distances between a launch site to its proximities: Mark down a point on the closest coastline using MousePosition and calculate the distance between the coastline point and the launch site.

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Component 1: Dropdown menu providing options for selecting all launch sites or specific launch site.

Component 2: For each selected option, if it is all sites, showed %success rate for each site in pie chart. For selected site, for %success vs failure for the site.

Component 3: A slidebar for defining the range of payload mass.

Component 4: For each option and payload range, shows payload mass versus success category colored by Booster version category.

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Task 1: Create a NumPy array from the column Class in data, by applying the method to_numpy() then assign it to the variable Y,make sure the output is a Pandas series (only one bracket df['name of column']).
Task 2: Standardize the data in X then reassign it to the variable X using the transform Standard Scaler function.
Task 3: Use the function train_test_split to split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2.
Task 4: Create a estimator object then create a GridSearchCV with cv = 10. Fit the object to find the best parameters from the dictionary parameters using GridSearchCV.
Task 5: Calculate the accuracy on the test data using the method score.
Task 6: Repeat 4-5 for SVM, kNN, and decision tree.
Take 7: Find the optimal ML model for predicting Falcon9 landing success.

Load the Falcon 9 data set in Pandas

Normalize the feature using StandardScaler

Split training and test set with 2/8 split

Create ML estimator object and define hyperparameters

Fit model with GridSearchCV(CV=10)

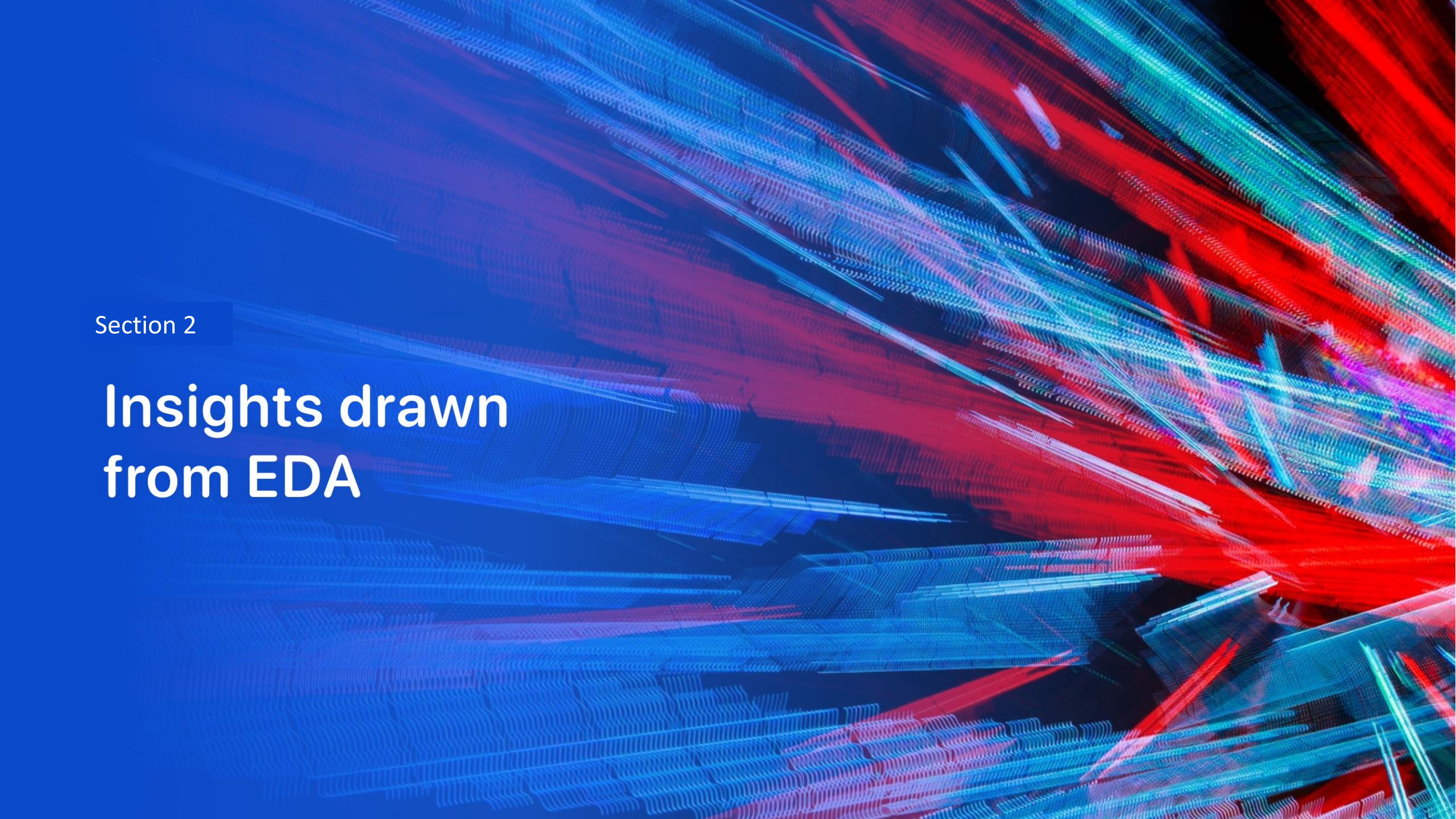
Evaluate accuracy on test data

Identify optimal ML model for predicting Falcon9 landing success

https://github.com/yuchen-lo/IBM_DS_Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

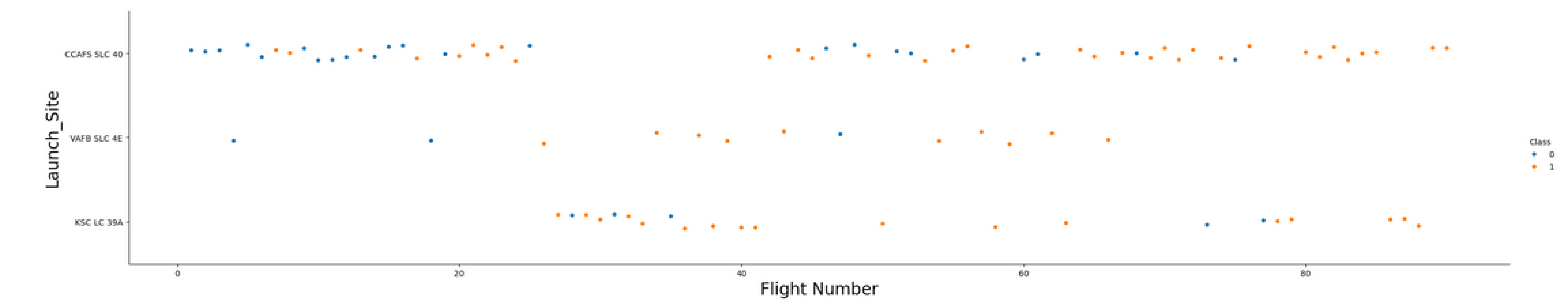
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



Section 2

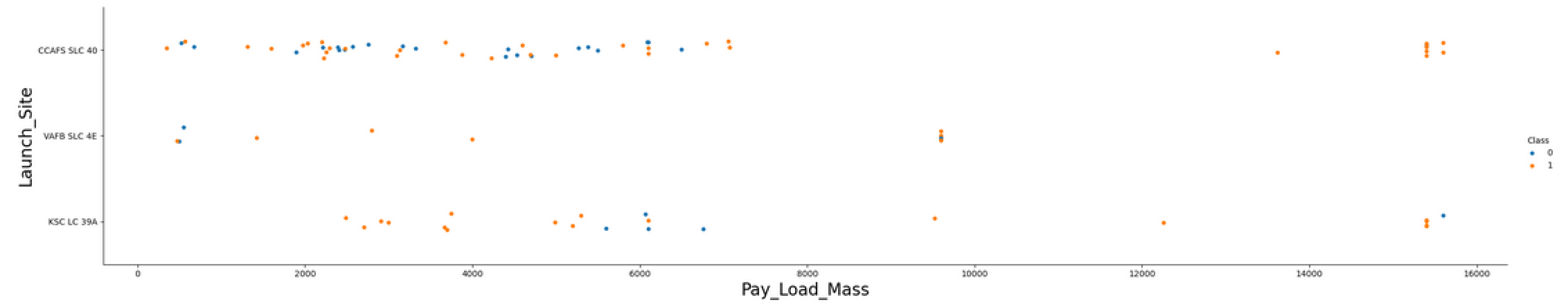
Insights drawn from EDA

Flight Number vs. Launch Site



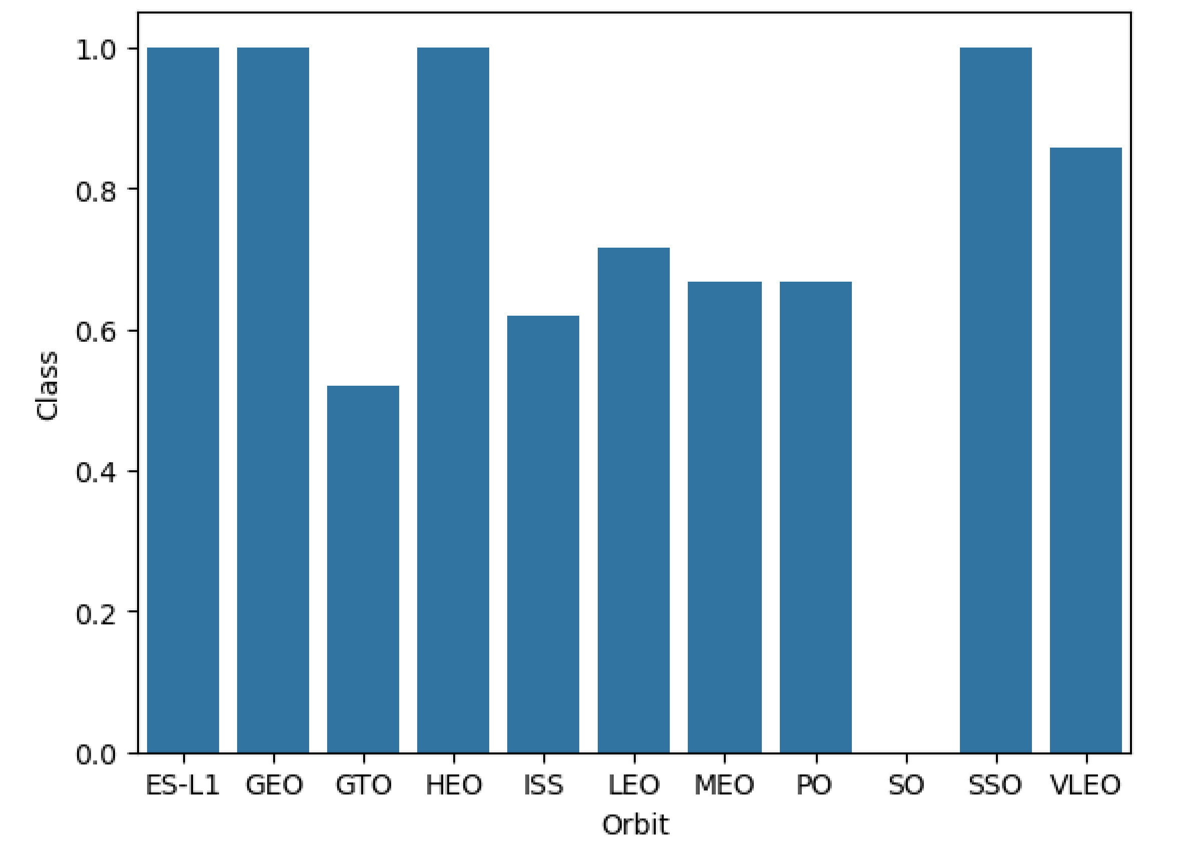
We see that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.

Payload vs. Launch Site



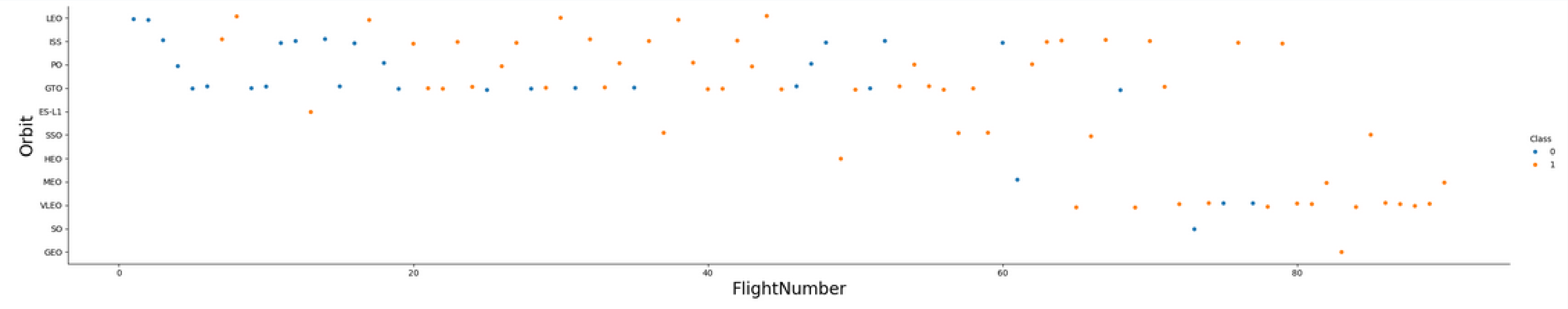
for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type



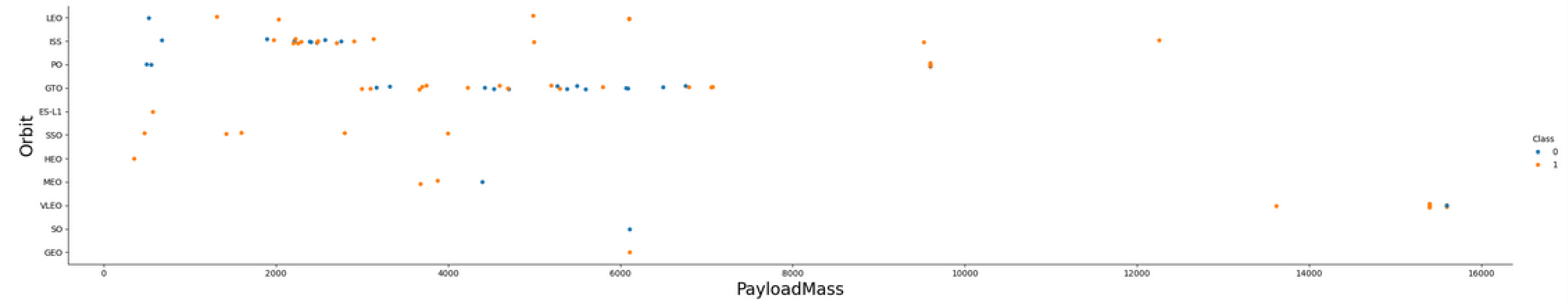
orbit types like ESL-1, GEO, HEO and SSO have better success rate.

Flight Number vs. Orbit Type



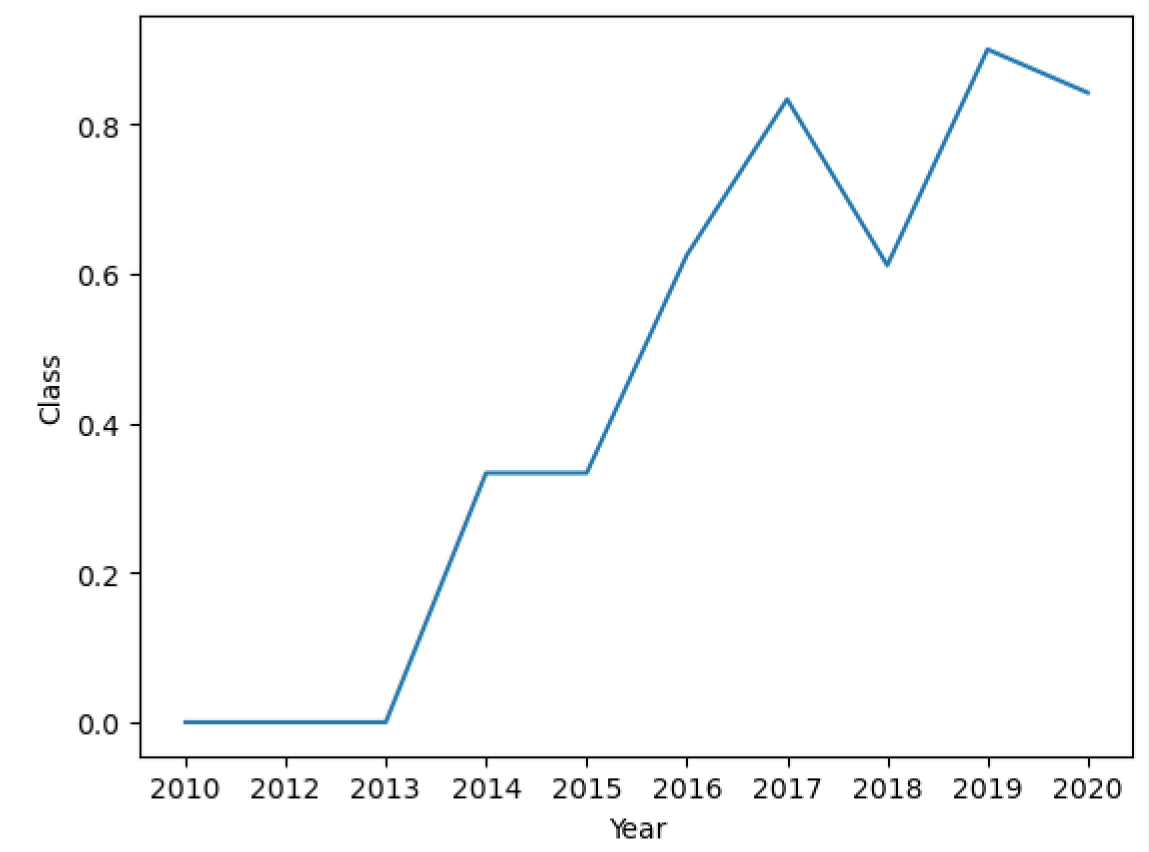
we see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend



we observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

1. **%sql select * from SPACEXTABLE;** Display the names of the unique launch sites in the space mission.

Launch Site Names Begin with 'CCA'

2. %sql select * from SPACEXTABLE where Launch_Site Like 'CCA%'; Display 5 records where launch sites begin with the string 'CCA'

Total Payload Mass

3. %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer Like '%NASA%'; Display the total payload mass carried by boosters launched by NASA (CRS).

Average Payload Mass by F9 v1.1

4. %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'; Display average payload mass carried by booster version F9 v1.1

First Successful Ground Landing Date

5. %sql select min(Date) from SPACEXTABLE where Landing_Outcome like '%Success%'; List the date when the first succesful landing outcome in ground pad was acheived.

Successful Drone Ship Landing with Payload between 4000 and 6000

6. %sql select Booster_Version from SPACEXTABLE where Landing_Outcome like '%Success%' and PAYLOAD_MASS__KG_ 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.

Total Number of Successful and Failure Mission Outcomes

7. %sql select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome; List the total number of successful and failure mission outcomes

Boosters Carried Maximum Payload

8. %sql select distinct(Booster_Version) from SPACEXTABLE where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTABLE);
List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

2015 Launch Records

9. %sql select substr(Date, 6,2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where substr(Date,0,5)='2015' and Landing_Outcome like '%Failure%';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.

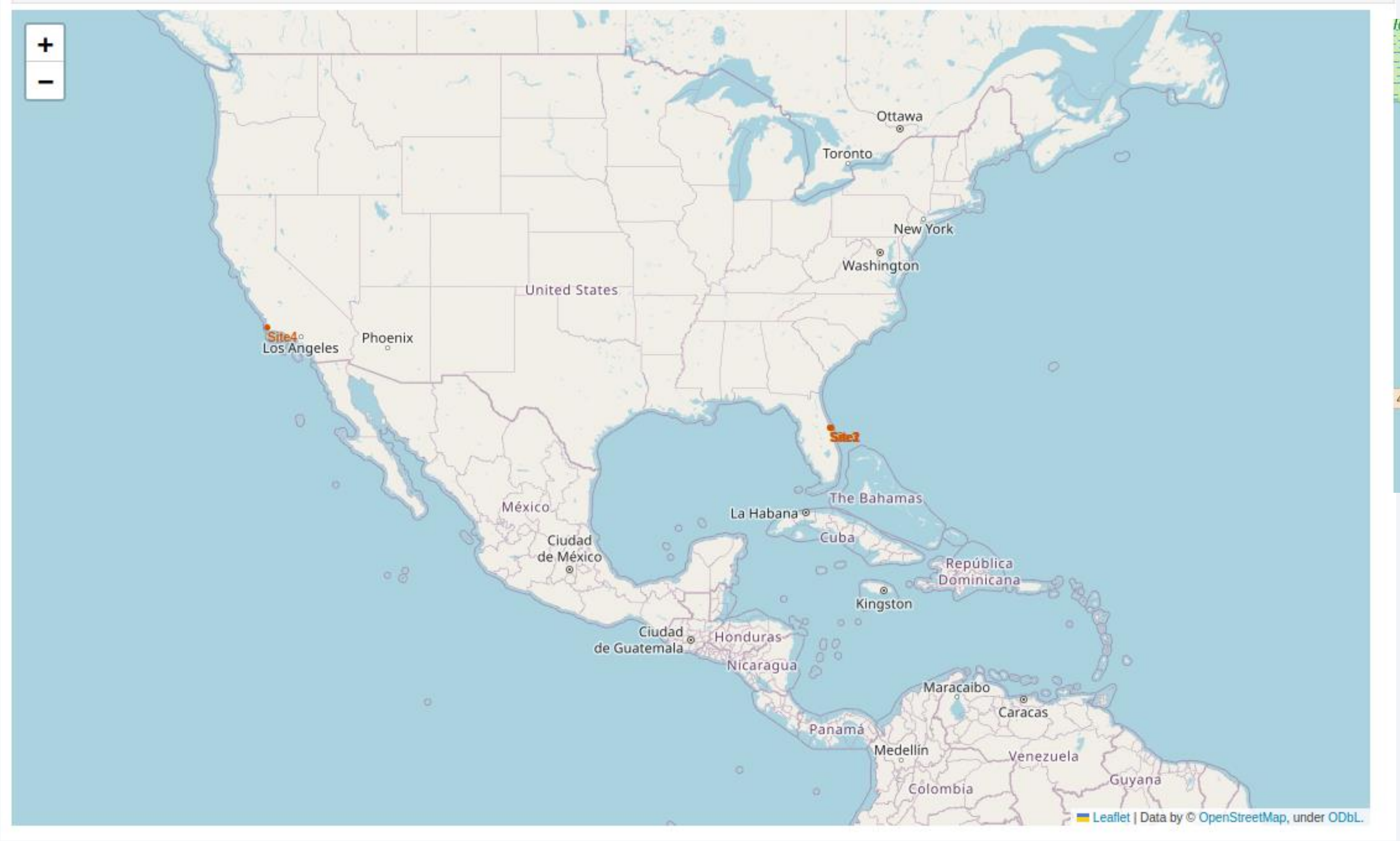
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

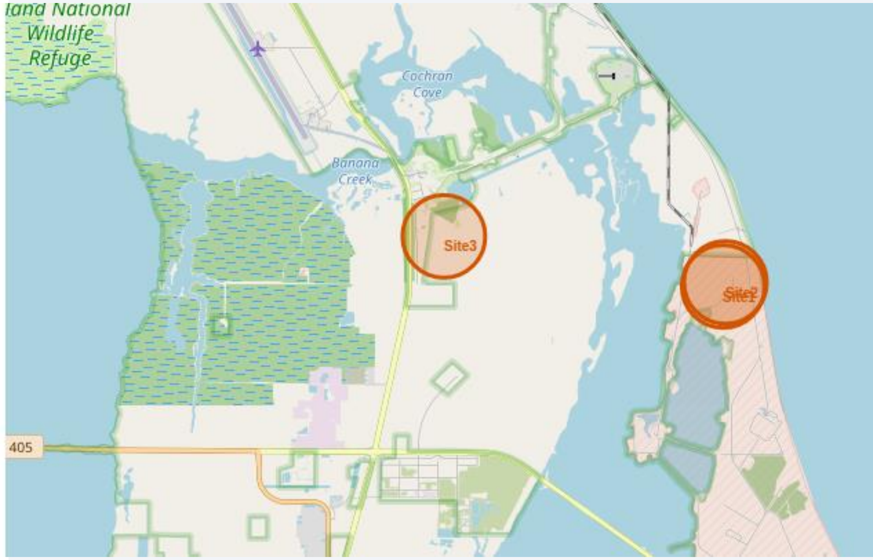
10. %sql select Landing_Outcome, count(*) as total from SPACEXTABLE where Date between "2010-06-04" and "2017-03-20" group by Landing_Outcome order by total desc; Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

Section 3

Launch Sites Proximities Analysis

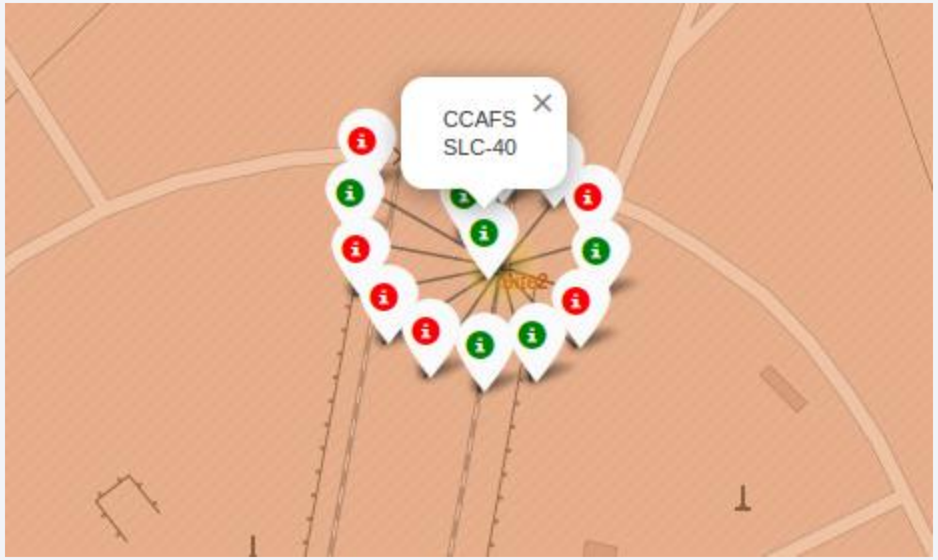
Mark all launch sites on a map

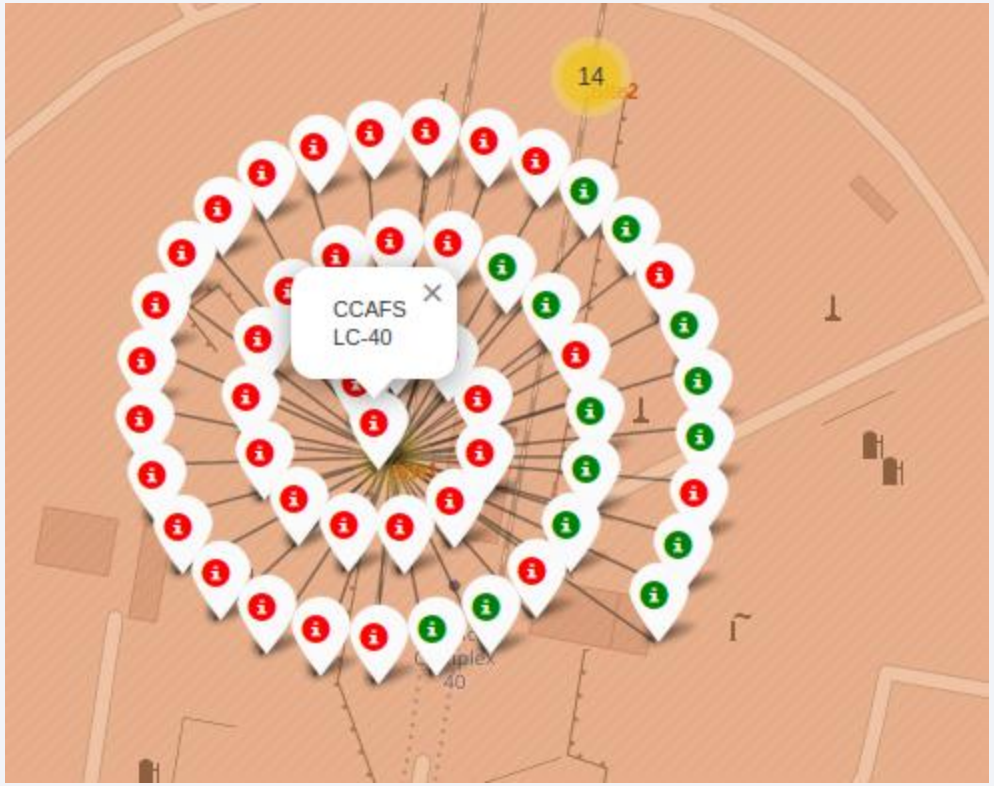




All launch sites are in proximity to the Equator line.
Site 1 and 2 are in proximity.

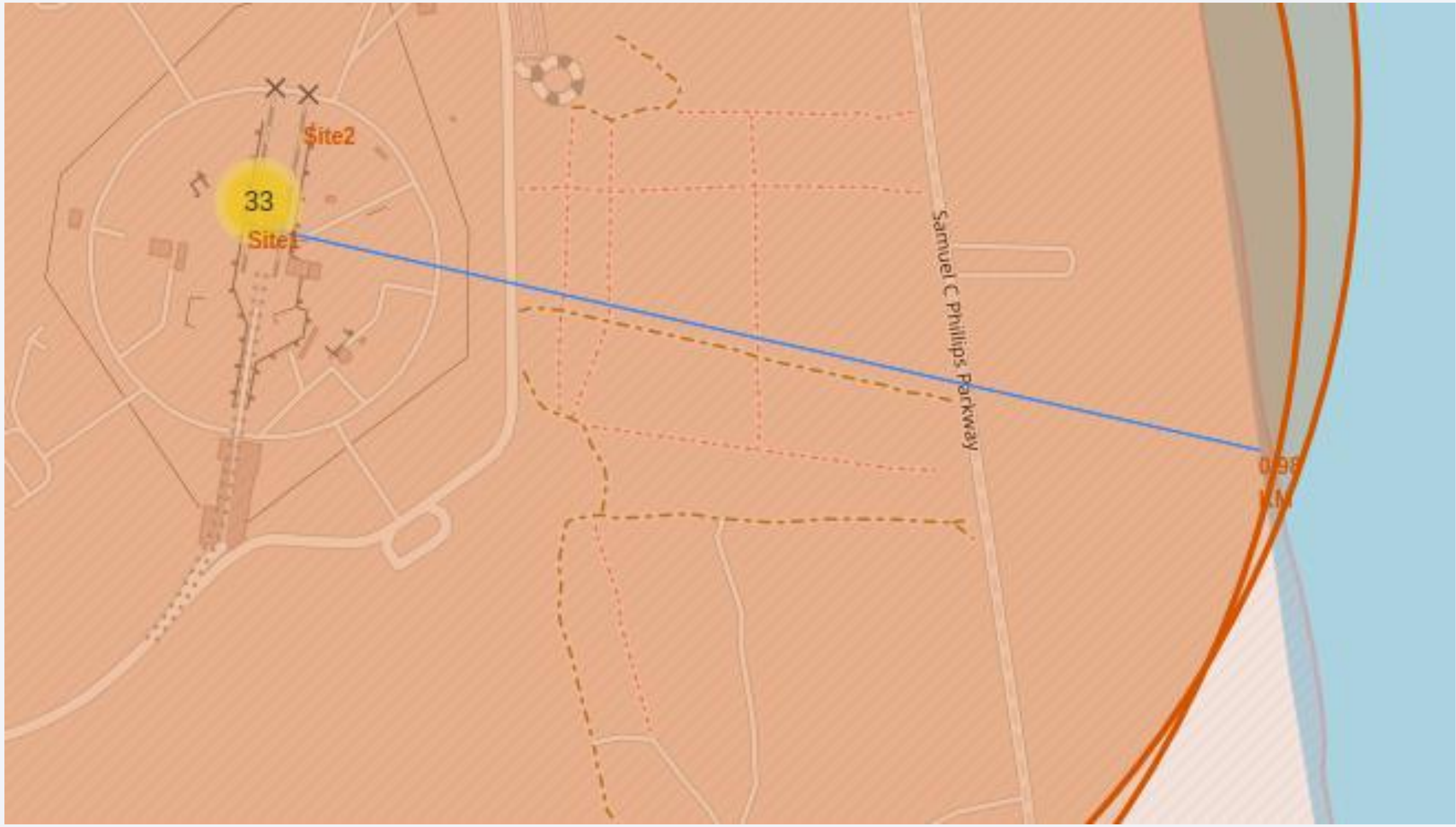
Mark the success/failed launches for each site on the map



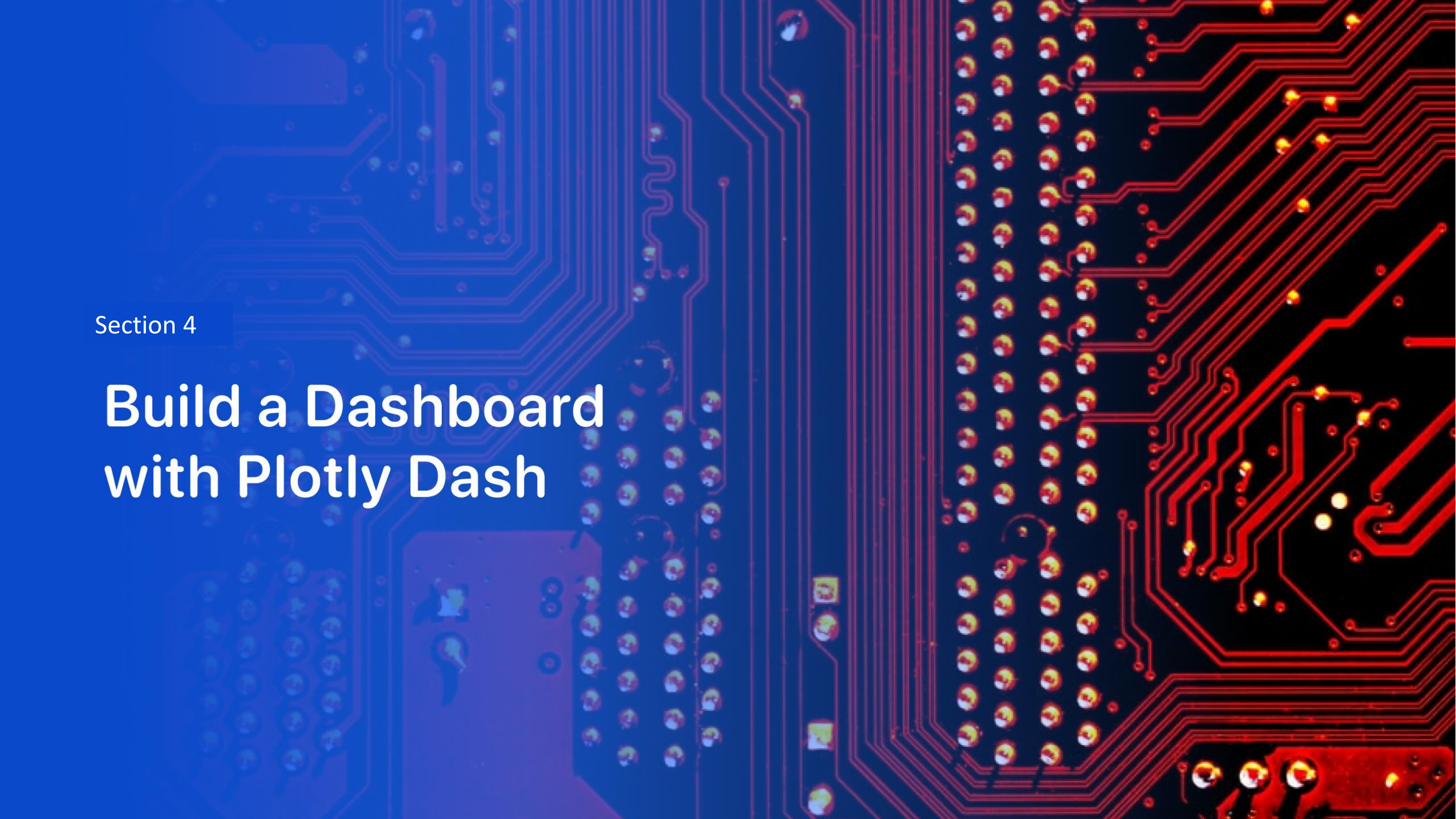


Success (green) and Failure (red) landing event on two sites.

Calculate the distances between a launch site to its proximities



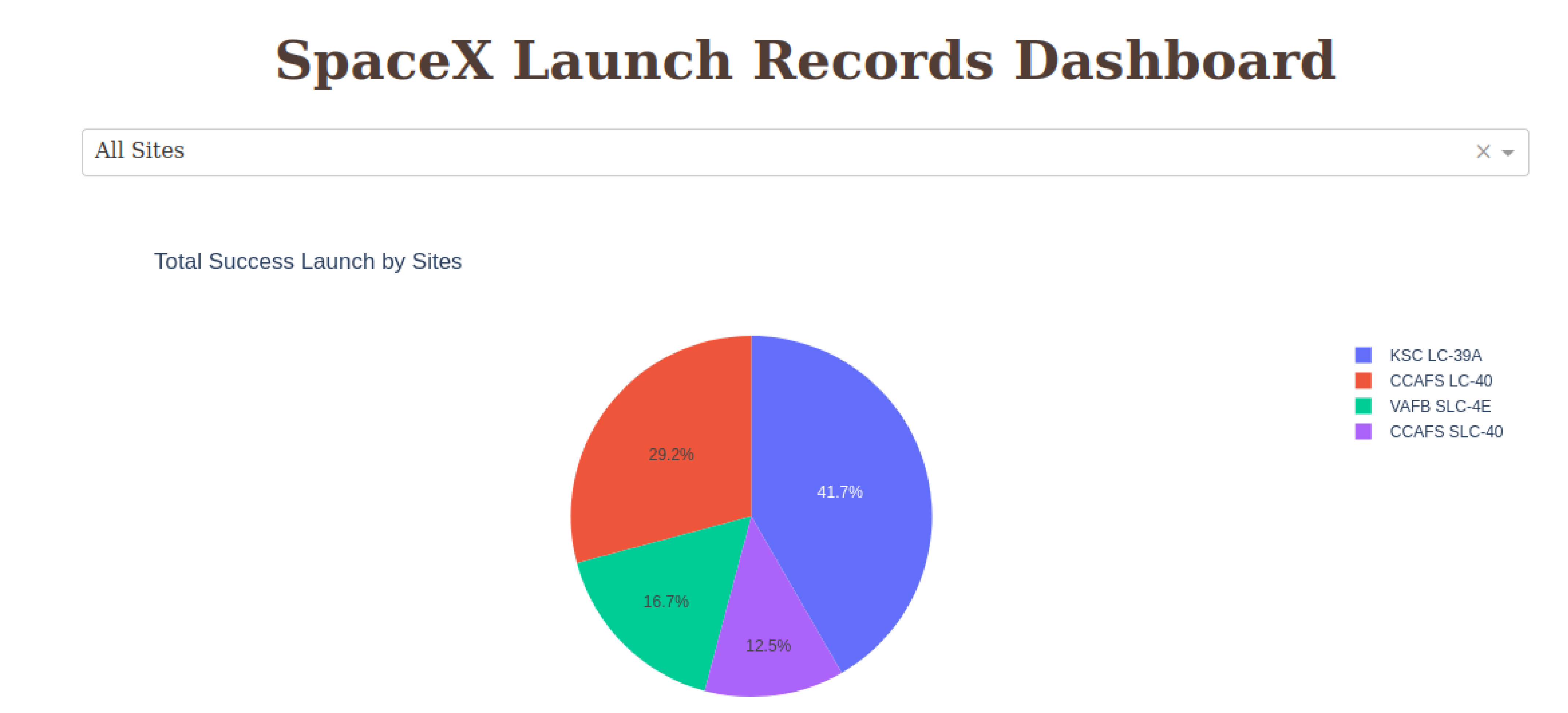
Nearest coastline is 0.98km from site 1.



Section 4

Build a Dashboard with Plotly Dash

SpaceX Launch Dash-All Sites

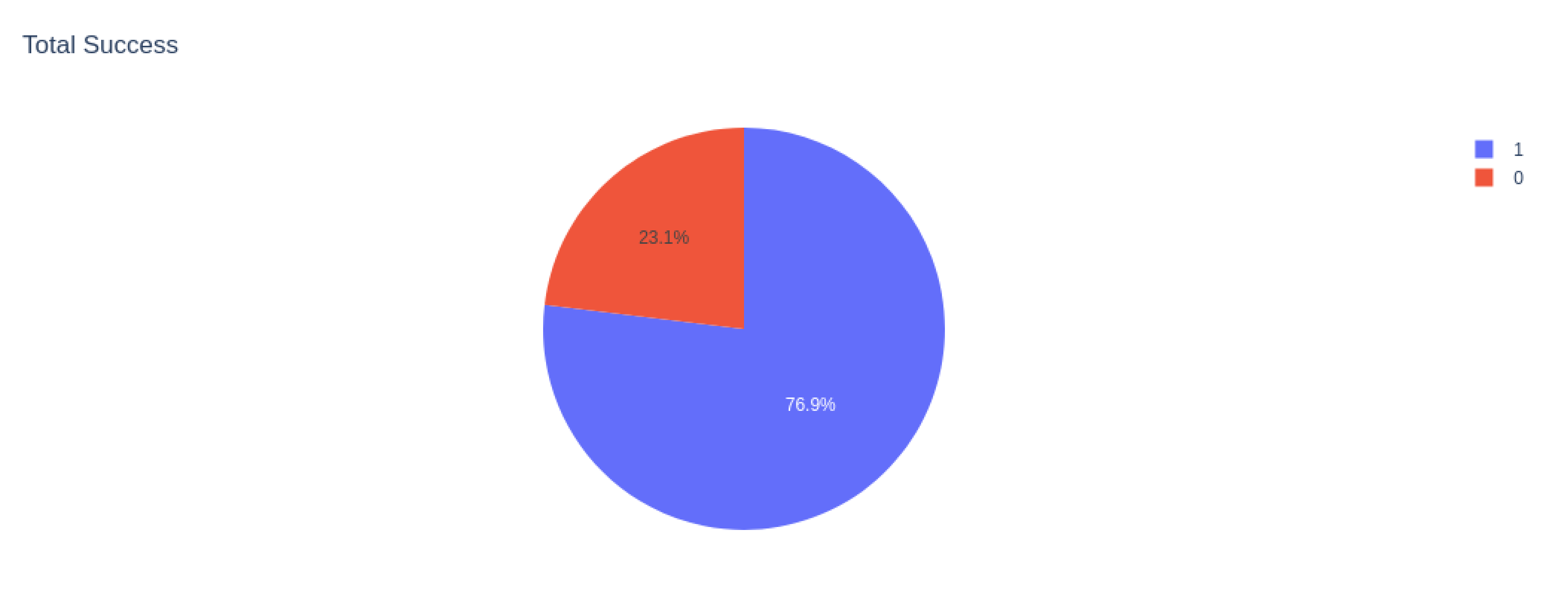


SpaceX Launch Records Dashboard

All Sites

Site KSC LC-39A has the highest success rate.

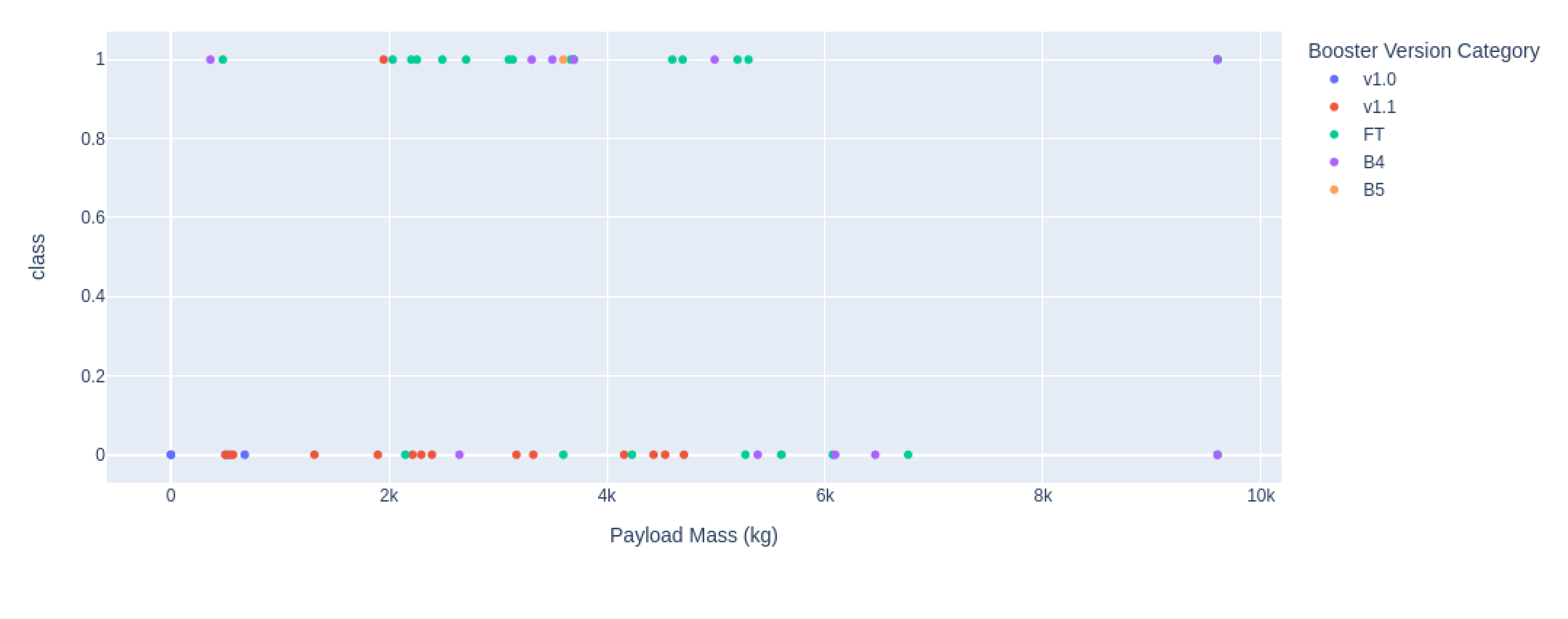
SpaceX Launch Dash-Best Site



Best site KSC LC-39A has the 77% success rate.

<Dashboard Screenshot 3>

SpaceX launch Dash-Best Booster Version

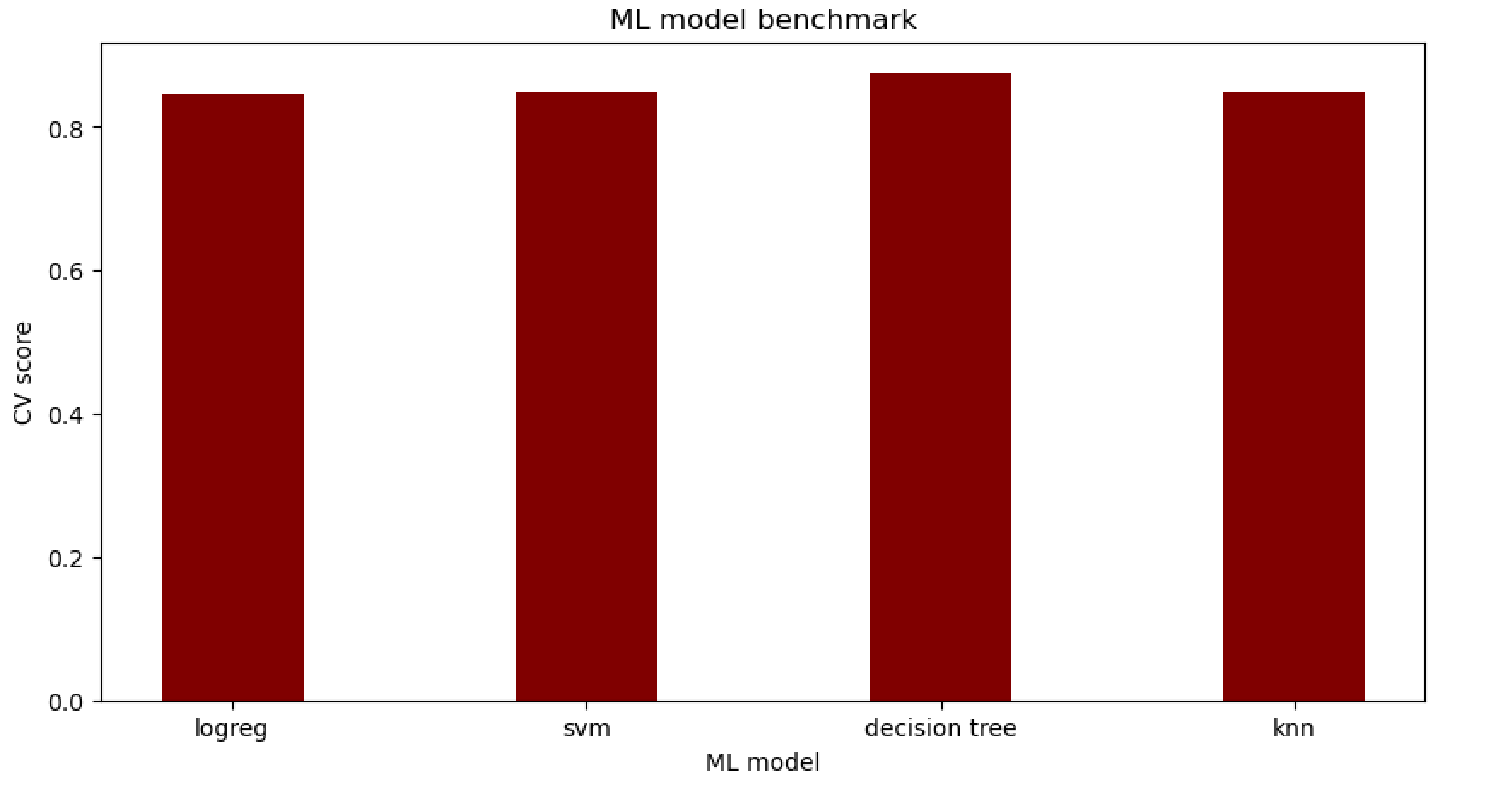


Booster version FT has the best success rate.

Section 5

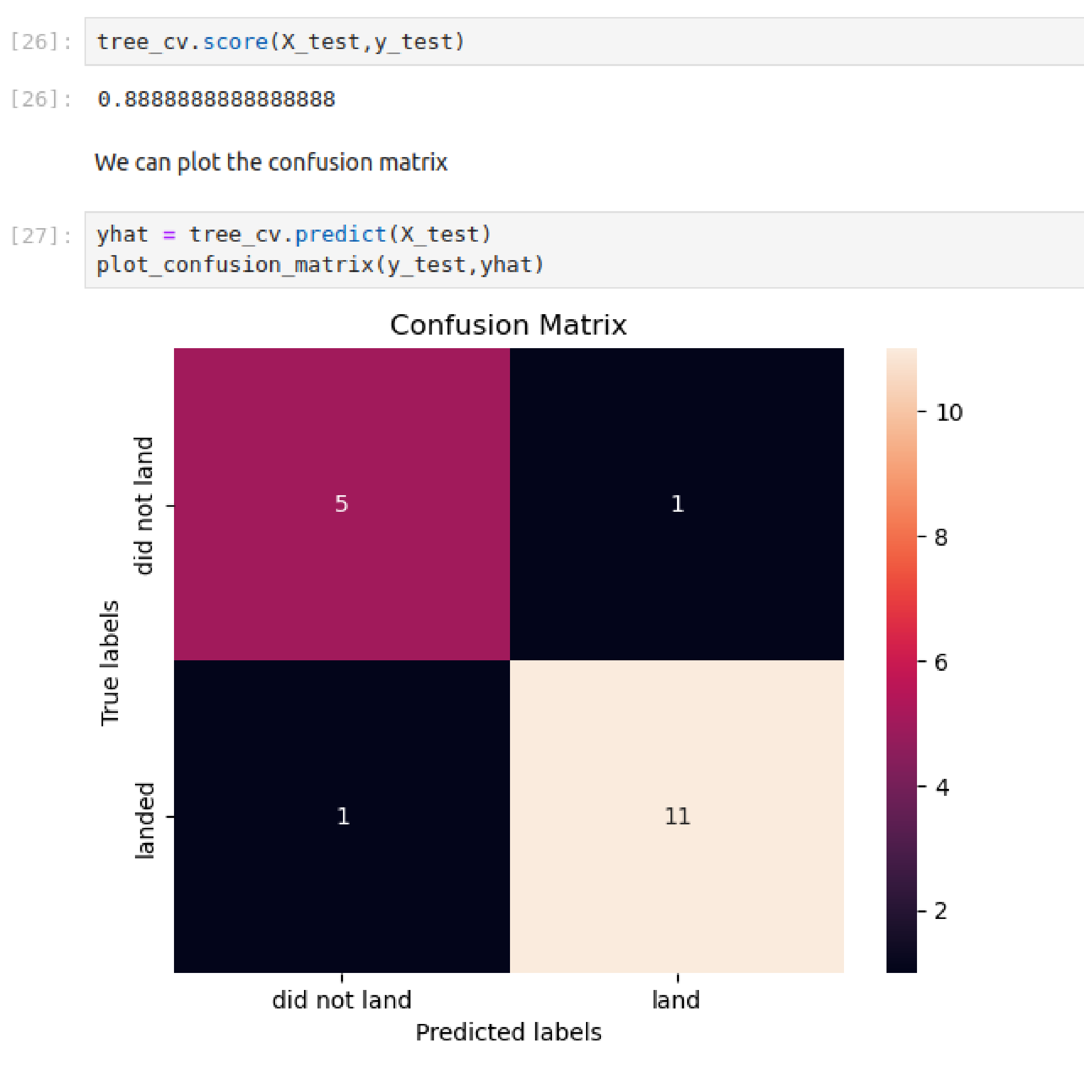
Predictive Analysis (Classification)

Classification Accuracy



Decision tree has the best performance.

Confusion Matrix



```
[26]: tree_cv.score(X_test,y_test)

[26]: 0.8888888888888888
```

We can plot the confusion matrix

```
[27]: yhat = tree_cv.predict(X_test)
      plot_confusion_matrix(y_test,yhat)
```

Decision tree has the best test performance. Confusion matrix shows only 1 FN and 1FP.

Conclusions

Point 1: **Data collection can be effective conducted using a combination of API query and webscraping.**

Point 2: **EDA shows several features were strongly correlated to landing success of Falcon 9.**

Point 3: **SPACEX Dash provides an intuitive and effective way to query complex dataset.**

Point 4: **Machine learning model can be used to predict landing success.**

Thank you!