Winning Space Race with Data Science

Stephen Sinocchi
4/10/2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - Two Data Collection approaches
    - SpaceX API using
    - Web Scraping from Wikipedia
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive visual analytics in screenshots
  - Predictive Analytics result

Introduction

- Project background and context

  Enable SpaceY to competitively bid against SpaceX for launches. Space X advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because Space X can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This goal is to project the data science methodology and create a machine learning pipeline to predict if the first stage will land successfully.

- Problems you want to find answers
  - What are the factors that determine a successful first stage landing or not
  - Condition that need to be in place for successful landing

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Utilize SpaceX APIs utilizing Python Request Library
  - Perform Wikipedia Web scrapping utilizing Python Beautiful Soup Library
- Perform data wrangling
  - One-hot encoding was applied to categorical outcome features
- 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

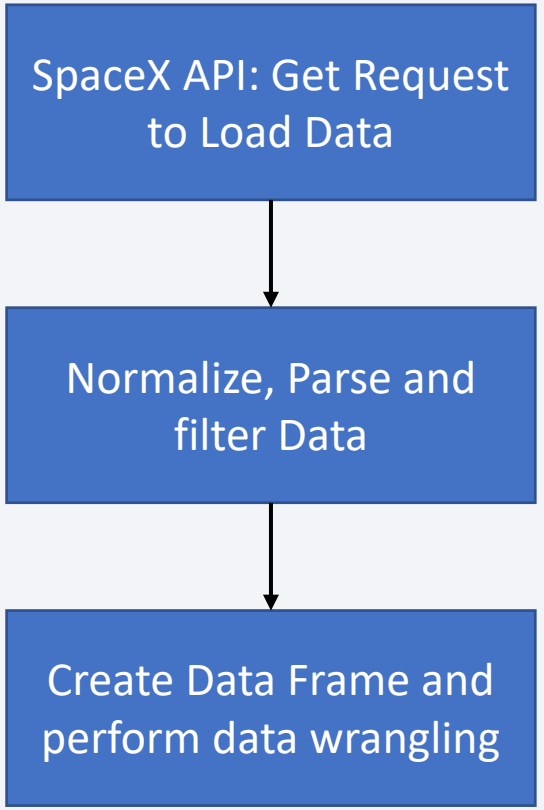
Data Collection

- Describe how data sets were collected.
  - Using Get Requests we parse the SpaceX launch data API
  - Decode the response content using JSON and turned into pandas data frame
  - Use API to get additional information about the launches using the IDs given for each launch
  - Build a launch dictionary, create a dataframe and filter Falcon 9 launch data
  - Perform data wrangling to replace null values with column mean

Data Collection – SpaceX API

- SpaceX offers API where data can be obtained
- Use get requests to load data from Space X API
- We Normalize, parse and filter Falcon 9 data
- Build Dictionary
- Create date frame
- Perform data wrangling
- The link to the notebook is:

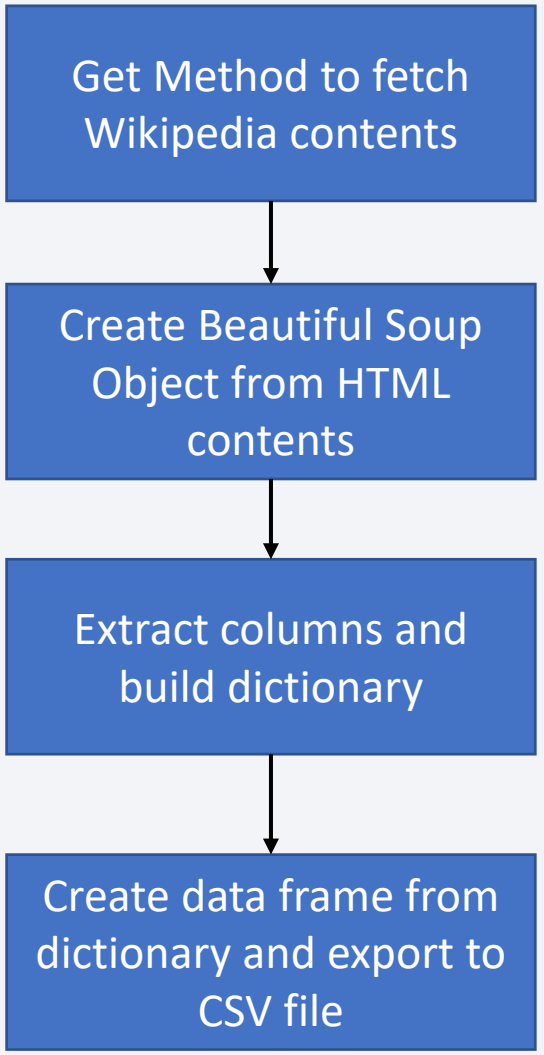
https://github.com/Stoach/Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Use get method assign SpaceX Wikipedia HTML page content to variable.
- Create Beautiful soup object
- Extract all column / variables names from HTML table headers
- Create dictionary by parsing HTML tables
- Covert dictionary to data frame and export CSV
- The link to the notebook:

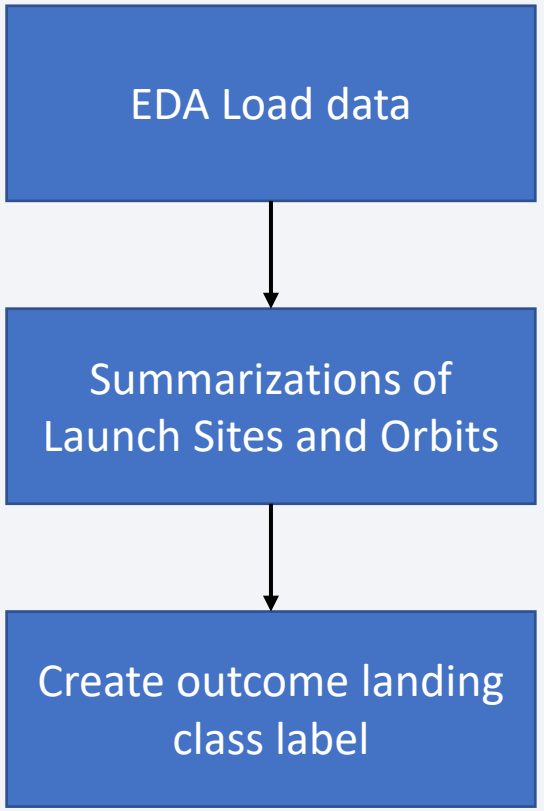
https://github.com/Stoach/Data-Science-Capstone/blob/main/jupyter-labs-webscraping%20(4).ipynb



Data Wrangling

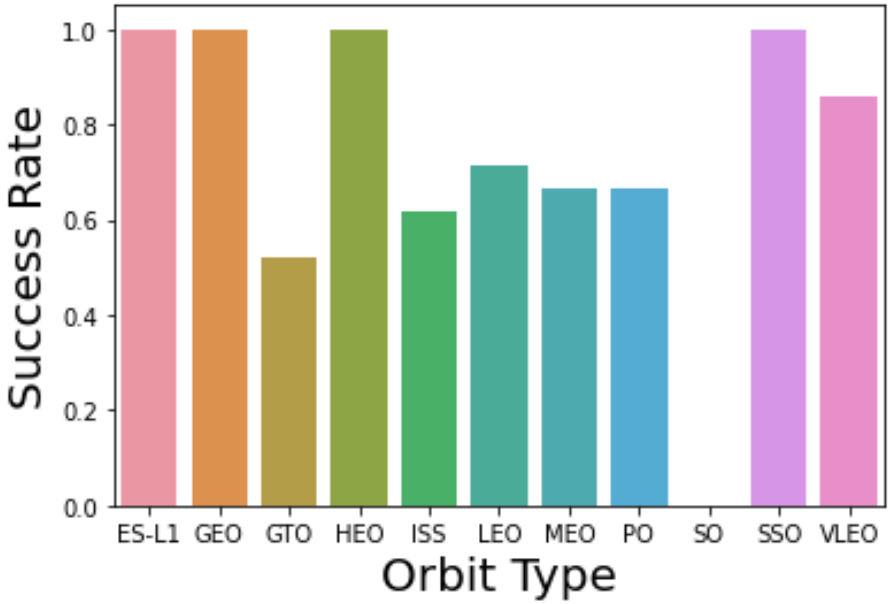
- Load Space X data set
- Determined percentage of missing values
- Reviewed data types
- Calculated number of launches at each site
- Counted number of Orbits
- Create landing outcome class label from outcome column
- Export data frame to csv.
- Link to Github notebook:

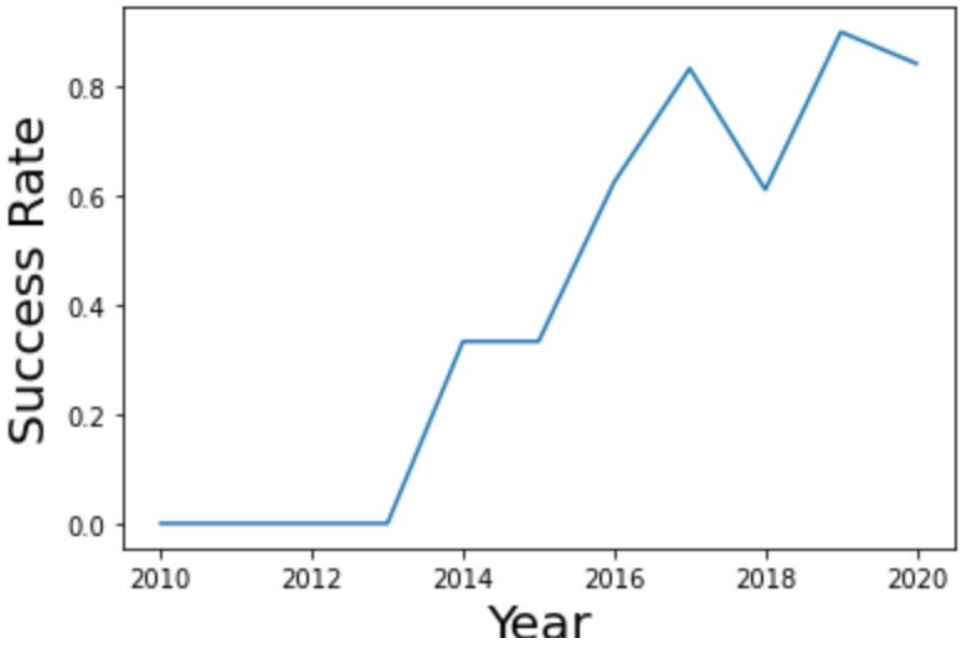
  https://github.com/Stoach/Data-Science-Capstone/blob/main/labs_jupyter_spacex_Data_wrangling.ipynb



EDA with Data Visualization

- Explored the data by visualizing the multiple relationship between flight number, payload, launch site, orbit, and success rate of orbit type and year.





- The link to the notebook is https://github.com/Stoach/Data-Science-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- From the SpaceX data set loaded into the DB2 SpaceX table
  - Selected distinct launch sites
  - Displayed five records where launch site starts with ‘CCA%’
  - Displayed the total payload mass carried by boosters launched by NASA (CRS)
  - Displayed average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was achieved.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
  - List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - 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
- Github Link:

https://github.com/Stoach/Data-Science-Capstone/blob/main/EDA%20with%20sql-coursera.ipynb

Build an Interactive Map with Folium

- Used circle markers, color labeled clusters and lines in the Folium map
- Marked Johnson Space Center and four launch sites with circles
- Assigned Green or Red label to launch outcomes. Class 1 = Green, class 0 = Red.
- Assigned color-labeled marker clusters to identify which launch sites have relatively high success rate or not.
- Calculated the distances between a launch site to its proximities. Answered several questions:
  - Are launch sites in close proximity to railways?
  - Are launch sites in close proximity to highways?
  - Are launch sites in close proximity to coastline?
  - Do launch sites keep certain distance away from cities?
- Github link:
  https://github.com/Stoach/Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Build a Plotly Dash to perform interactive visual analytics on SpaceX launch data in real-time
- We added the following plots / graphs:
  - Pie chart depicting total success launches by site
  - Scatter chart illustrating correlation between payload mass and launch success
- After visual analysis we answered the five questions:
  - Which site has the largest successful launches?
  - Which site has the highest launch success rate?
  - Which payload range(s) has the highest launch success rate?
  - Which payload range(s) has the lowest launch success rate?
- GitHub URL:

  https://github.com/Stoach/Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Loaded the data using numpy and pandas, transformed and split our data into training and testing.
- We built different machine learning models and tune different hyperparameters using GridSearchCV.
- We found the best performing classification model.
- Notebook Link:

  https://github.com/Stoach/Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

1. Set the Y variable and transform X to

```
In [ ]: Y = data["Class"].to_numpy()
        Y[0:5]
```

2. Transform X data in to z-scores using StandardScaler

```
In [ ]: # students get this
        transform = preprocessing.StandardScaler().fit(X)

In [ ]: X=transform.transform(X)

In [ ]: df = pd.DataFrame(X)
        df
```

3. Split the data into train and testing sets

```
X_train, X_test, Y_train, Y_test

In [ ]: X_train, X_test, Y_train, Y_test =  train_test_split(X,Y,test_size=0.2,random_state=2)
```

4. Start with Logistic Regression model, assign parameters, set Lr object, determine hyperparameters using GridSerchCV and determines accuracy score.

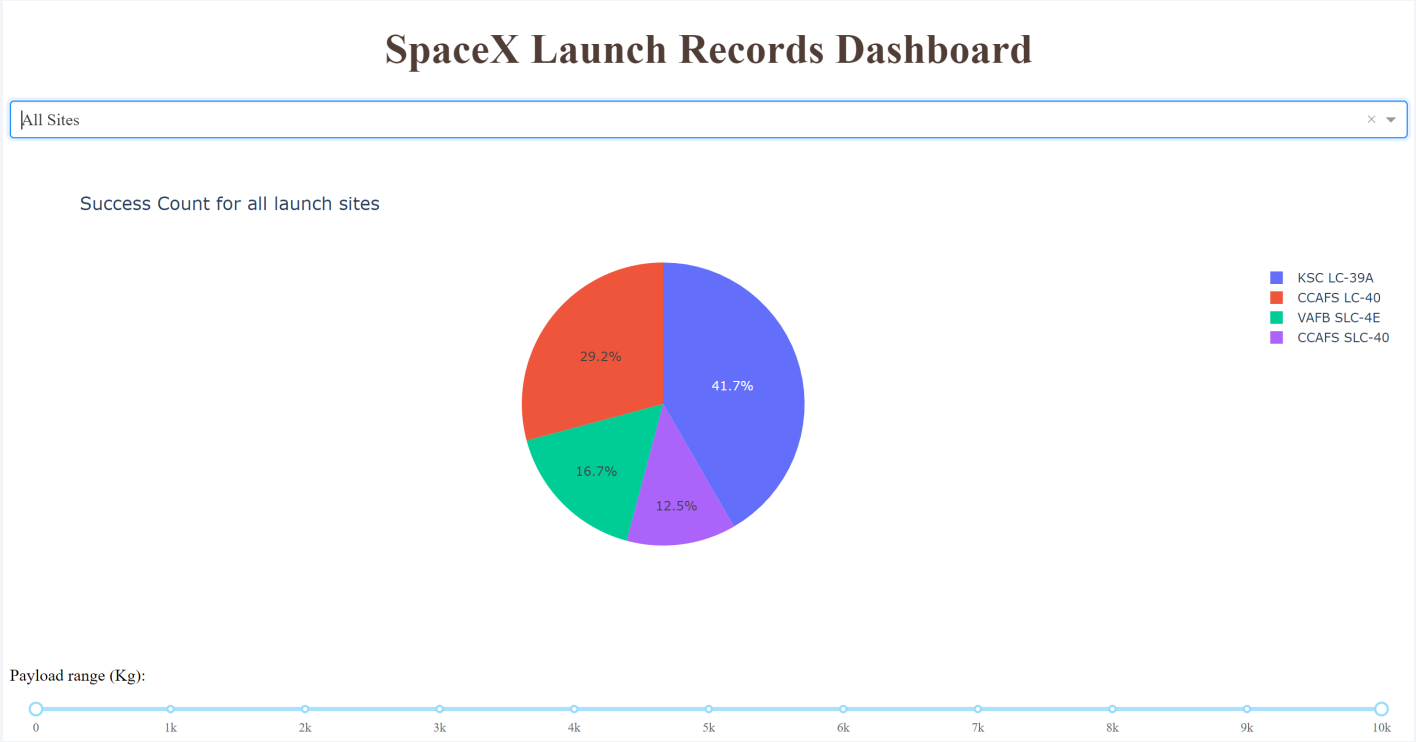
```
In [ ]: parameters ={'C':[0.01,0.1,1],
                    'penalty':['l2'],
                    'solver':['lbfgs']}

In [ ]: parameters

Out[ ]: {'C': [0.01, 0.1, 1], 'penalty': ['l2'], 'solver': ['lbfgs']}

In [ ]: parameters ={"C":[0.01,0.1,1],'penalty':['l2'], 'solver':['lbfgs']}# l1 lasso l2 ridge
        lr=LogisticRegression()

In [ ]: logreg_cv = GridSearchCV(lr, parameters, cv=10)

In [ ]: logreg_cv.fit(X_train, Y_train)

In [ ]: print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
        print("accuracy :",logreg_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

5. Repeat for SVM, Decision Tree and K Nearest neighbors

Results

- Exploratory data analysis results
  - Success rate kept increasing from 2013 until 2020.
- Predictive analysis results
  - Evaluated the following Algorithms:
    - Logistic Regression
    - SVM
    - Decision trees
    - K Nearest Neighbors
  - All algorithms scored the same with 0.8333333

- Interactive analytics demo in screenshots

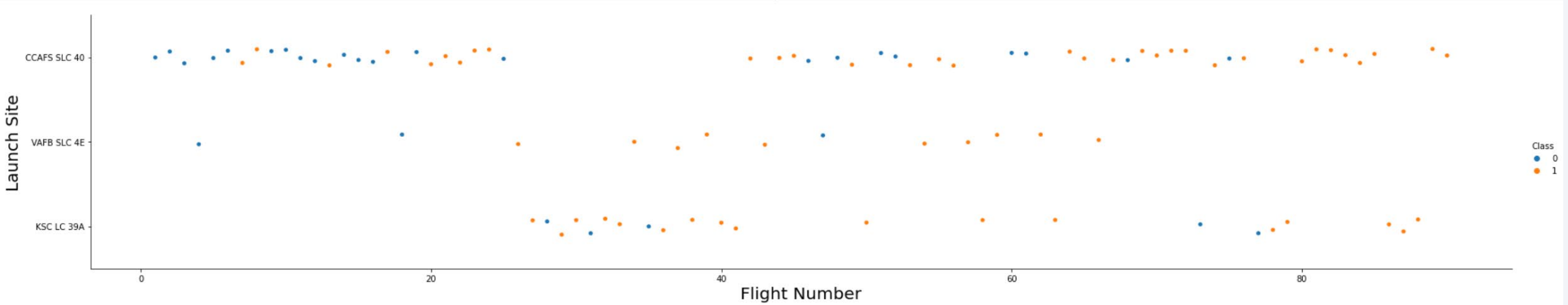


Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

Show a scatter plot of Flight Number vs. Launch Site

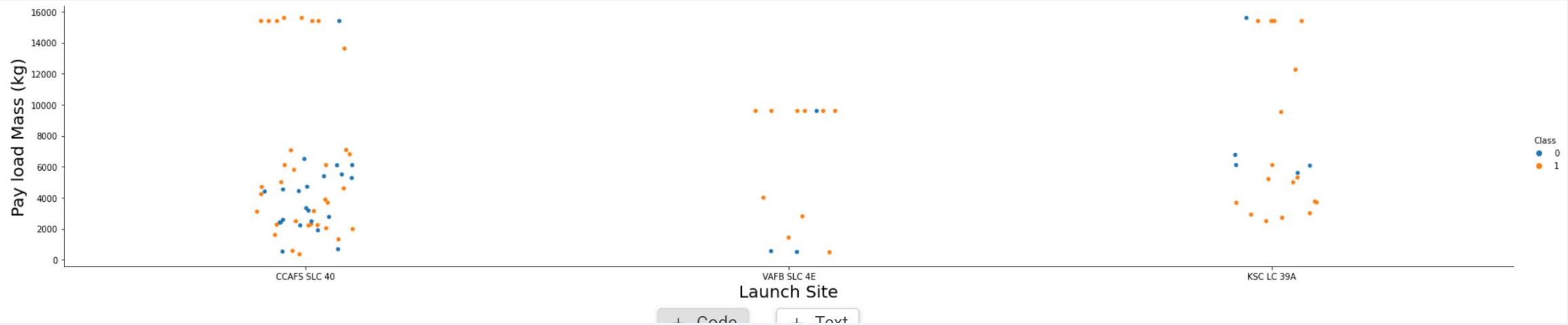


Explanation:

- CCAFS SLC 40 has highest numbers of launches. Class 0 and 1 occurrences are close
- VAFB SCL 4E has least amount of launches. Class 1 occurrences are significantly greater than class 0.
- Class 1 occurrences are greater for later flights are all three launch sites. They are getting better at it.

Payload vs. Launch Site

Show a scatter plot of Payload vs. Launch Site

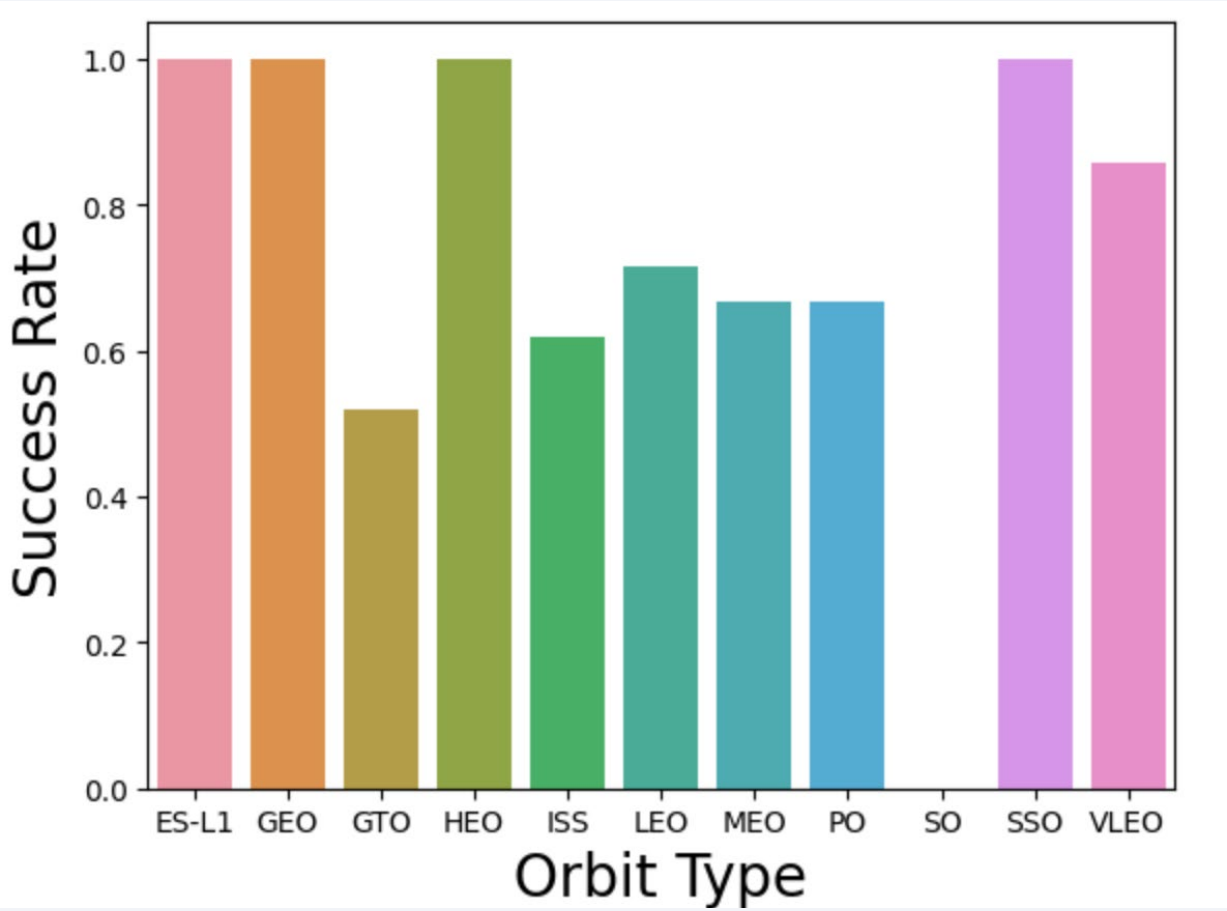


Explanations:

- Note for VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).
- CCAFS SLC 40 has a lot of launches less than 8000 and higher occurrence of class 1 for heavy payload (greater than 12000).
- KSC LC 30A used for light, middle and heavy payloads. Overall high occurrence of class 1.

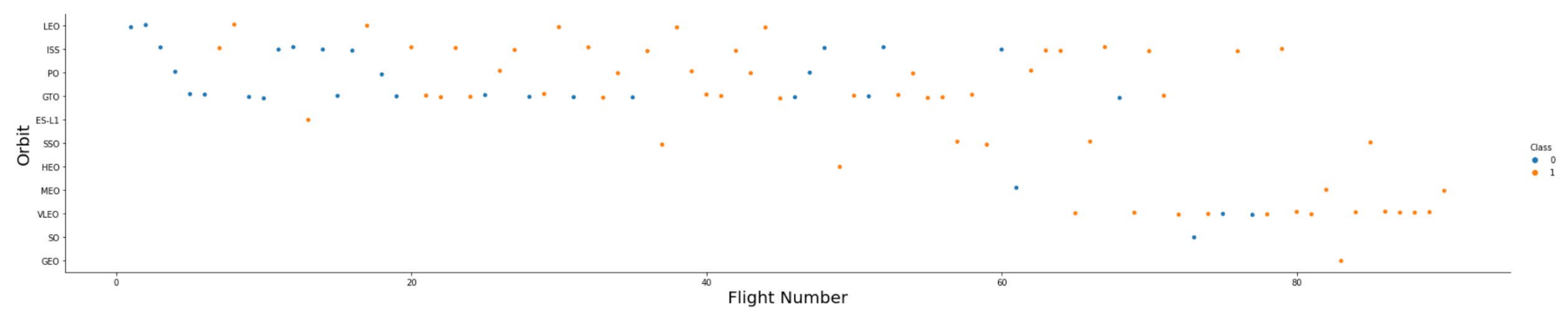
Success Rate vs. Orbit Type

- The plot illustrate that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- GTO had the lowest success rate.



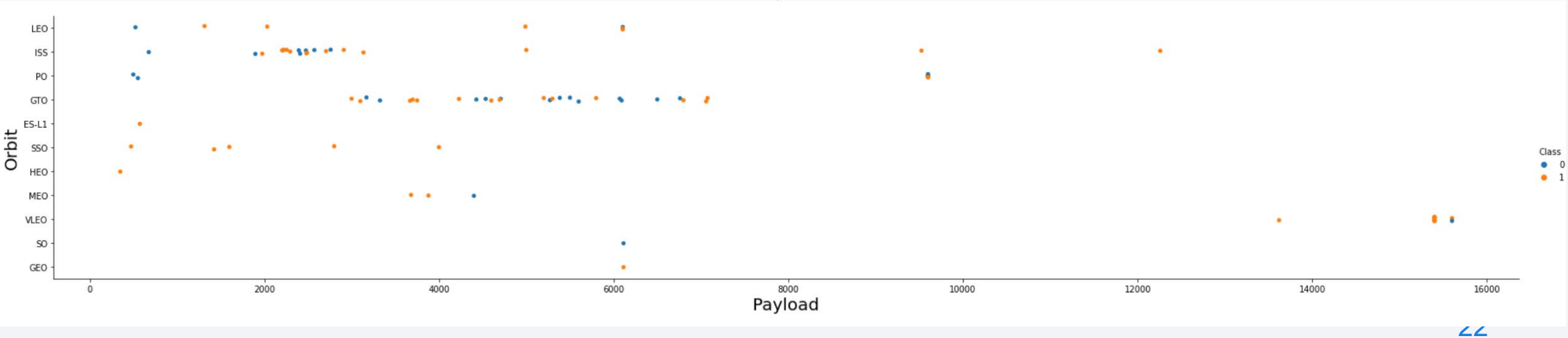
Flight Number vs. Orbit Type

- You should 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.
- First 80 flights primarily targeted orbits LEO, ISS, PO, GTO, ES-L1



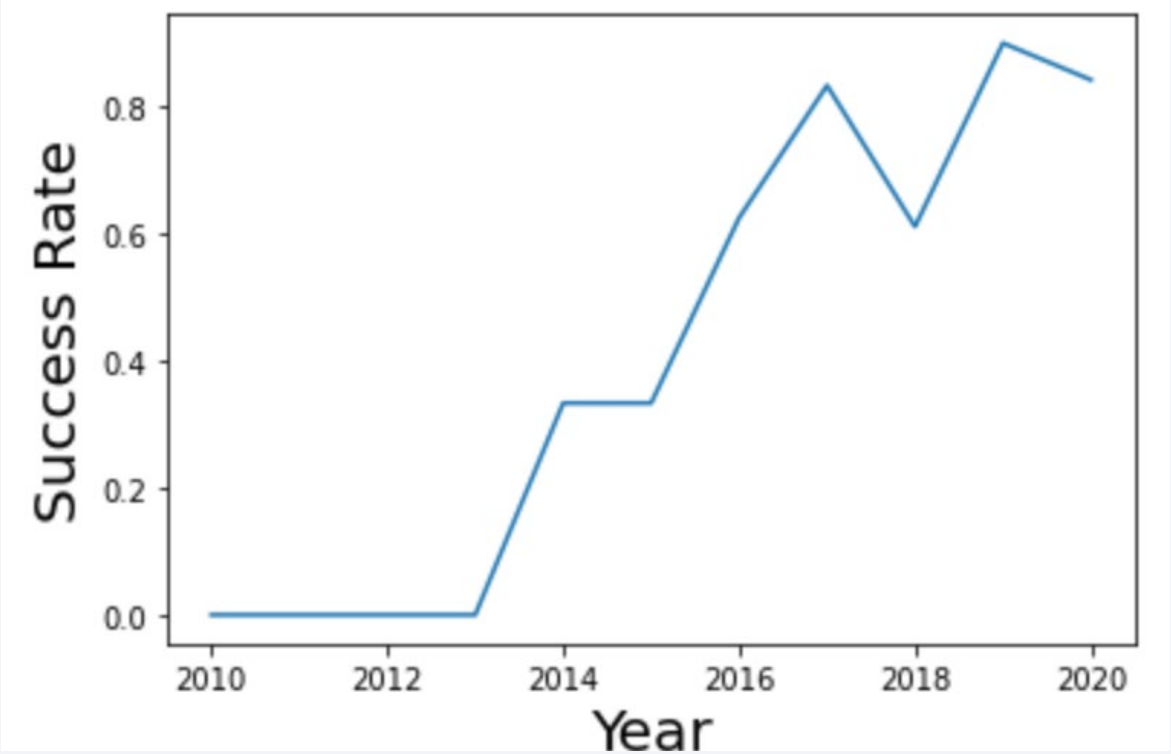
Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for 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.
- With lighter payloads the successful landing or positive landing rate are more for SSO



Launch Success Yearly Trend

- The line plot illustrates the success rate since 2013 kept increasing till 2020
- There was a two point success rate dip between 2017 and 2018



All Launch Site Names

- The key distinct displays only unique launch sites in the launch_site column in the SPACEXTBL

In [4]:
```
%sql select distinct Launch_Site from SPACEXTBL
```

Out[4]:

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

Launch Site Names Begin with 'CCA'

- Display 5 records where launch sites begin with the string 'CCA‘
- Note the key word like ‘CAA%’. The % acts as a wild card.

In [5]:
```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5
```

```
 * ibm_db_sa://gtc27297:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
Done.
```

Out[5]:

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

- Utilize the Sum() function on the payload_mass_kg_ column. It displays the total payload mass for all row where the customer = ‘NASA (CRS)’

```
In [7]: %sql select sum(payload_mass__kg_) from SPACEXTBL WHERE customer = 'NASA (CRS)'
```

```
Out[7]:
 1
---
45596
```

Average Payload Mass by F9 v1.1

- Utilize the avg() function to calculate the payload mass for booster version F9 v1.1

```
In [8]: %sql select avg(payload_mass__kg_) from SPACEXTBL WHERE booster_version = 'F9 v1.1'
```

```
 * sqlite:///my_data1.db
Done.
Out[8]:
avg(payload_mass__kg_)
2928.0
```

First Successful Ground Landing Date

- By using the MIN() function on the data column, we can fetch the earliest or minimum date of a successful landing outcome on a ground. Note the criteria of the where clause.

In [9]:
```
%sql select min(DATE) from SPACEXTBL WHERE landing__outcome = 'Success (ground pad)'
```

Out[9]:

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- We select the booster_version column from the SPACEXTBL with two where clause criteria. We use the AND clause because both condition must be true.

In [10]:
```
%sql select booster_version from SPACEXTBL where landing__outcome = 'Success (drone ship)'\
and payload_mass__kg_ between 4000 and 6000
```

Out[10]:

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

Total Number of Successful and Failure Mission Outcomes

- Use the count() function and group by mission outcome to count the number of each mission outcome

In [11]:
```
%sql select mission_outcome, count(mission_outcome) from SPACEXTBL GROUP BY mission_outcome
```

Out[11]:

mission_outcome	2
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

In [12]:

```
%sql select booster_version, payload_mass__kg_ from SPACEXTBL\
where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTBL)
```

- By using a sub-query to fetch the maximum payload we can query the booster versions and payload from the SPACEXTBL where the payload equals the maximum payload.

Out[12]:

booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

- Select the booster version, launch site and landing outcome where the landing out equals = ‘Failure (Drone Ship)’ and the date between January 1st and December 31 2005.

In [13]:
```
%sql select booster_version, launch_site from SPACEXTBL where landing__outcome = 'Failure (drone ship)' and year(DATE) = 2015
```

Out[13]:

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

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

- By grouping the landing outcomes and applying the count function, the query returns the total number of landing outcomes in the table. Ordering by the count in descending order we can rank from highest to lowest landing outcome counts.
- The between clause in the where criteria is used filter between the desired dates.

In [14]:
```
%sql select count(landing__outcome), landing__outcome from SPACEXTBL \
where DATE between '2010-06-04' and '2017-03-20' group by landing__outcome\
order by count(landing__outcome) desc
```

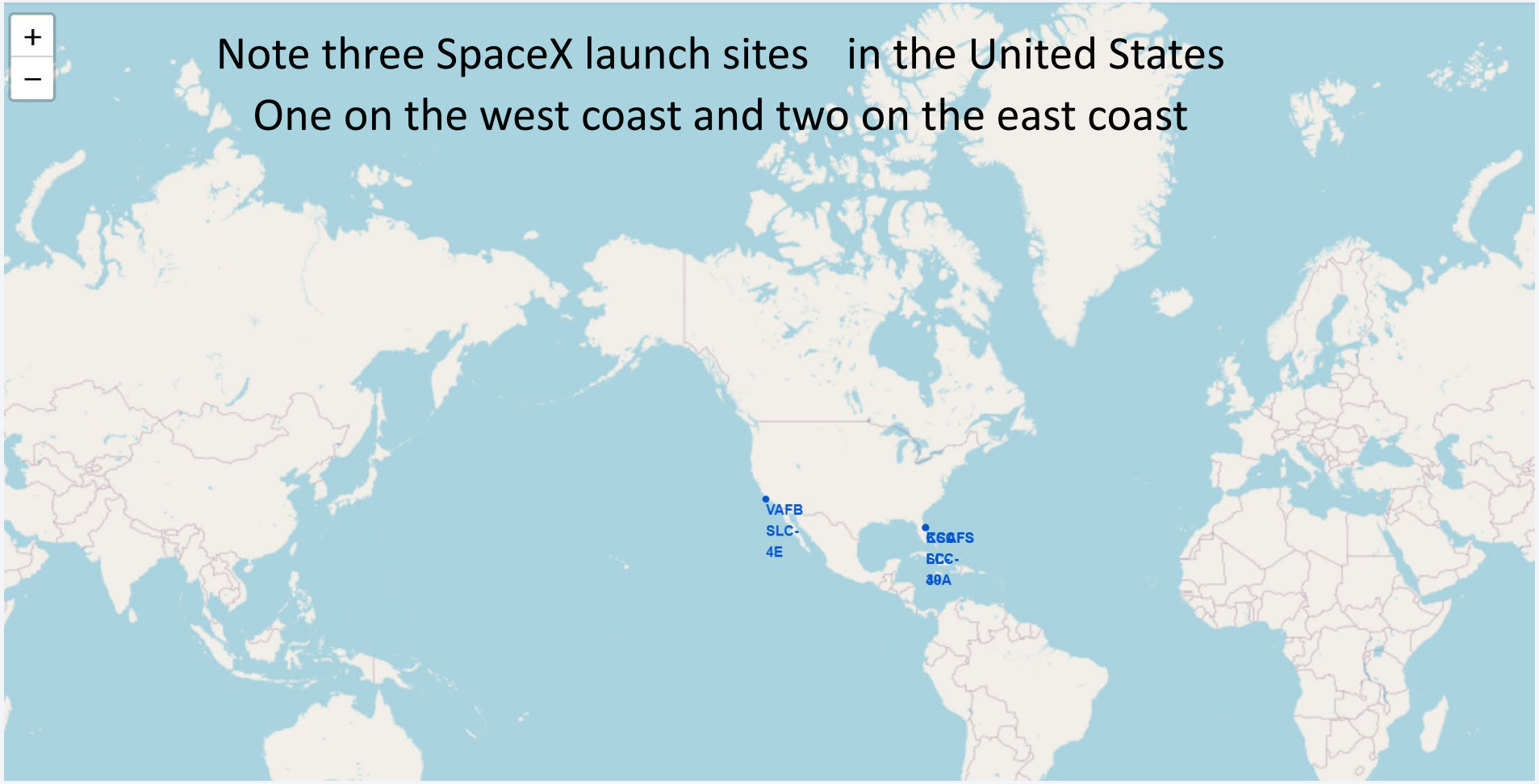
Out[14]:

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

Section 3

Launch Sites Proximities Analysis

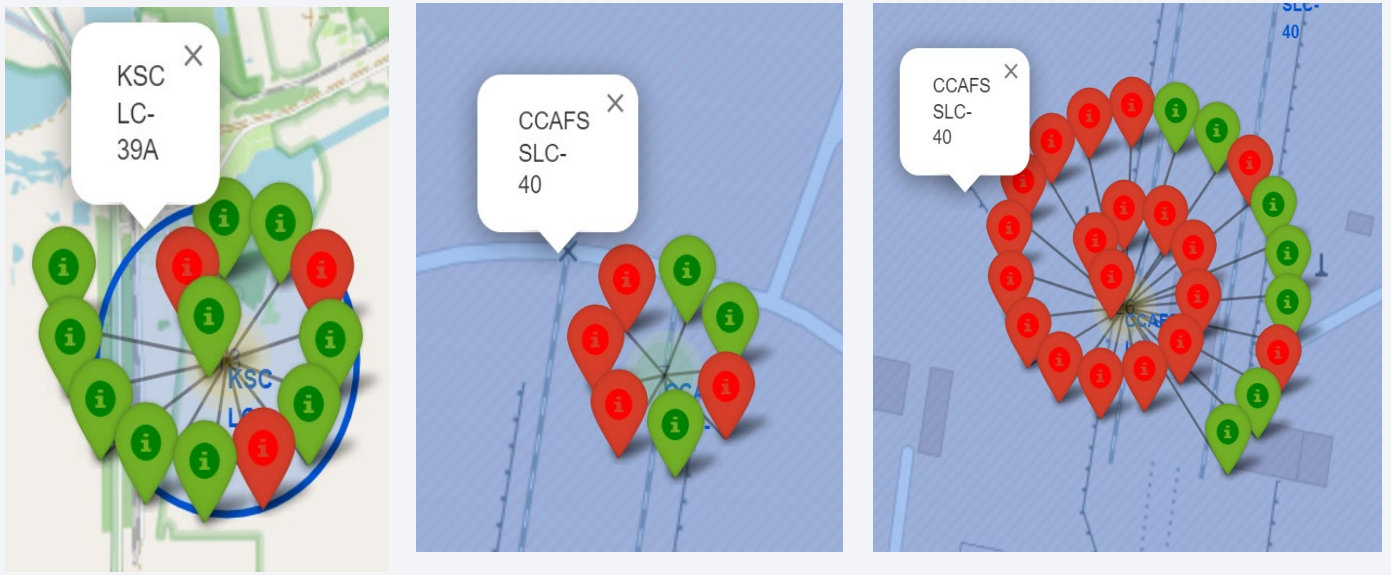
All SpaceX Launch Site Global Map Markers



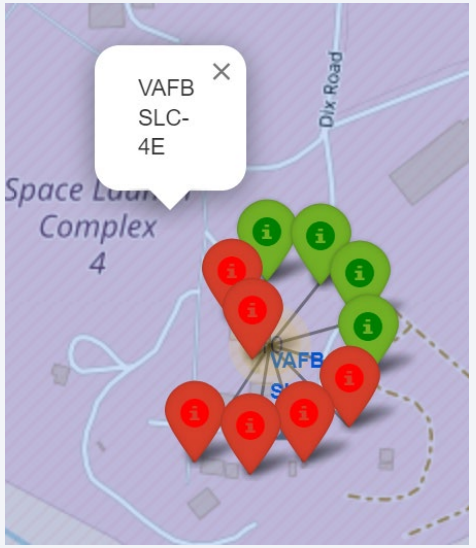
Markers showing launch sites with color labels

Note that a launch only happens in one of the four launch sites, which means many launch records will have the exact same coordinate. Marker clusters can be a good way to simplify a map containing many markers having the same coordinate.

Three Florida Launch Sites



One California Launch Sites

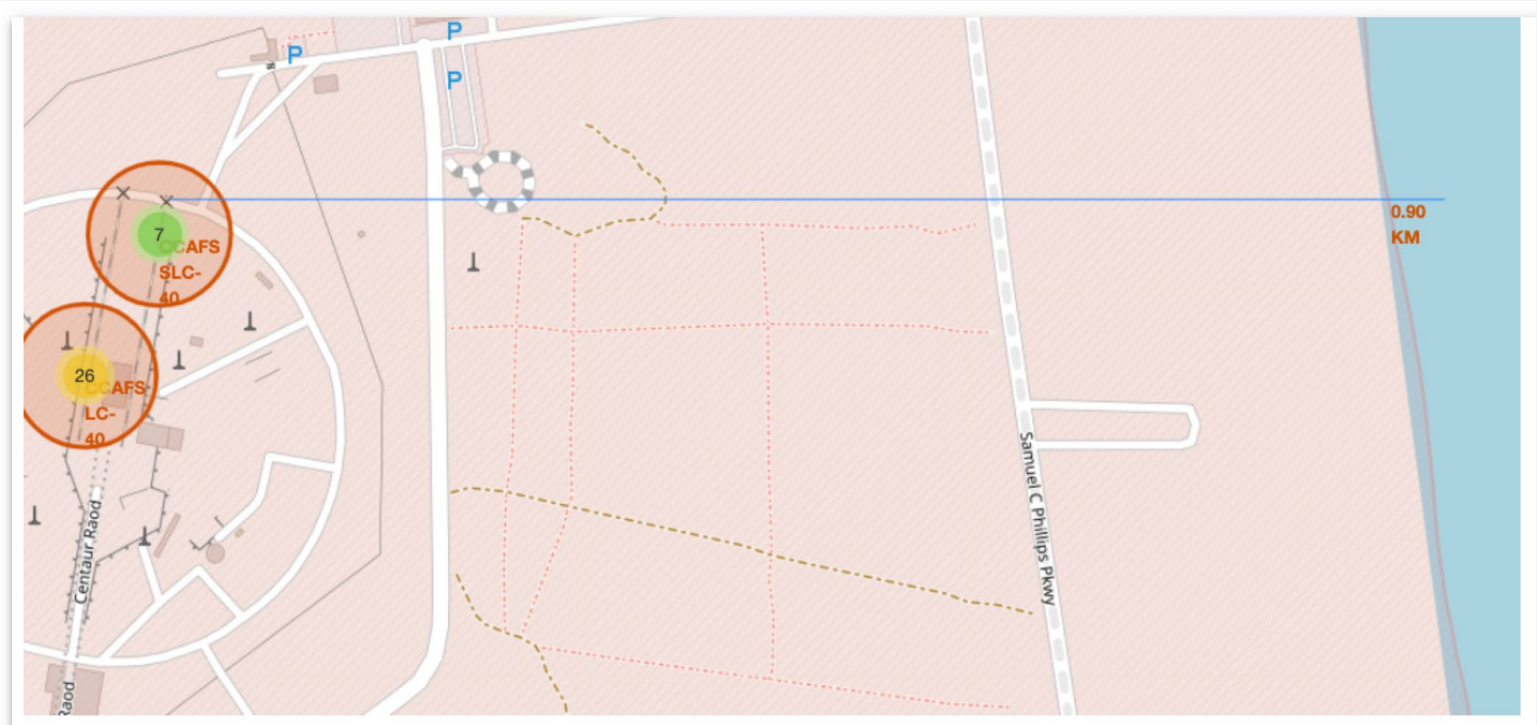


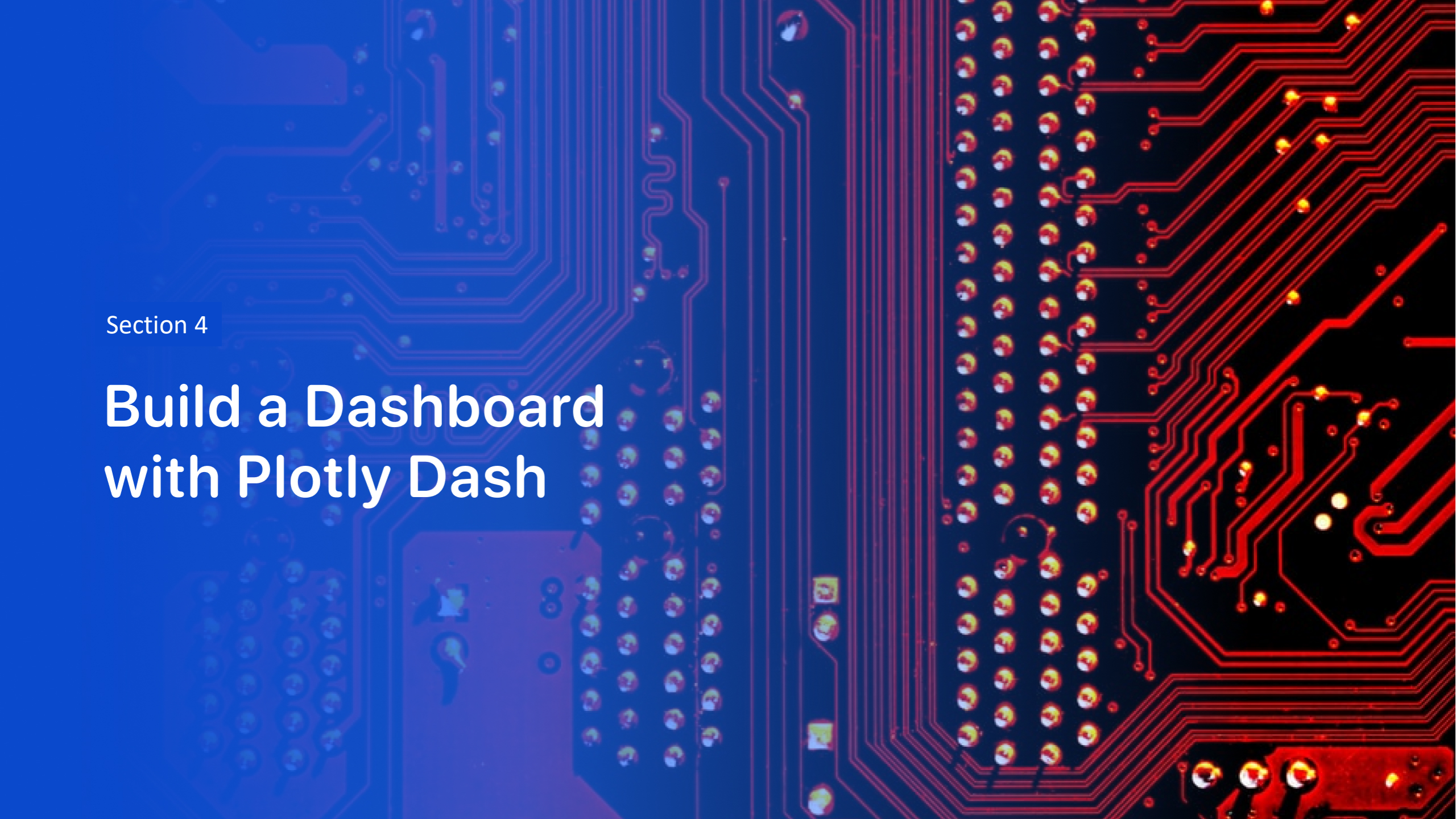
Green Markers illustrate Successful Launch

Red Markers illustrate Failed Launch

Launch Site Distance to the Coastline

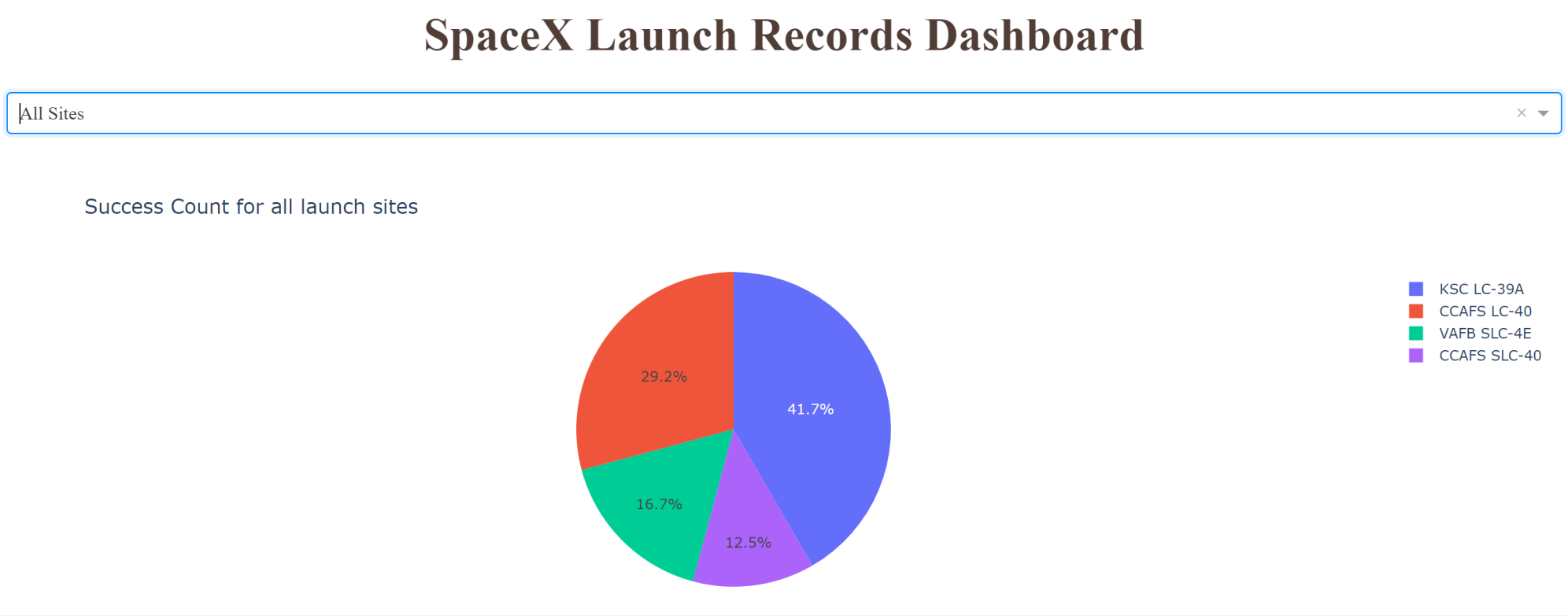
To illustrate the distance from launch site CCAFS SLC-40 to the coastline to draw a polyline between the two. We can see the distance is 0.90 KM.




Section 4
Build a Dashboard with Plotly Dash

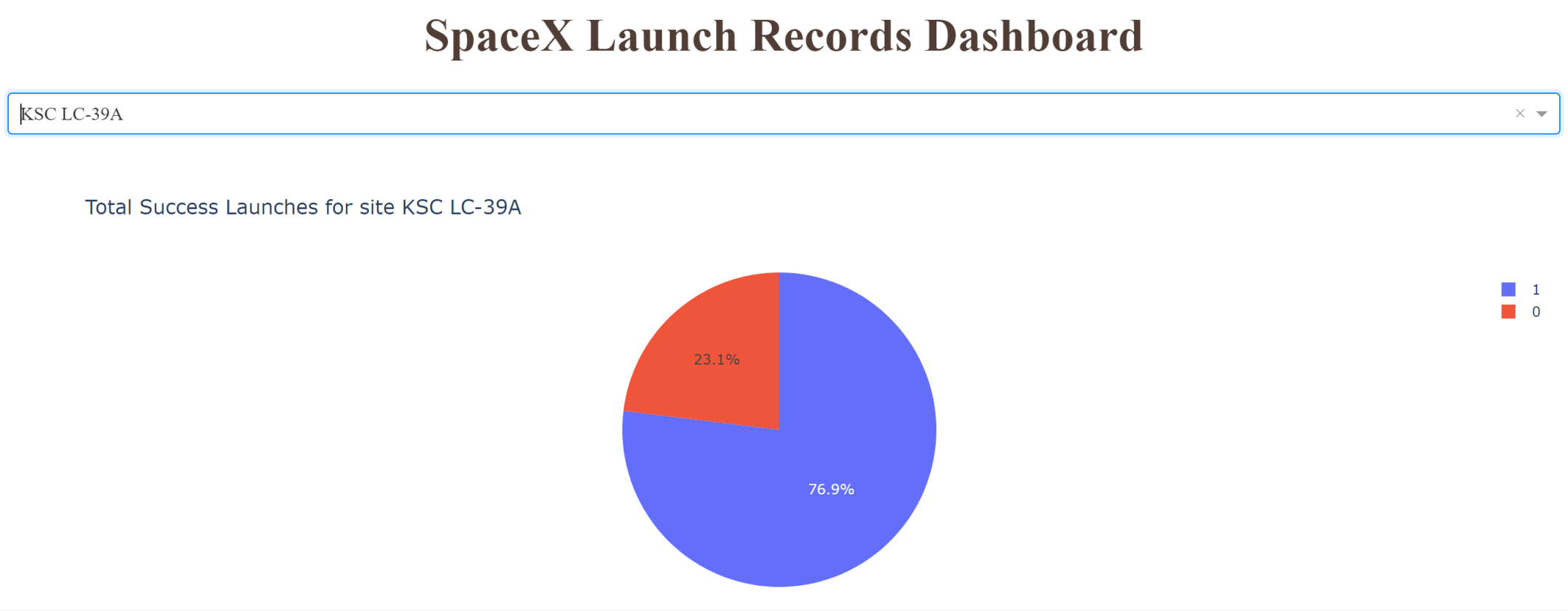
Pie Chart of SpaceX Launch Site Success Count Percentages

- All sites are selected from drop down list box
- KSC LC-39A has the highest success percentage



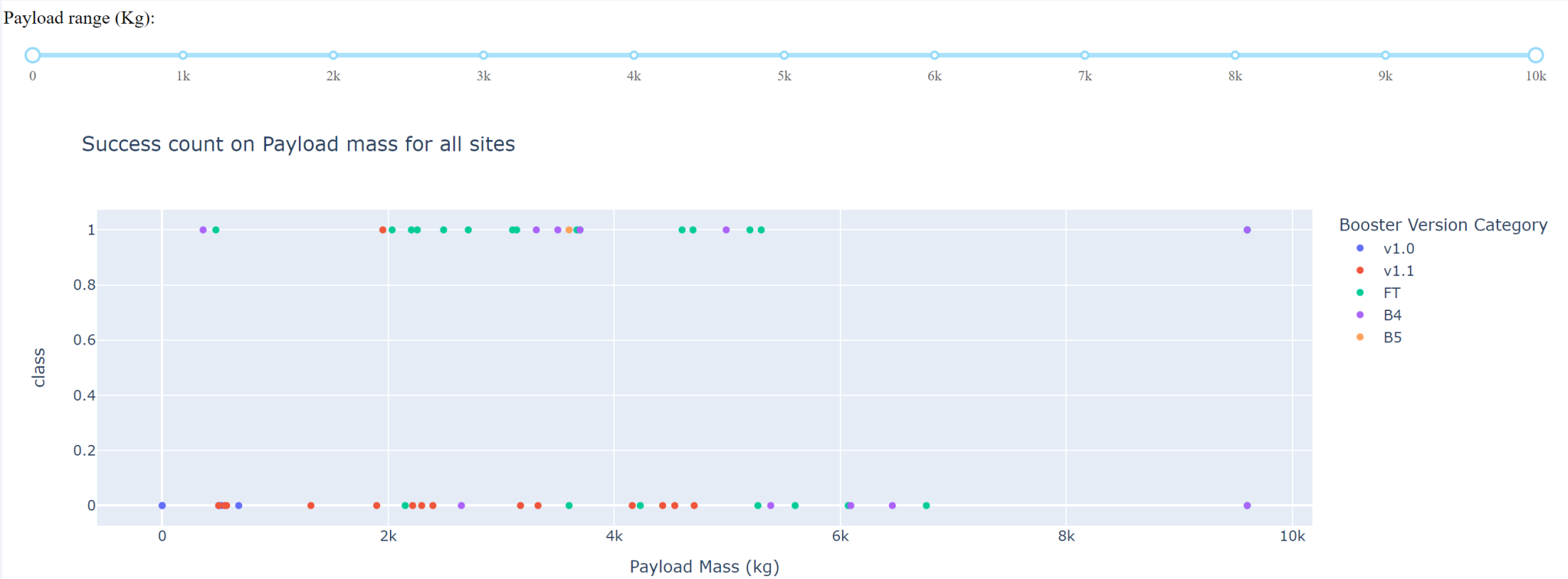
Pie Chart of Launch Site with Highest Success Rate

- KSC LC-39A has a success rate of 79.9% and failure rate of 23.1%.



Scatter Plot of Payload vs. Launch Outcome for All Sites

- Note payload sider scale
- We can see a higher success rate for lighter payloads compared to heaver payloads

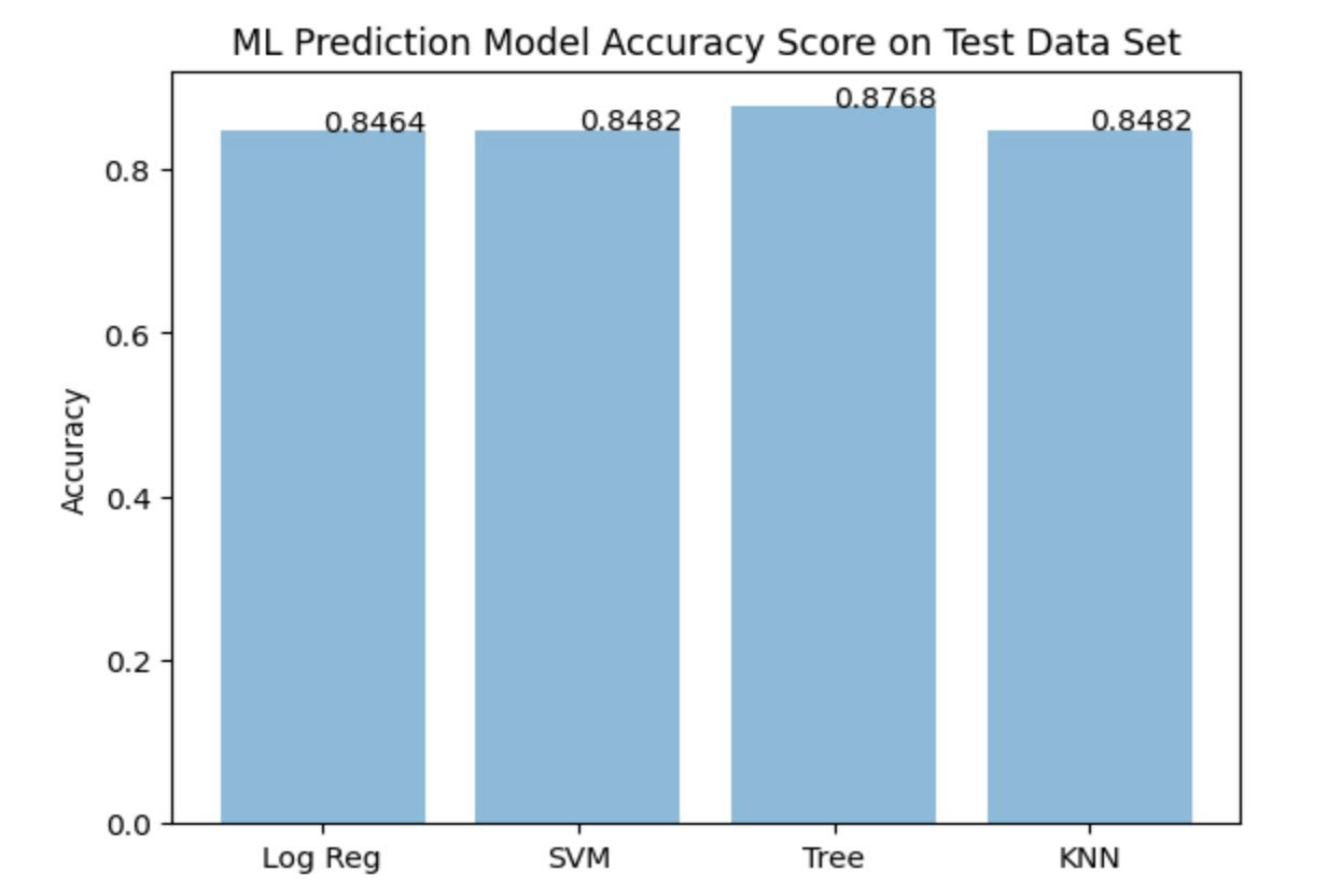


Section 5

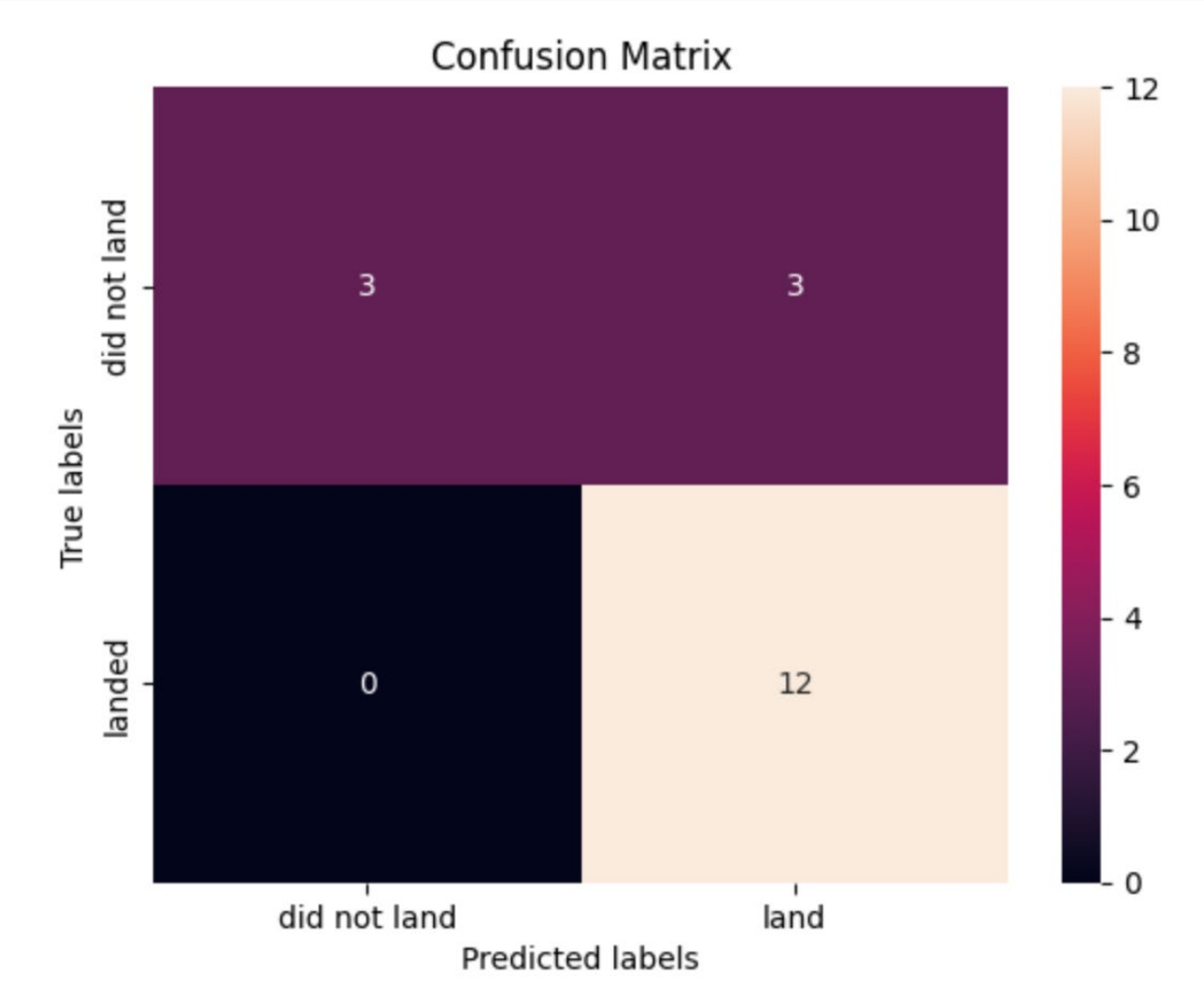
Predictive Analysis (Classification)

Classification Accuracy

From the bar char below, the Decision Tree Model has the highest accuracy at 0.87.



Confusion Matrix with Explanation



A confusion matrix summarizing the performance of the Decision Tree algorithm.

The Matrix illustrates:

- 3 True Positives (Upper Left Quadrant)
- 3 False Positive (Upper Right Quadrant)
- 0 False Negative (Lower Left Quadrant)
- 12 True Negatives (Lower Right Quadrant)
- The concern is the 3 False Positives. Unsuccessful lands predicted a successful.

Conclusions

We can conclude that:

- CCAFS SLC-40 has highest numbers of launches
- Heavier Payloads (greater than 8000) have a high success rate at all Launch Sites
- Launch success rate increased between 2013 and 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate
- KSC LC-39A had the most successful launches of any sites.
- CCAFS SLC-40 has the most unsuccessful launches of any sites.
- The Decision tree classifier is the best machine learning algorithm for this task.

Thank you!