Winning Space Race with Data Science

Gregory Prophete
February 12, 2023

Outline

Executive Summary

- Summary of methodologies
  - Data Collection through API (Spacex API) and Webscraping (Wikipedia)
  - Data Wrangling
  - Exploratory Data Analysis Using SQL, Pandas and MatPlotlib
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive Dashboard (Screenshots)
  - Predictive Analysis result

Introduction

- Project background and context

The Falcon 9 rocket launch service is marketed by SpaceX on its website at a price tag of $62 million. Comparable services from other launch providers start at a much higher $165 million. A significant factor in SpaceX's lower cost is their capability to recover and reuse the rocket's first stage. Understanding whether this stage will successfully land is key to predicting the total cost of a launch. This data could prove valuable in a competitive scenario where another company wishes to outbid SpaceX. The project's objective is to develop a machine learning system that can accurately forecast the successful landing of the rocket's first stage.

- Problems you want to find answers

  -The differents factors which determine if the rocket will land successfully?

  -The interaction amongst various features that determine the success rate of a successful landing.

  -What prerequisites in terms of operational conditions are essential for a successful rocket landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - Data was collected using SpaceX API and web scraping from Wikipedia's website
- Perform data wrangling
    - One Hot encoding was applied to categorical 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

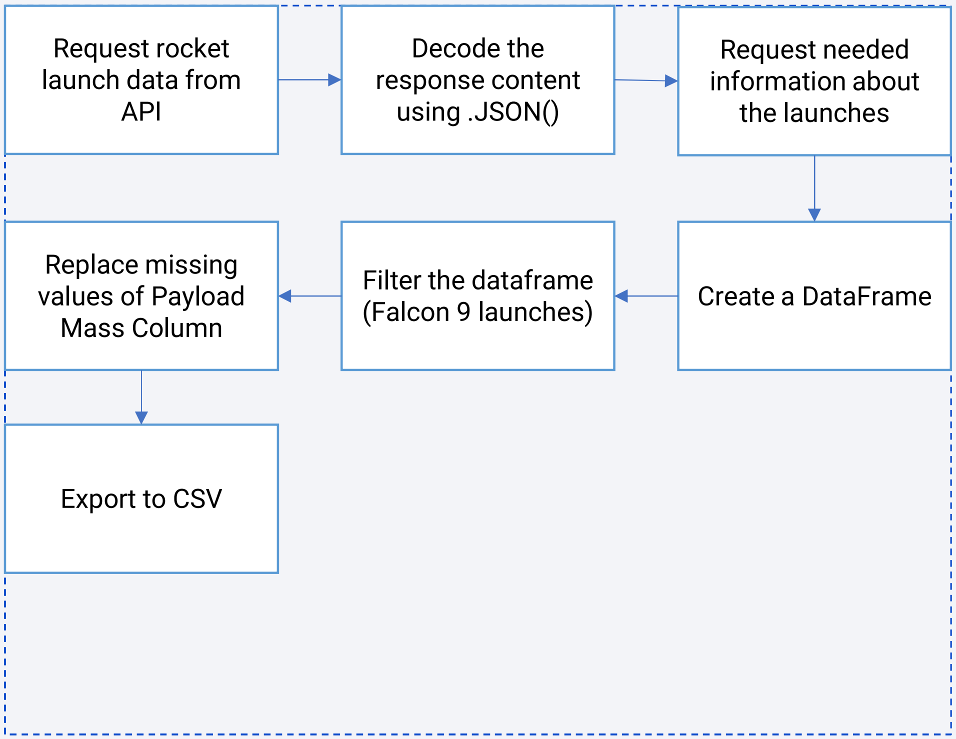
Data Collection

- Describe how data sets were collected.
  - Resquest and parse the Spacex launch Data using GET Request
  - Filter the Dataframe to only include Falcon 9 launches
  - We cleaned the data and checked for missing Values
  - Performed Web Scraping from Wikipedia with BeautifulSoup
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- Data was collected from the SpaceX API, then cleaned and formatted as necessary

- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

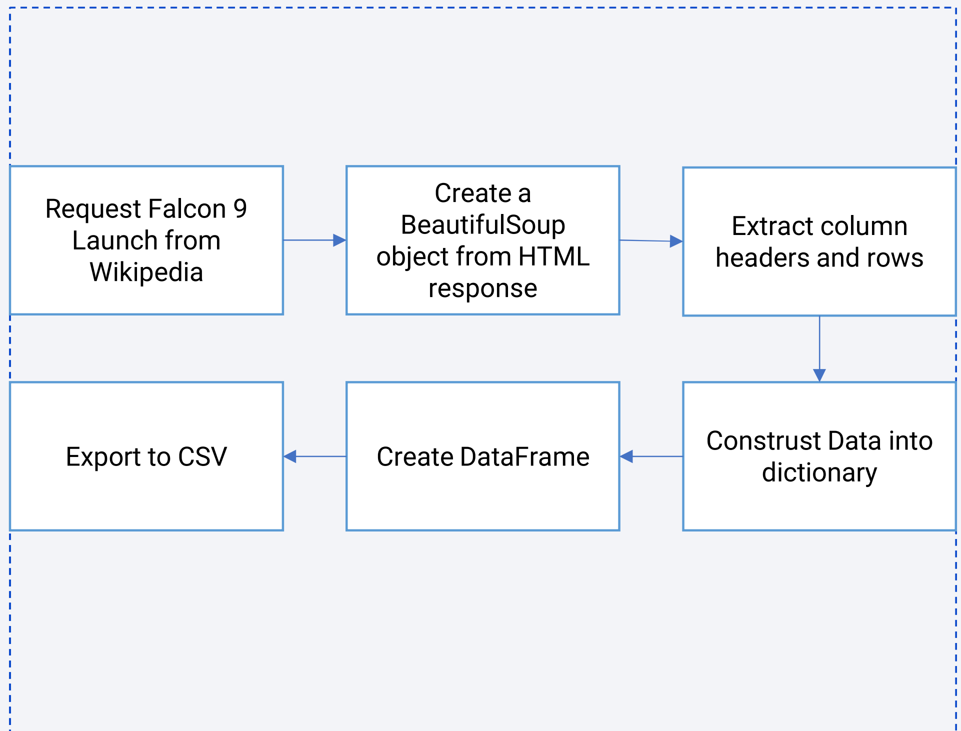


LINK: SPACEX API NOTEBOOK

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose



LINK: WEBSCRAPING NOTEBOOK

Data Wrangling

- Describe how data were processed

  Performed EDA (Exploratory Data Analysis)

  Determined Training Labels

  Calculated the number of launches on each site

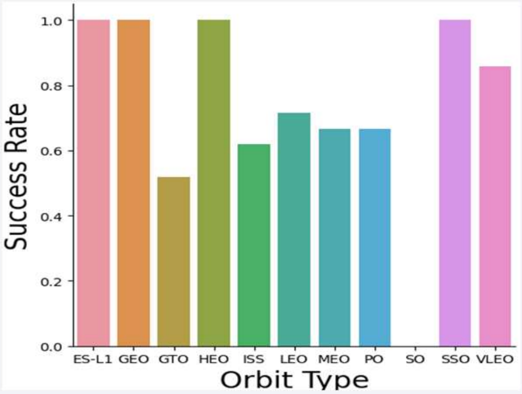
  Calculated the number of occurrence of mission outcom per orbit type

  Created landing outcome label from outcome label

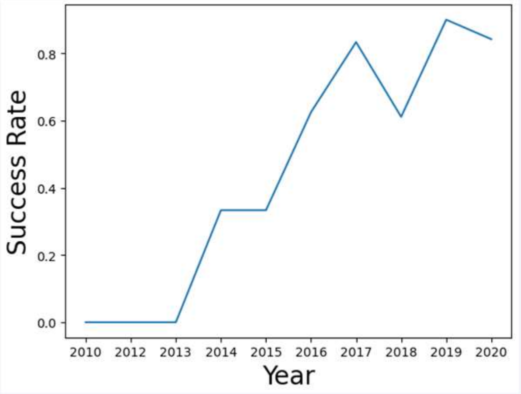
- Link : DATA WRANGLING

EDA with Data Visualization

Check if there are any relationship between success rate and orbit type



To see the evolution of the success Rate by year



LINK: [EDA WITH DATA VISUALIZATION]

EDA with SQL

- Display the names of the unique launche sites in the space mission
- Display 5 records where launches sites begin with the string “CCA”
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display 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.
- 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 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

Link : [EDA WITH SQL]

Build an Interactive Map with Folium

- Mark all launch sites, and add map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- Assign the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0 for failure, and 1 for success.
- Using the color-labeled marker clusters, we identify which launch sites have relatively high success rate.
- Calculate the distances between a launch site to its proximities and answer some question for instance:
  - Are launch sites near railways, highways and coastlines.
  - Do launch sites keep certain distance away from cities.

Link : [Interactive Map Folium]

Build a Dashboard with Plotly Dash

- Build an interactive dashboard with Plotly dash
- Plot pie charts showing the total launches by a certain sites
- Plot scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.

Link : App

Predictive Analysis (Classification)

- Loaded the data
- Created a numpy array from the column Class
- Standardize the data
- Split the data into training and testing (Train = 80%, Test = 20%)
- Built different ML models
- Tuned different parameters (GridSearchCV)
- Calculated the accuracy and confusion matrix of each model used.

Link: [Predictive Analysis]()

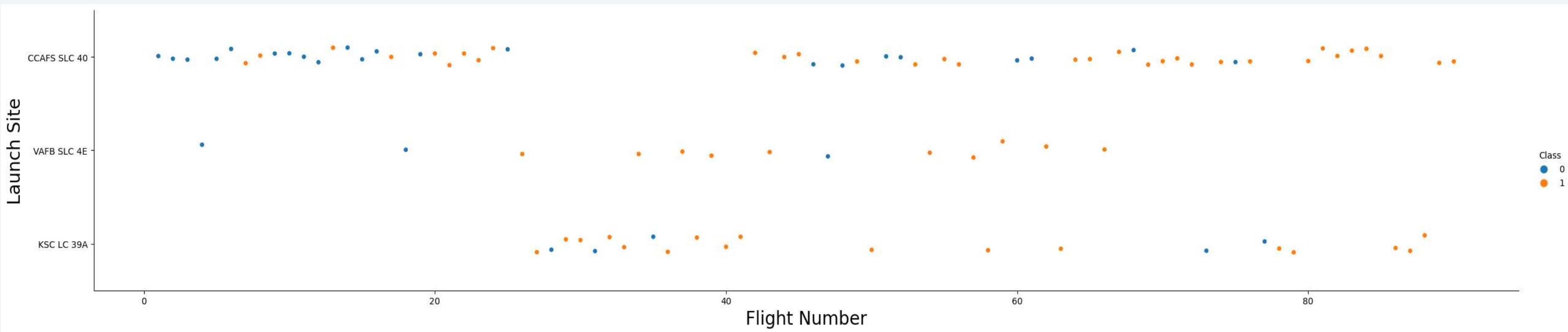
Results

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

Section 2

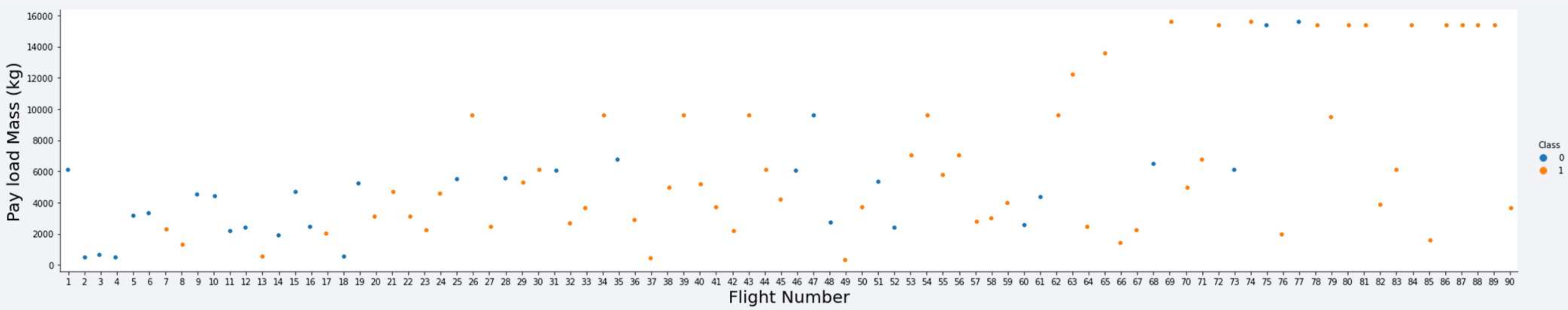
Insights drawn from EDA

Flight Number vs. Launch Site



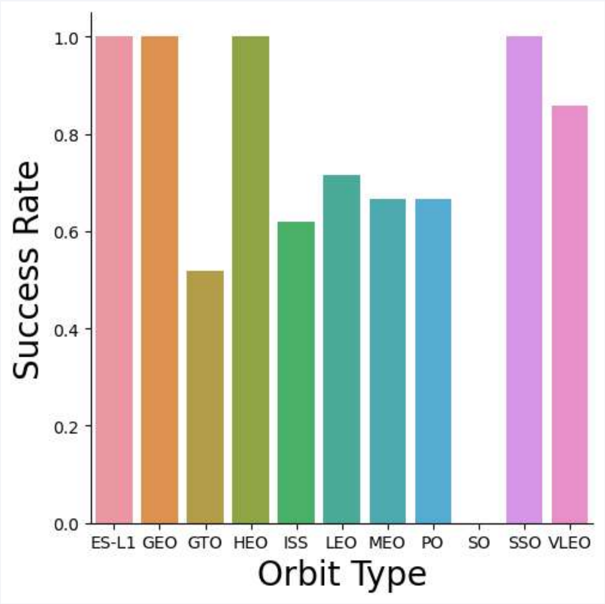
The larger the amount of flight at a launch site, the higher the number of successful launches

Payload vs. Launch Site



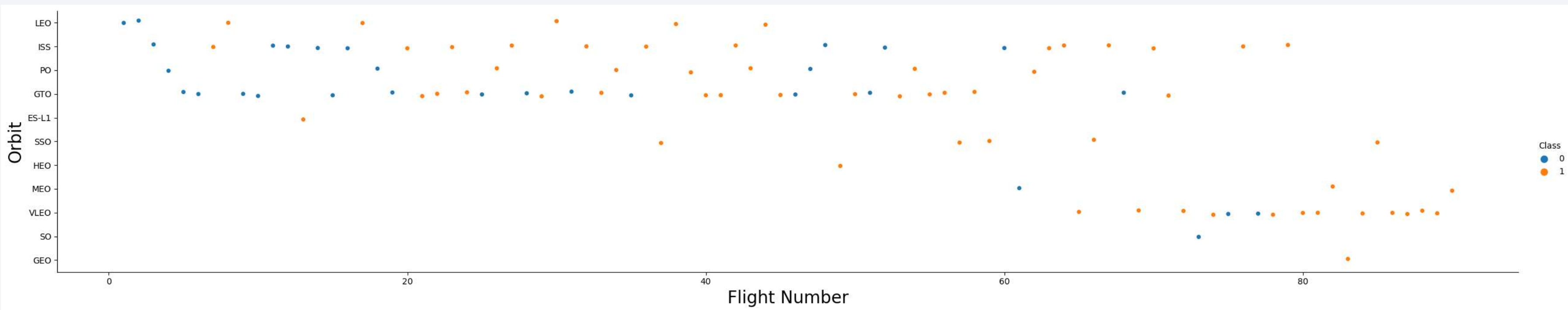
- We can see that the early flights have mostly failed (0-20)
- the VAFB SLC 4E and KSC LC 39A success rates are around 77%, which is higher than CCASS SLC 40 (60%) which has about a 61% off all launches
- We can see that there's some improvements that has been made at each failure

Success Rate vs. Orbit Type



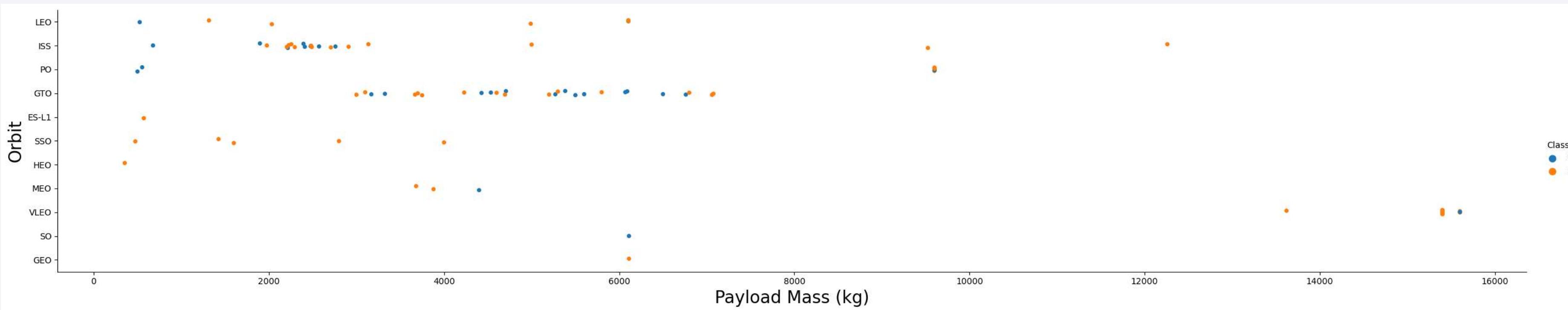
- Orbits with 100% success rate are: ES-L1, GEO, HEO, SSO
- Orbits which success rate are between 51% and 85%: GTO, ISS, LEO, MEO, PO

Flight Number vs. Orbit Type



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.

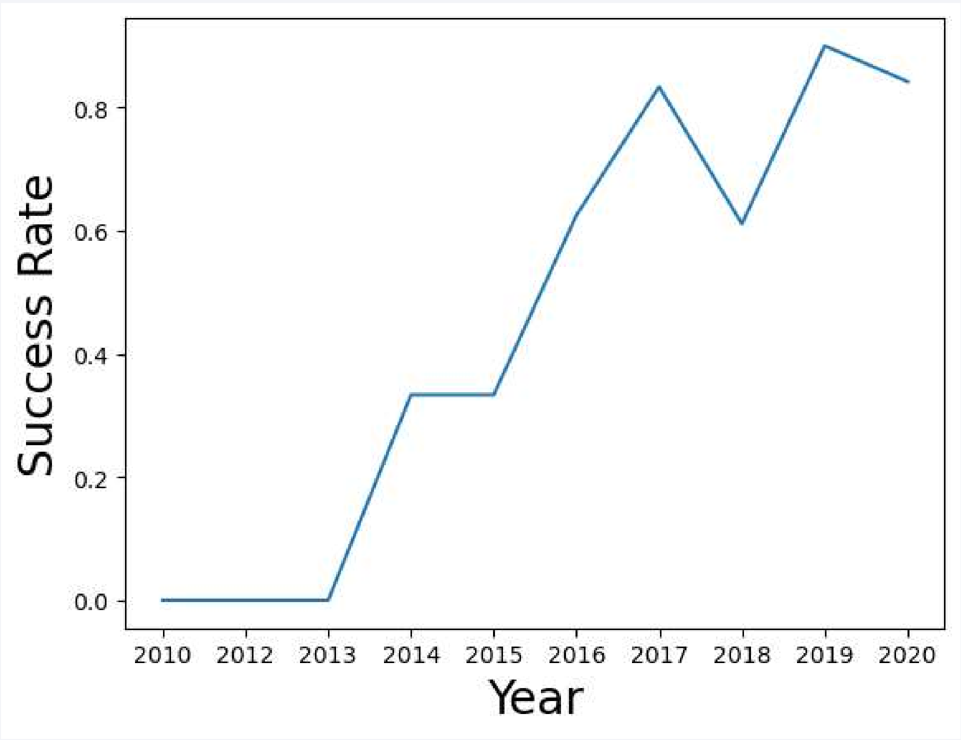
Payload vs. Orbit Type



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



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

All Launch Site Names

```
💡 Click here to ask Blackbox to help you code faster
%sql select DISTINCT LAUNCH_SITE from SCHEMA1.SPACEXTABLE;
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

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

Launch Site Names Begin with 'CCA'

💡 Click here to ask Blackbox to help you code faster

```
%sql SELECT * FROM SCHEMA1.SPACEXTABLE WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
* ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

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

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

💡 Click here to ask Blackbox to help you code faster

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SCHEMA1.SPACEXTABLE WHERE CUSTOMER LIKE 'NASA%'
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

total_payload
99980

Average Payload Mass by F9 v1.1

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

+ Code + Markdown

Click here to ask Blackbox to help you code faster

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD FROM SCHEMA1.SPACEXTABLE WHERE BOOSTER_VERSION LIKE 'F9 v1.1 %'
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

average_payload
2337

First Successful Ground Landing Date

💡 Click here to ask Blackbox to help you code faster

```
%sql SELECT MIN(DATE) AS FIRST_LANDING_OUTCOME from SCHEMA1.SPACEXTABLE WHERE LANDING_OUTCOME = 'Success (ground pad)'
```

```
* ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

first_landing_outcome
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
💡 Click here to ask Blackbox to help you code faster
%sql SELECT BOOSTER_VERSION FROM SCHEMA1.SPACEXTABLE WHERE LANDING_OUTCOME = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
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

```
💡 Click here to ask Blackbox to help you code faster
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SCHEMA1.SPACEXTABLE GROUP BY MISSION_OUTCOME
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

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

Boosters Carried Maximum Payload

Click here to ask Blackbox to help you code faster

```
%sql SELECT BOOSTER_VERSION,LAUNCH_SITE,PAYLOAD_MASS__KG_ FROM SCHEMA1.SPACEXTABLE WHERE PAYLOAD_MASS__KG_= (SELECT MAX(PAYLOAD_MASS__KG_) FROM SCHEMA1.SPACEXTABLE) ORDER BY BOOSTER_VERSION
```

Pytho

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

booster_version	launch_site	payload_mass__kg_
F9 B5 B1048.4	CCAFS SLC-40	15600
F9 B5 B1048.5	KSC LC-39A	15600
F9 B5 B1049.4	CCAFS SLC-40	15600
F9 B5 B1049.5	CCAFS SLC-40	15600
F9 B5 B1049.7	CCAFS SLC-40	15600
F9 B5 B1051.3	CCAFS SLC-40	15600
F9 B5 B1051.4	KSC LC-39A	15600
F9 B5 B1051.6	KSC LC-39A	15600
F9 B5 B1056.4	CCAFS SLC-40	15600
F9 B5 B1058.3	KSC LC-39A	15600
F9 B5 B1060.2	KSC LC-39A	15600
F9 B5 B1060.3	CCAFS SLC-40	15600

2015 Launch Records

```
💡 Click here to ask Blackbox to help you code faster
%sql select distinct landing_outcome from schema1.spacextable
```

```
 * ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr
Done.
```

landing_outcome
Controlled (ocean)
Failure
Failure (drone ship)
Failure (parachute)
No attempt
Precluded (drone ship)
Success
Success (drone ship)
Success (ground pad)
Uncontrolled (ocean)

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

💡 Click here to ask Blackbox to help you code faster

```
%sql SELECT LANDING_OUTCOME, COUNT(*) AS COUNT_OUTCOME FROM SCHEMA1.SPACEXTABLE WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING_OUTCOME ORDER BY COUNT_OUTCOME DESC;
```

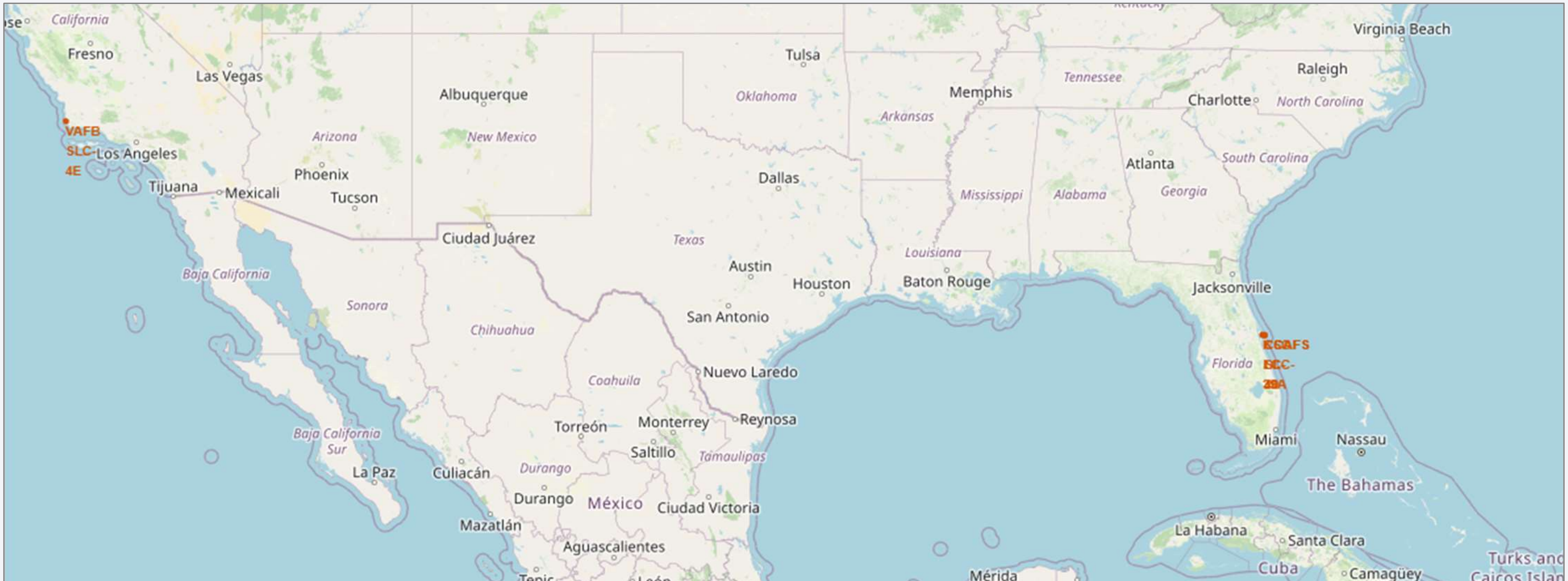
```
* ibm_db_sa://07f22adf:***@184d1610-cd16-4e0c-8c56-91794b162862.bs2ipr8s07b7bgm1ta80.databases.appdomain.cloud:30556/bludb?security=SSL
Done.
```

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

Section 3

Launch Sites Proximities Analysis

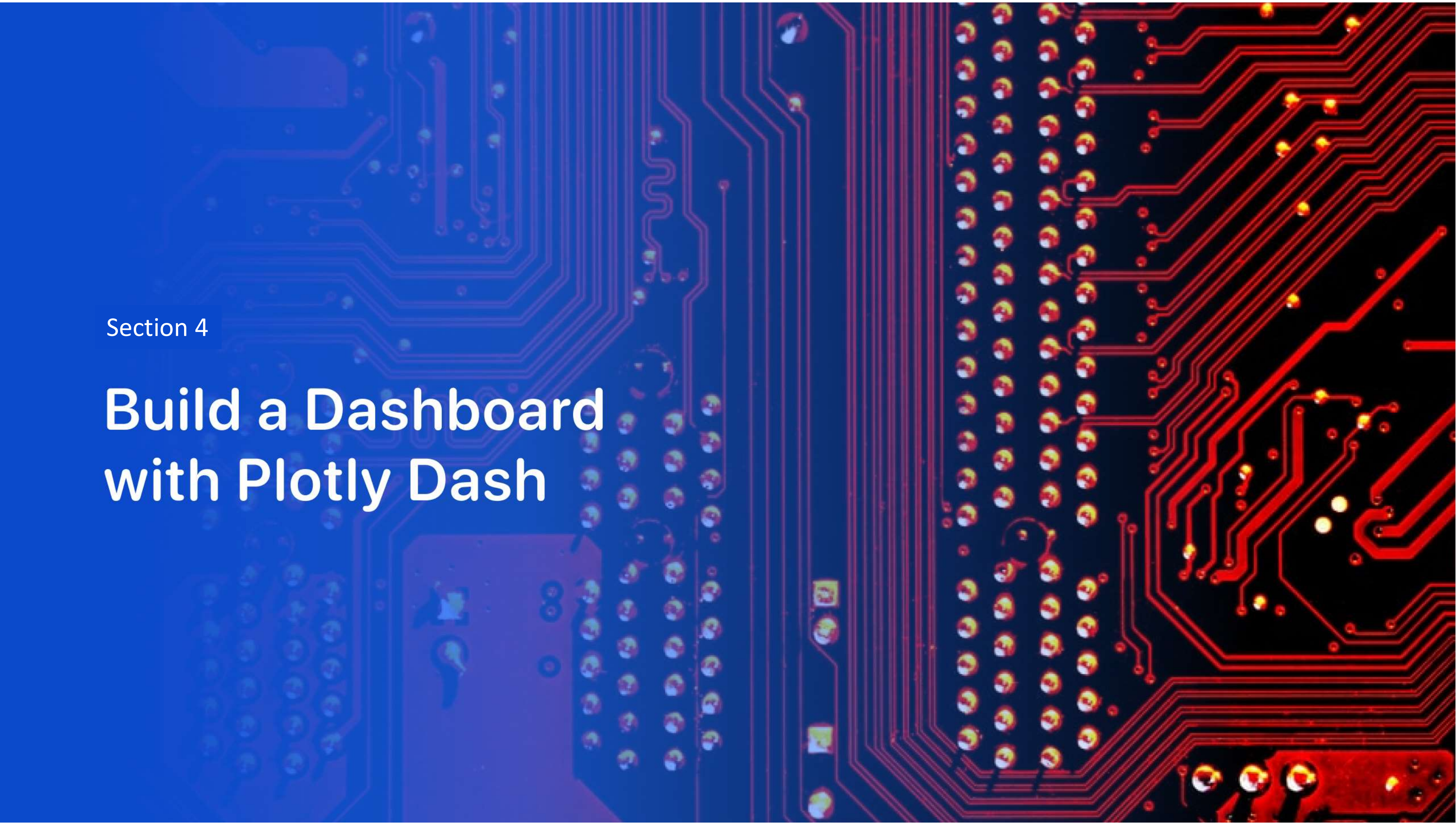
Launch Sites Global Map



- All launch sites considered in this project are in very close proximity to the coast While starting rockets towards the ocean we minimize the risk of having any debris dropping or exploding near people.

Launch Sites Location

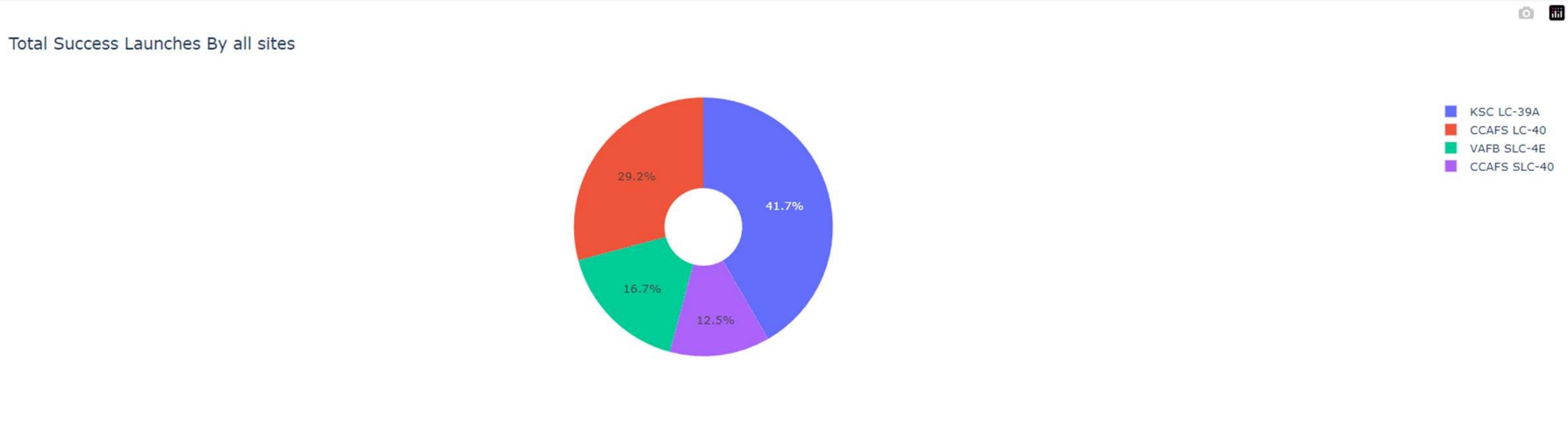
<Folium Map Screenshot 3>



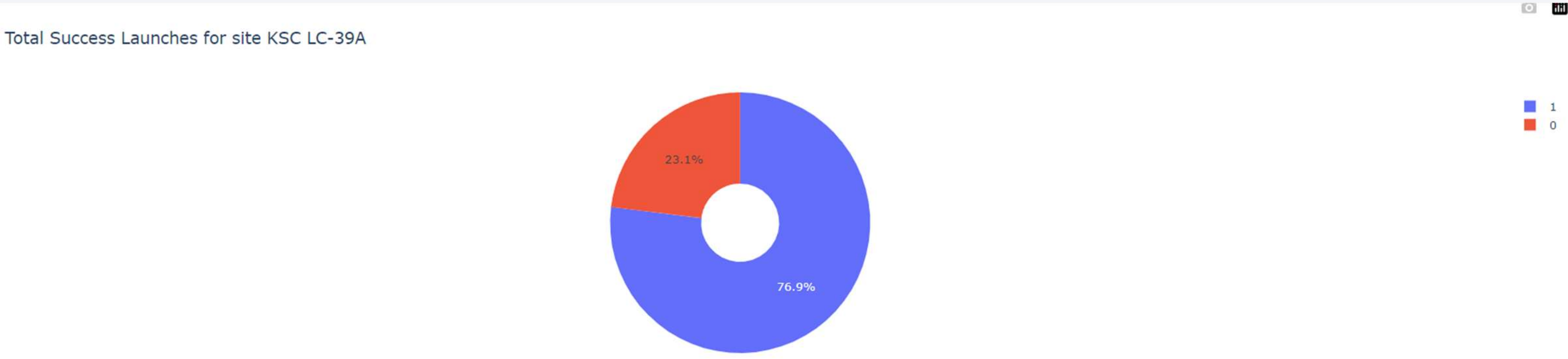
Section 4

Build a Dashboard with Plotly Dash

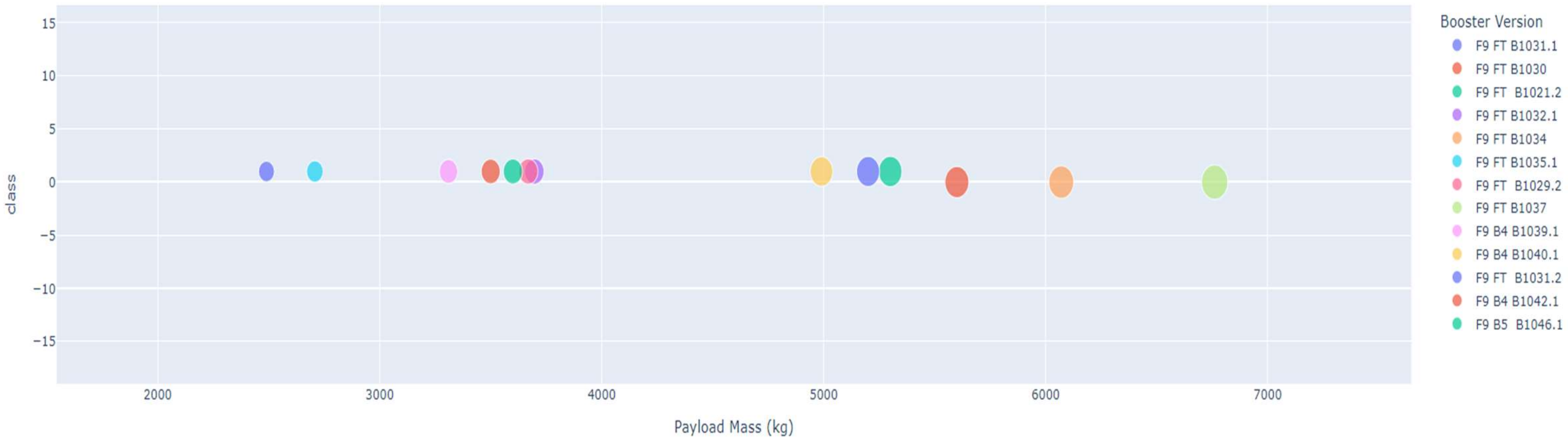
Success Launches by all sites



Launch site with highest Launch Success ratio



Payload Mass / Class



Section 5

Predictive Analysis (Classification)

Classification Accuracy

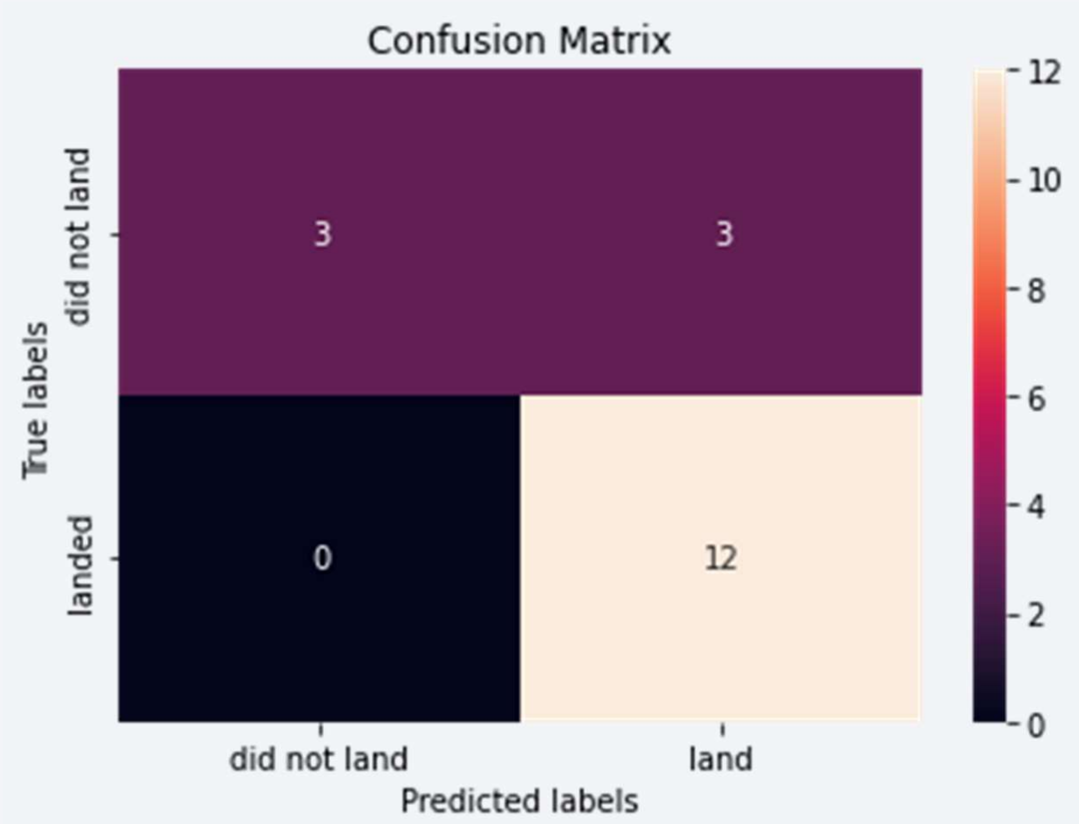
Click here to ask Blackbox to help you code faster

```
# Examining the scores from the who   (variable) logreg_cv: GridSearchCV[LogisticRegression]
jaccard_scores = [
                  jaccard_score(Y, logreg_cv.predict(X), average='binary'),
                  jaccard_score(Y, svm_cv.predict(X), average='binary'),
                  jaccard_score(Y, tree_cv.predict(X), average='binary'),
                  jaccard_score(Y, knn_cv.predict(X), average='binary'),
                 ]

f1_scores = [
             f1_score(Y, logreg_cv.predict(X), average='binary'),
             f1_score(Y, svm_cv.predict(X), average='binary'),
             f1_score(Y, tree_cv.predict(X), average='binary'),
             f1_score(Y, knn_cv.predict(X), average='binary'),
            ]

accuracy = [logreg_cv.score(X, Y), svm_cv.score(X, Y), tree_cv.score(X, Y), knn_cv.score(X, Y)]

scores = pd.DataFrame(np.array([jaccard_scores, f1_scores, accuracy]),
                      index=['Jaccard_Score', 'F1_Score', 'Accuracy'],
                      columns=['LogReg', 'SVM', 'Tree', 'KNN'])
scores
```

✓ 0.0s

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.833333	0.845070	0.921875	0.819444
F1_Score	0.909091	0.916031	0.959350	0.900763
Accuracy	0.866667	0.877778	0.944444	0.855556

Confusion Matrix



Conclusions

- The larger the amount of flight at a launch site, the greater the success rate.
- Launch Site KSC LC-39A has the highest success rate
- Orbit types : ES-L1, GEO, HEO and SSO had the highest success rate.
- There was an increase in success rate from 2013 to 2020.

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!