IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection API
  - Data Collection Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis Using SQL
  - Exploratory Data Analysis through Data Visualization
  - Interactive Visual Analytic
  - Predictive Analysis with Machine Learning

- Summary of all results
  - Result of Exploratory Data Analysis
  - Result of Interactive Visual Analytic
  - Result of Predictive Analysis with Machine Learning

Introduction

- Project background and context
- We will be working with SpaceX launch data that is gathered from an API, specifically the SpaceX REST API. SpaceX 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 SpaceX can reuse the first stage

- Problems you want to find answers
- Determine the cost of a launch by determining if the first stage will land

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Get request to the SpaceX API plus related Web Scraping
- Perform data wrangling
  - Dealing with Missing Values
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Performing of exploratory Data Analysis and determining of Training Labels, Finding of the best Hyperparameter for SVM, Classification Trees and Logistic Regression

Data Collection

- We started by **requesting rocket launch data** from SpaceX API with the related URL.
- After we **decoded the response** content as a Json using .json() and turned it into a Pandas dataframe using .json_normalize().
- Then we used the API again to **get information about the launches** using the IDs given for each launch. Specifically we used columns rocket, payloads, launchpad, and cores.
- The data from these requests were stored in lists and used to create a **new dataframe**.
- After we applied **getBoosterVersion**.
- Finally, we **constructed our dataset** using the data we have obtained by combining the columns into a dictionary

Data Collection – SpaceX API

- We **removed the Falcon 1** launches keeping only the Falcon 9 launches., **filtered the dataframe** using the BoosterVersion column to only keep the Falcon 9 launches and saved the filtered data to a **new dataframe** called data_falcon9

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40
5	8	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40
6	10	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40
7	11	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E
8	12	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40
...	...	...	...	...	...	...
89	102	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A
90	103	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A
91	104	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A
92	105	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40
93	106	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40

External reference

- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/a2048826-83dc-43a1-8238-563202af7a31/view?access_token=04112383ffa41886e57ee52f1f5948f710722e8be5a1d230083880ec12101018

Data Collection - Scraping

- First we **requested** the Falcon9 Launch Wiki page from its URL. Then we **extracted** all column/variable names from the HTML table header. Finally we created a **data frame** by parsing the launch HTML tables

- **External reference**

- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/92f140af-505e-4dbc-9dbe-0d3f5cf2822a/view?access_token=2e70e4668d2f2d4187555c1db51728c4118003ac4921c1f34b78809527ef8b0b

```
F9 v1.0B0003.1
F9 v1.0B0004.1
F9 v1.0B0005.1
F9 v1.0B0006.1
F9 v1.0B0007.1
F9 v1.1B1003
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
```

Data Wrangling

- We calculated the **number of launches** on each site, the number and occurrence of **each orbit** and the number and occurence of mission outcome per **orbit type**. Finally we created a landing **outcome** label from Outcome column

- **External reference**
- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/cc38610d-f7b3-49da-b27d-4ca0018699c2/view?access_token=718ebfbc858a80682570833042e63e53479f5b957fd659adcbbc8f74c2a9c5f9

EDA with Data Visualization

We visualized the **relationship** between:

- Flight Number and Launch Site
- Payload and Launch Site
- success rate of each orbit type
- FlightNumber and Orbit type
- Payload and Orbit type

- **External reference**

- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/07023eea-8f0e-4e40-b91b-ab9beea6d72e/view?access_token=a925af9887729c335d7e2b06f618cdae7c45b6352a04006b203e2a4a66f454d5

EDA with SQL

- Displayed the names of the unique launch sites in the space mission
- Display ed5 records where launch sites begin with the string 'CCA'
- Displayed the total payload mass carried by boosters launched by NASA (CRS)
- Displayed average payload mass carried by booster version F9 v1.1
- Listed the date when the first successful landing outcome in ground pad was achieved.
- Listed the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listed the total number of successful and failure mission outcomes
- Listed the names of the booster versions which have carried the maximum payload mass
- Listed the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Ranked 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

- **External reference**
- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/07c8ca86-b83f-46fb-9a35-cbbfd176b070/view?access_token=b0f1d9343431c92911830e6fd52a35c32376e90f6b999979e95e7215550bf898

Build an Interactive Map with Folium

- We marked all **launch sites** on a map
  - **Goal:** add each site's location on a map using site's latitude and longitude coordinates

- We marked **the success/failed launches** for each site on the map
  - **Goal:** enhance the map by adding the launch outcomes for each site, and see which sites have high success rates.

- We calculated the **distances** between a launch site to its proximities
  - **Goal:** explore and analyze the proximities of launch sites

- **External reference**

  https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/c1de30d6-eda5-46ea-b839-37217661bb93/view?access_token=6fb2972b4de70a30f5872b7e5aac12876510cec21624e687290d6184357f33cf

Build a Dashboard with Plotly Dash

- We added a Launch Site Drop-down Input Component, a callback function to render success-pie-chart based on selected site dropdown, a Range Slider to Select Payload and a callback function to render the scatter plot. Then we got the selected launch site to render a pie chart visualizing launch success counts and created a dropdown menu to select one specific site and check its detailed success rate. To find if variable payload is correlated to mission outcome, we had to be able to easily select different payload range and see if we could identify some visual patterns.

- **External reference**

- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/26464dbd-605c-49da-9e70-40cfe083f8e0/view?access_token=ccf150d655ebfc9ecf8fe1206689fdd26ba83c0ab740a6a5d32013a30d7365d4

Predictive Analysis (Classification)

- We We splitted the data into training and testing data using the function train_test_split. The training data were divided into validation data, a second set used for training data; then the models were trained and hyperparameters were selected using the function GridSearchCV. Then we created a logistic regression object and calculated the accuracy on the test data using the method score. After we examined the confusion matrix, and saw that the major problem is false positives. Finally, we created a support vector machine object, a decision tree classifier object and a k nearest neighbors object and calculated the scores of all

- **External reference**

- https://eu-de.dataplatform.cloud.ibm.com/analytics/notebooks/v2/26464dbd-605c-49da-9e70-40cfe083f8e0/view?access_token=ccf150d655ebfc9ecf8fe1206689fdd26ba83c0ab740a6a5d32013a30d7365d4

Results

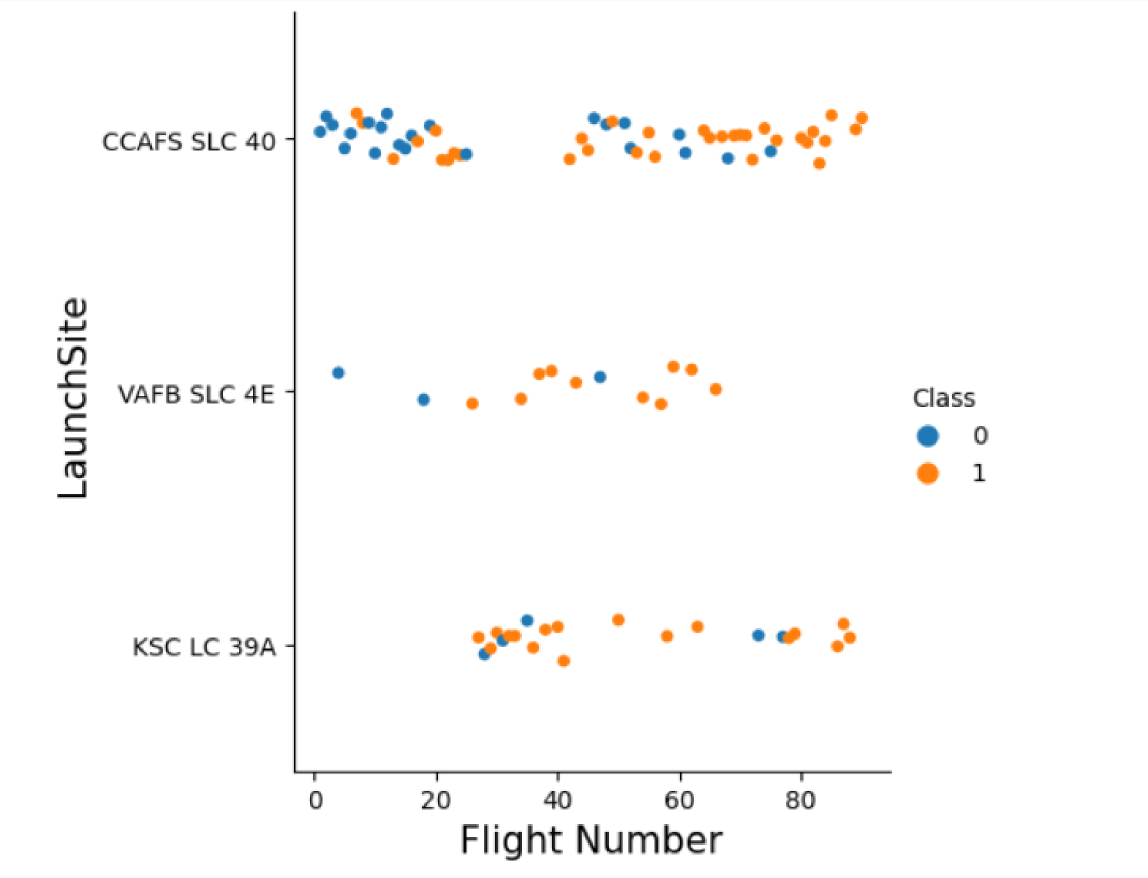
- The dataset shows better performance with low payloads
- The best prediction models are Logistic Regression, K-nearest neighbours and SVM
- The most successful performances come from KSC LC 39A site

Section 2

Insights drawn from EDA

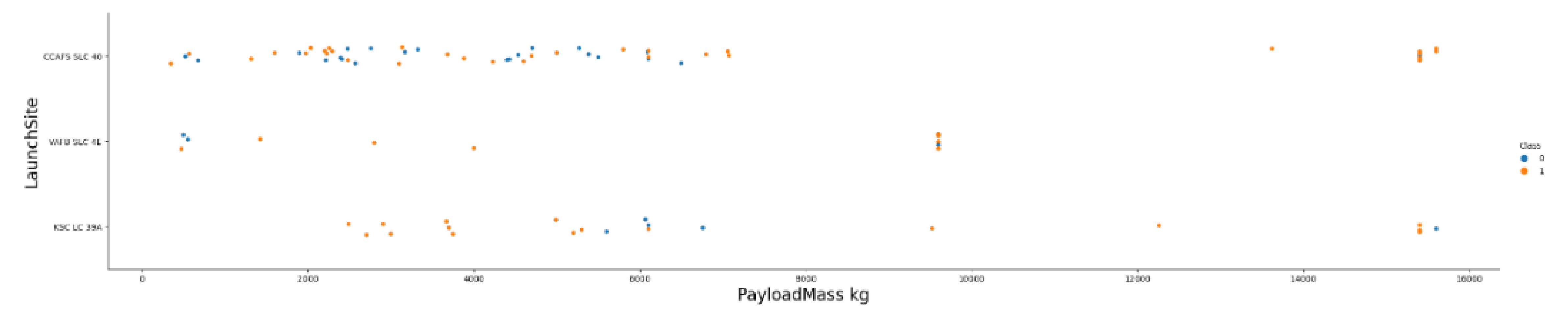
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site



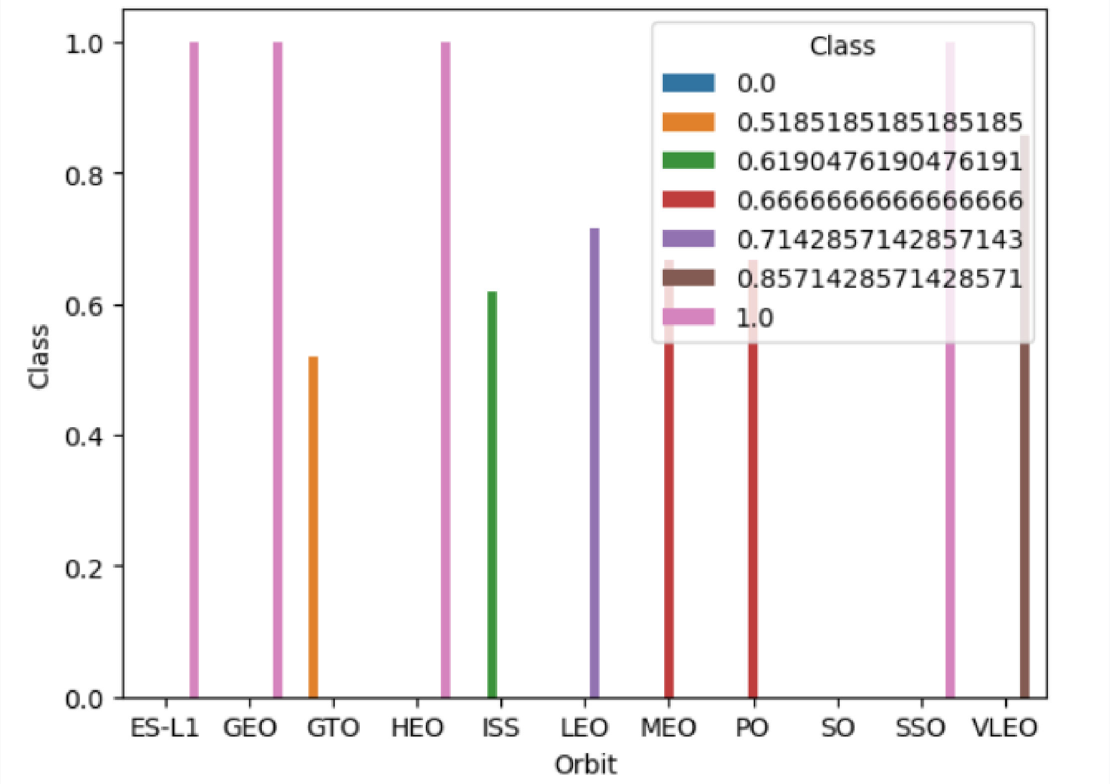
Payload vs. Launch Site

- Show a scatter plot
  of Payload vs. Launch Site



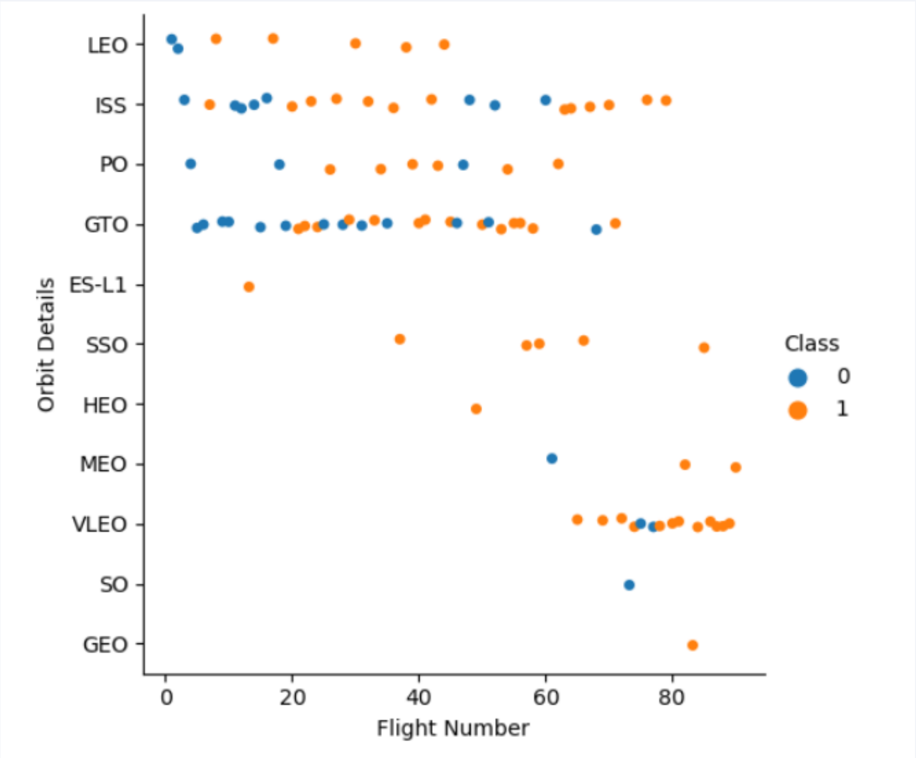
Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type



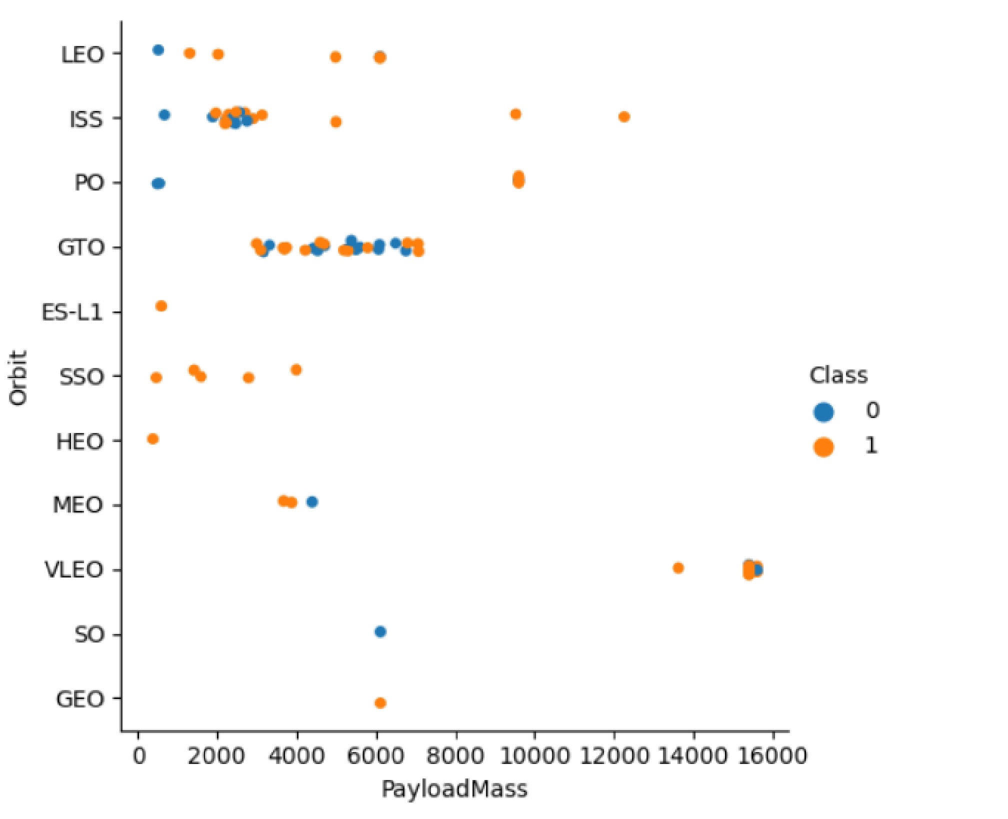
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type



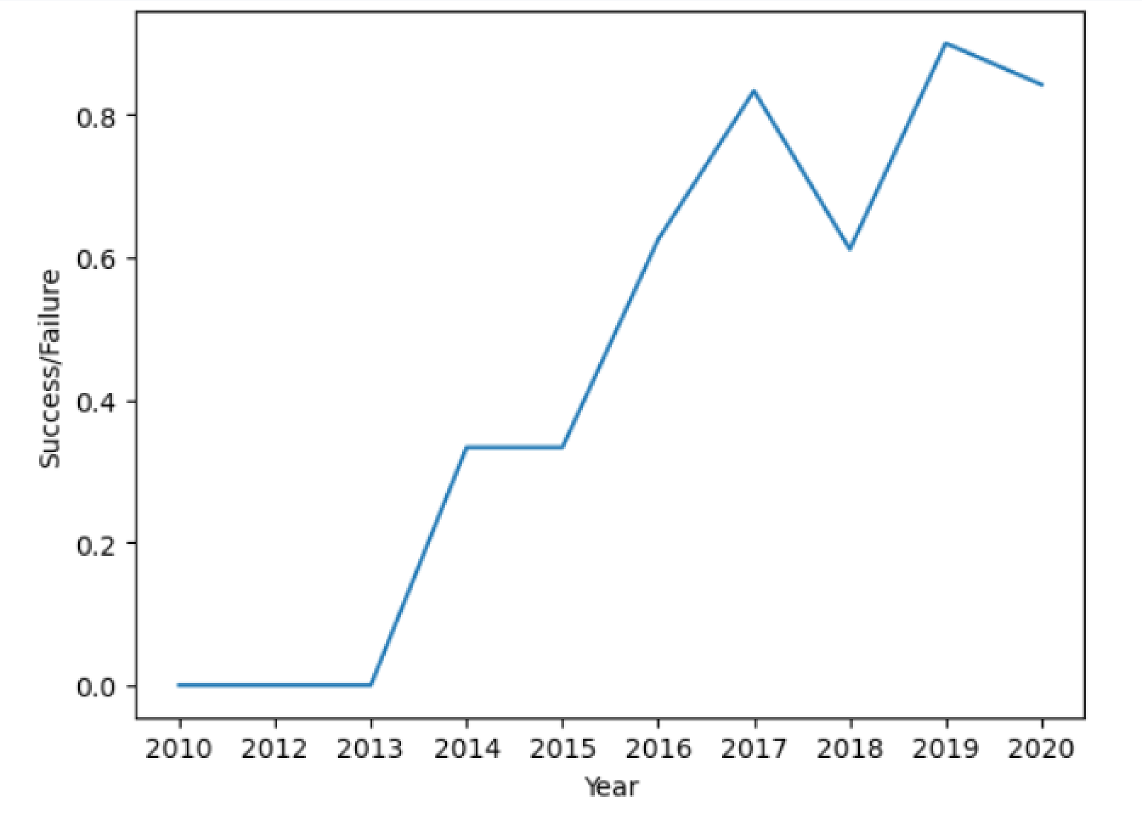
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type



Launch Success Yearly Trend

- Show a line chart of yearly average success rate



All Launch Site Names

- Find the names of the unique launch sites

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

Launch Site Names Begin with 'CCA'

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

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

1
45596

Average Payload Mass by F9 v1.1

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

1
340

First Successful Ground Landing Date

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

1
2015-12-22

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

booster_version
F9 FT B1021.1
F9 FT B1023.1
F9 FT B1029.2
F9 FT B1038.1
F9 B4 B1042.1
F9 B4 B1045.1
F9 B5 B1046.1

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

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

2015 Launch Records

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

landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

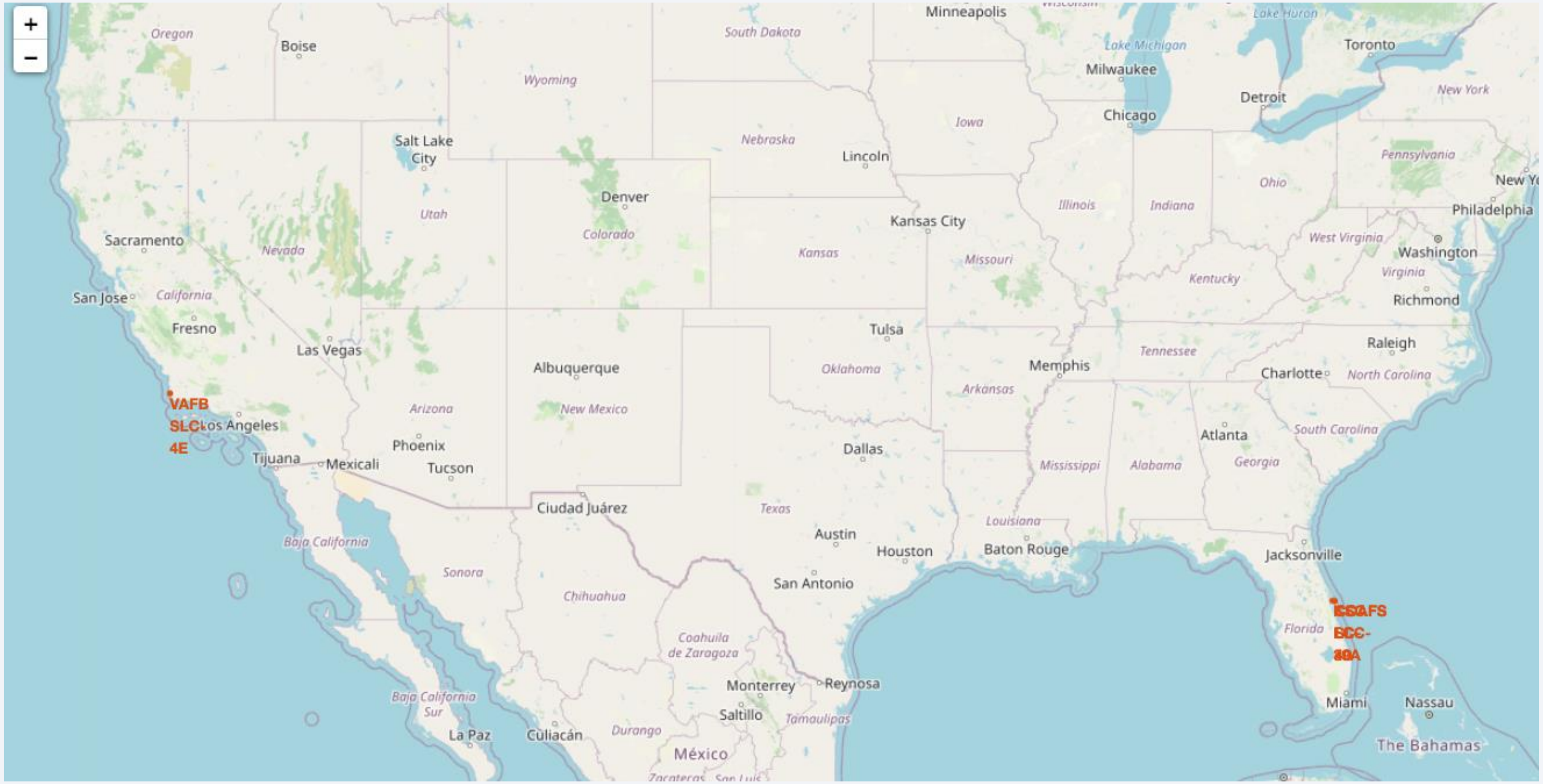
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

landing__outcome	total_number
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

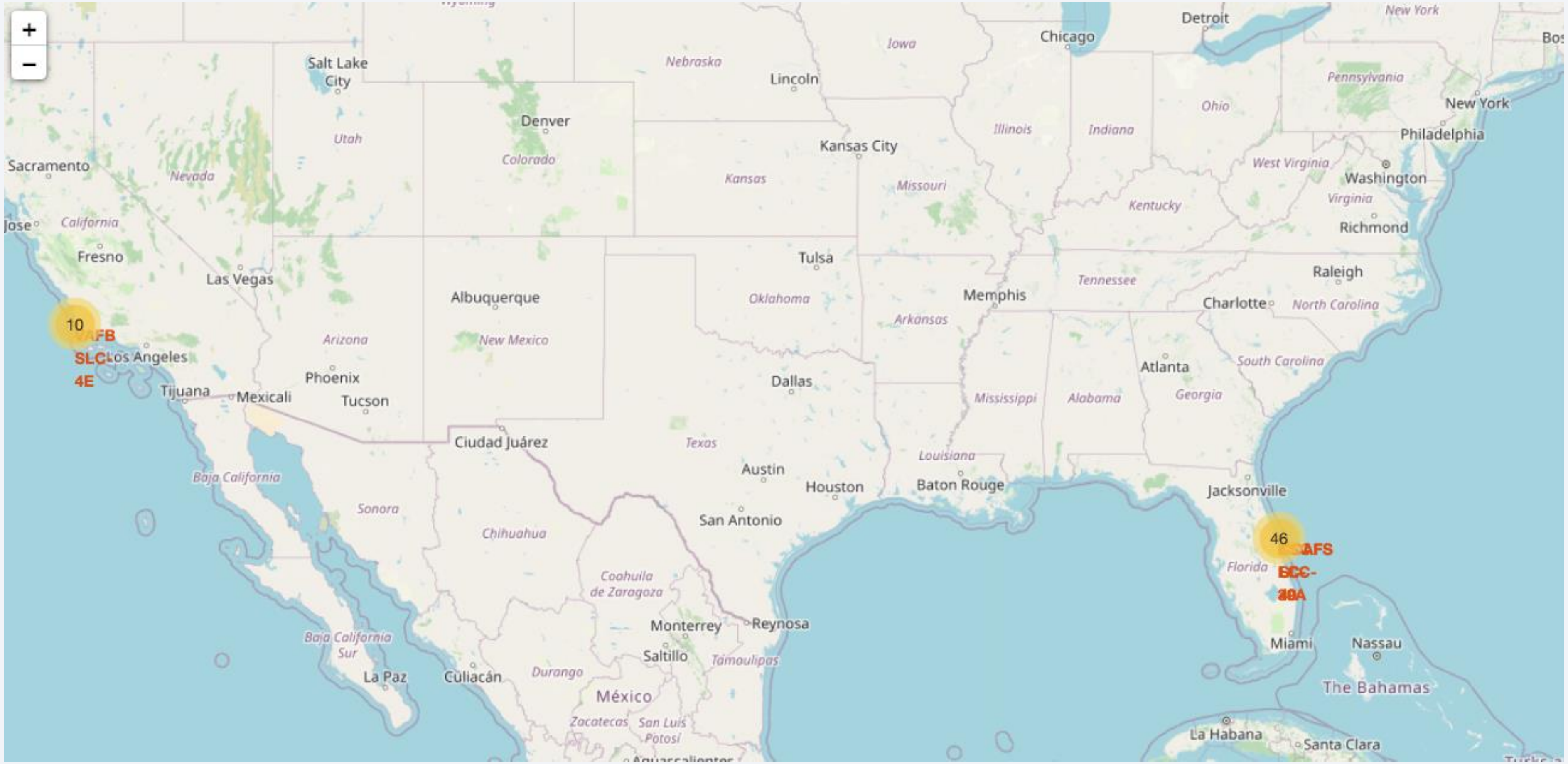
Section 3

Launch Sites Proximities Analysis

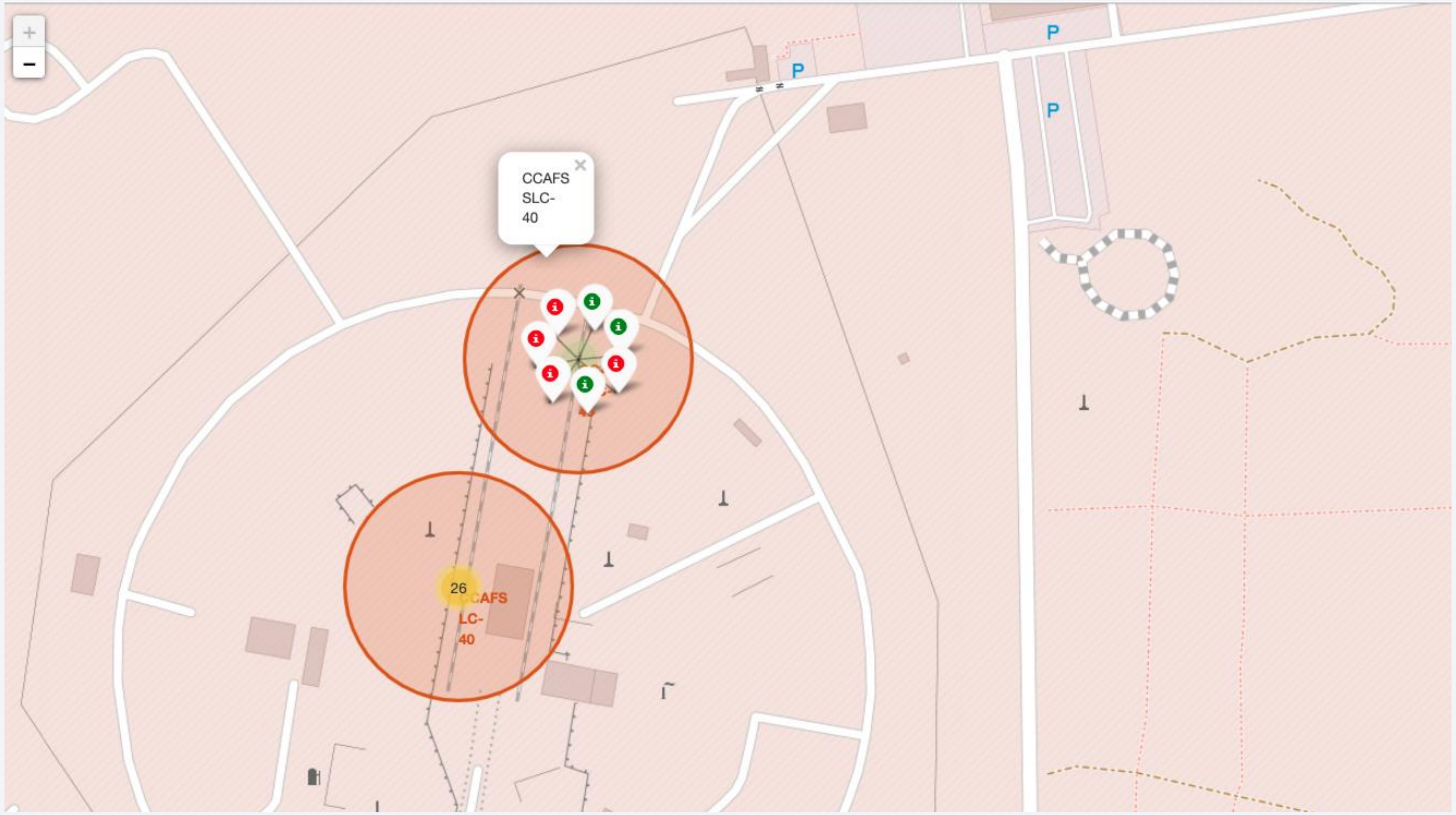
All Launch Sites on the Map



Success/fail Launches for each Site on the Map



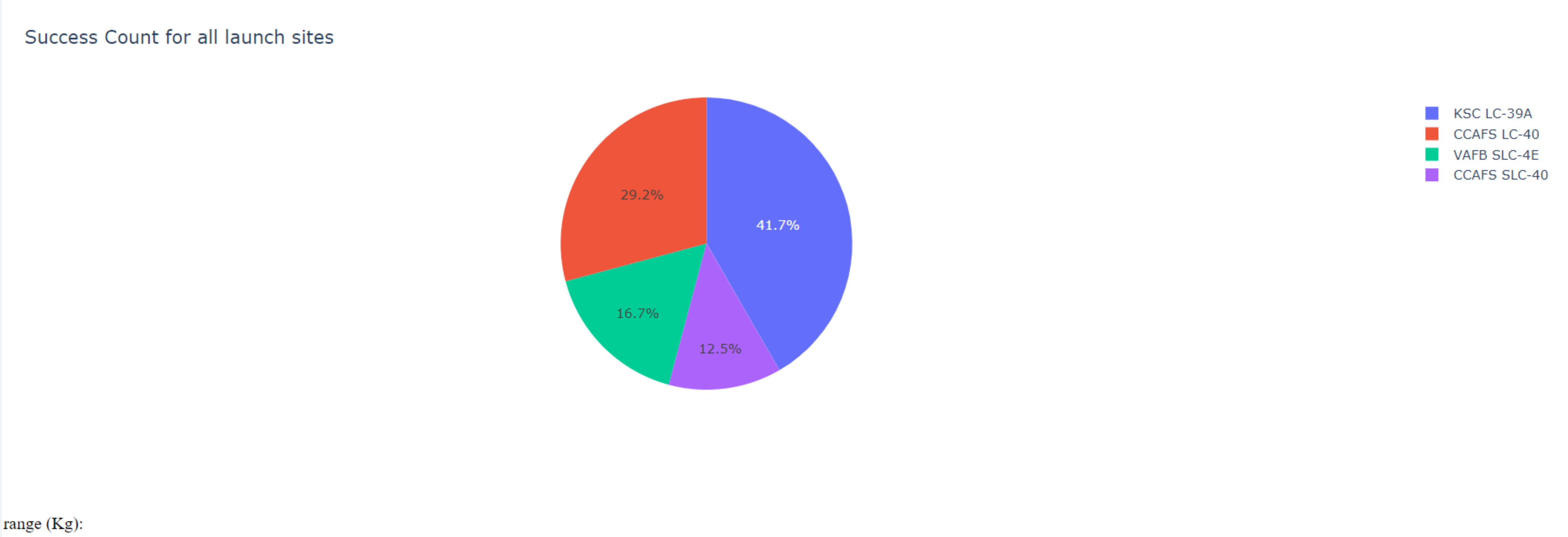
Distances between a launch site and its proximities



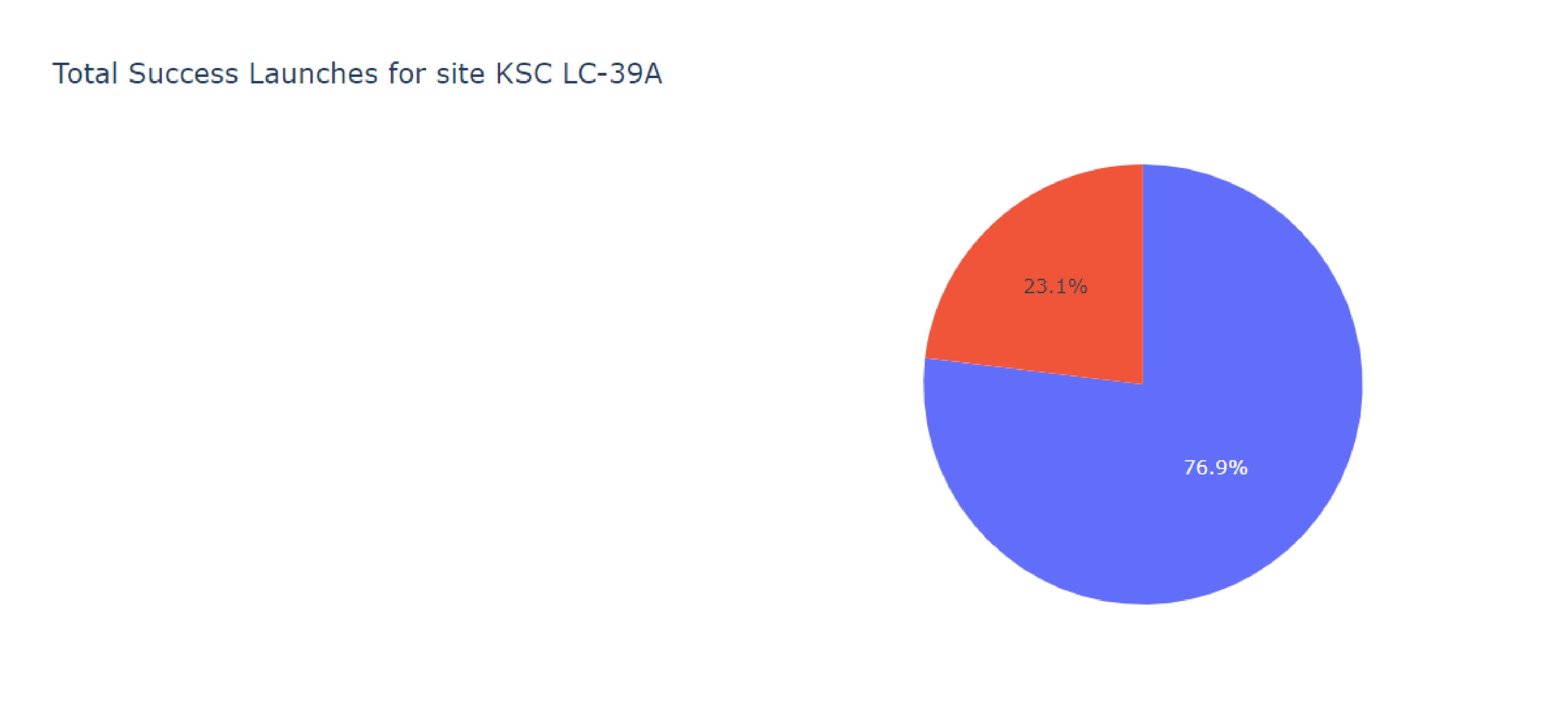
Section 4

Build a Dashboard with Plotly Dash

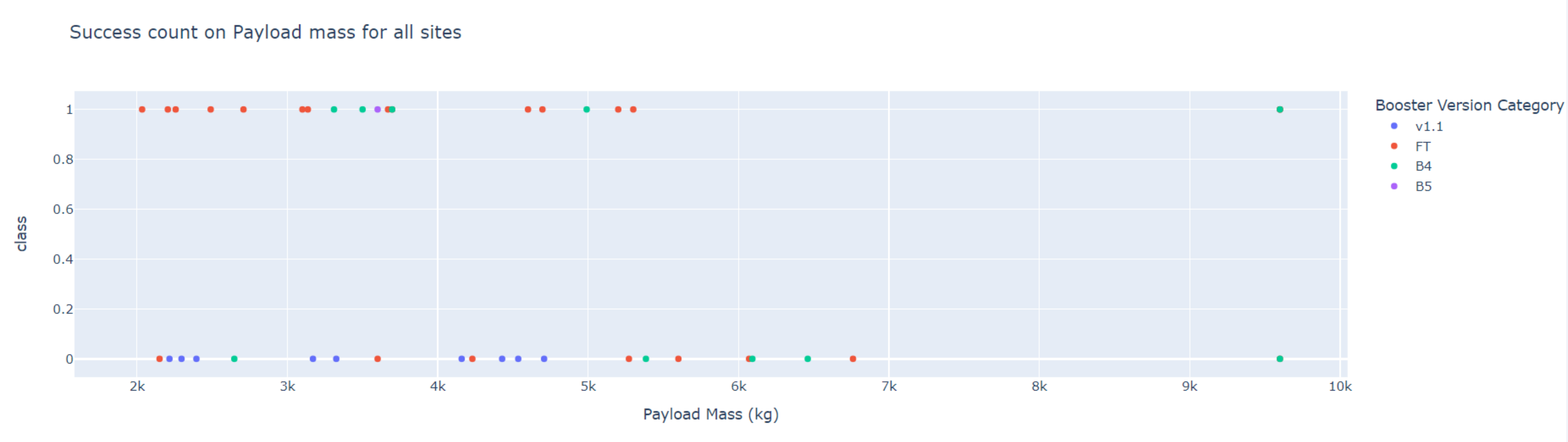
Success count for all sites



Launch site with highest launch success ratio



Payload vs. Launch Outcome



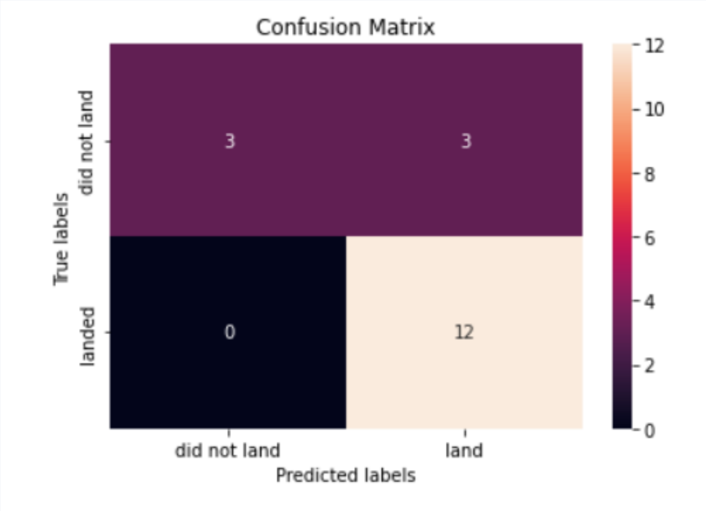
Section 5

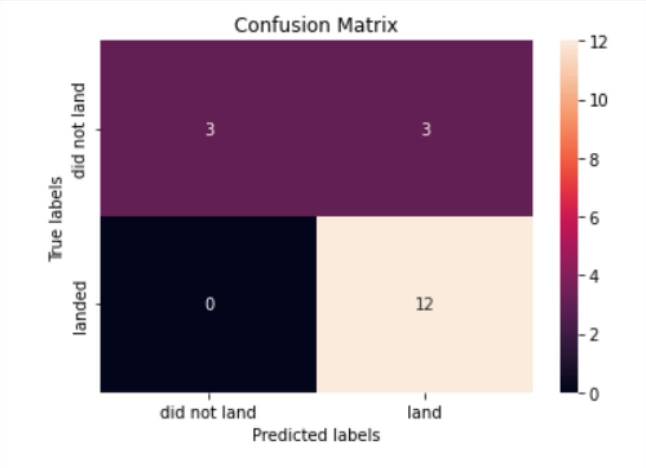
Predictive Analysis (Classification)

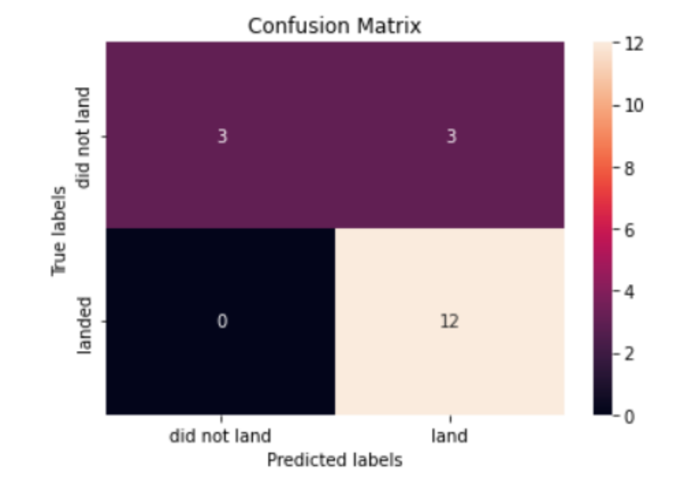
Classification Accuracy

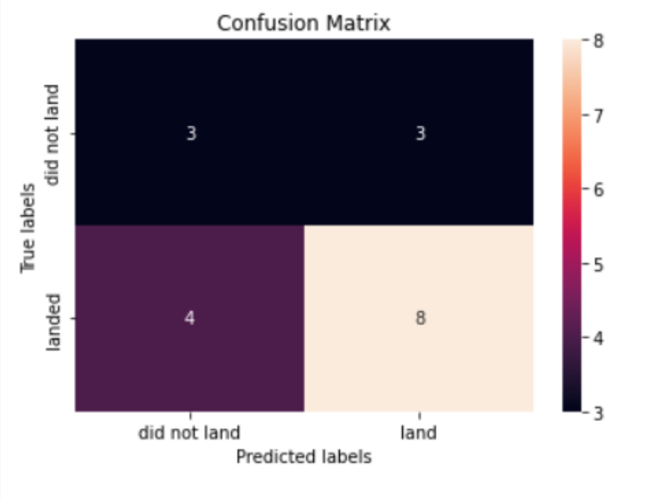
- Logistic Regression = 83,3%
- K-nearest neighbours = 83,3%
- SVM = 83,3%
- Decision tree = 77,8%

Confusion Matrices









Conclusions

- The dataset shows better performance with low payloads
- The best prediction models are Logistic Regression, K-nearest neighbours and SVM
- The most successful performances come from KSC LC 39A site

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!