IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Hamdan azhar
20/08/2023

Outline

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

Executive Summary

- The main objective is to determine if the first stage of Falcon9 rocket launch will land successfully or not.
- First data is collected from several sources like SpaceX api , Wikipedia etc.
- Then data is cleaned.
- Then data is observed using visualization in matplotlib and using SQL.
- Then data is observed making interactive maps and dashboards using folium and dash.
- Finally machine learning models are made using existing data to solve the problem.

Introduction

- The main objective is to determine if the first stage of Falcon9 rocket launch will land successfully or not.
- We have several sources to get the data for analysis.
- Several techniques of collecting data including web scraping and using APIs are used to solve this problem.
- Also visualization is done and an appropriate machine learning model is selected to solve the problem.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using the SpaceX api and then was cleaned to be used for analysis
- Perform data wrangling
  - Some missing values in data were filled and labels in data were decided for training supervised models
- Perform exploratory data analysis (EDA) using visualization and SQL
  - EDA was performed and datasets were observed using SQL and visualization tools

Methodology

Executive Summary

- Perform interactive visual analytics using Folium and Plotly Dash
  - Marking success / failed launches for launch sites and calculating their distances from proximities on the world map using folium and analyzing success / failed launces for launch sites on pie chart and payload mass vs success rate for booster version category for launch sites using dash.
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- Data collection
  - Collected data using SpaceX API.
- Initial Data Subset
  - Created a focused subset with 'rocket', 'payloads', 'launchpad', 'cores', 'flight_number', and 'date_utc'.
- Data Cleaning and Transformation:
  - Cleaned 'cores' by removing rows with multiple cores.
  - Cleaned 'payloads' by excluding rows with multiple payloads.
  - Transformed 'cores' and 'payloads' by extracting single values from lists.

Data Collection

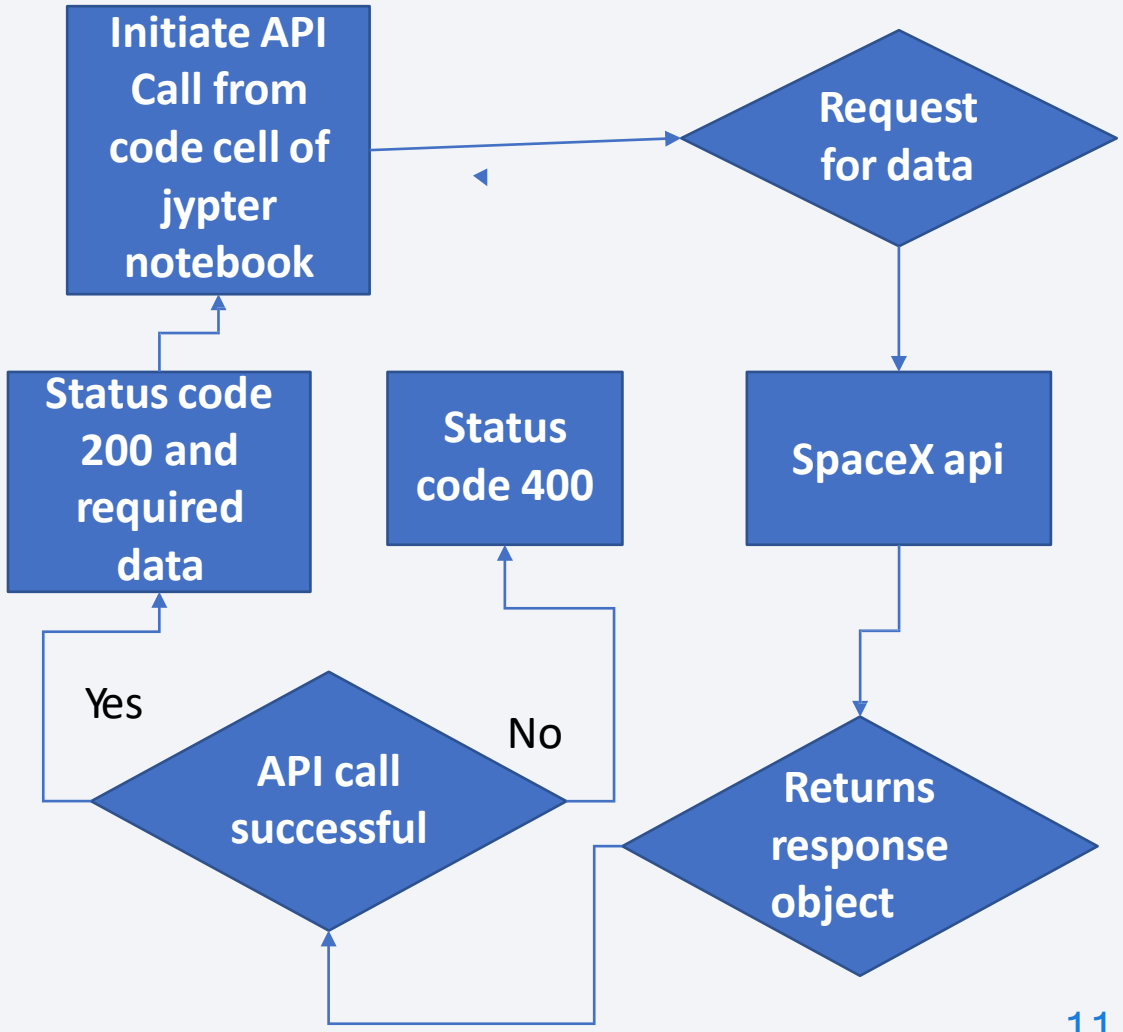
- Data Processing and Restriction:
  - Processed 'date_utc' to extract launch dates.
  - Restricted analysis to launches up to Nov 13, 2020.
  - Used the SpaceX API again to get further information for each rocket launch using data in ’rocket’ ,’payloads’, ’cores’ and ‘launchpads’ columns
- Filtering the dataframe
  - We further filter the dataset to include those records whose Booster version is Falcon 9

Data Collection

- Some Data wrangling
  - Replaced the missing data in 'Payload' column in the data set with the mean of existing data in 'Payload' column in the data set.
- Conclusion:
  - Preprocessed dataset ready for in-depth analysis.
  - Enables exploration of trends, patterns, and correlations in SpaceX rocket launches. We save the dataset into ‘dataset_part_1.csv’ for further analysis.

Data Collection - SpaceX API

- Visual representation of calling SpaceX api from code cell of jypter notebook to retrieve space data
- Github url: notebook 1

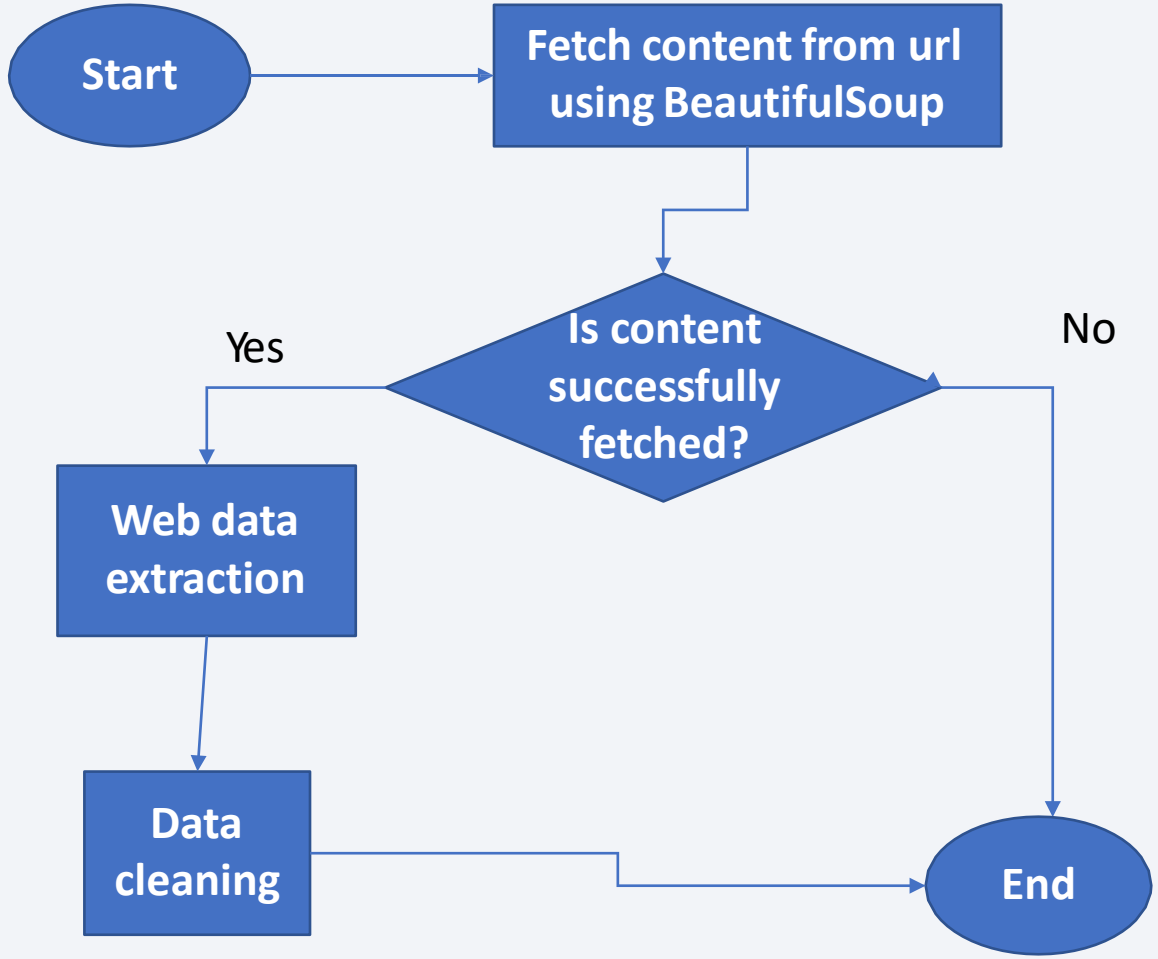


Data Collection - Scraping

- Website url: List of Falcon 9 and Falcon Heavy launches - Wikipedia
- Data was scraped from the above website using beautifulSoup
- The useful information related to falcon9 rocket launches was extracted.
- The data was further cleaned and then stored in ‘spacex_web_scraped.csv’ for further use.

Data Collection - Scraping

- Visual representation of fetching data from website url and extracting useful information from it.
- Github url: notebook 2



Data Wrangling

- Objectives completed
  - performed Exploratory Data Analysis (EDA) to find some patterns in the data and determined the label for training supervised models. We use dataset_part1.csv file for this purpose.
- Exploratory data analysis
  - Determined the number of launches on each site
  - Calculated the number and occurrence of each orbit in which rockets launches.
  - Calculated the number and occurrence of each outcome (result) of launch.

Data Wrangling

- Label Encoding
  - We observed that the ‘Outcome’ column contains categorical data which is very column of interest. We need to convert it into numerical data to use it for training supervised models.
  - We make a new column called ‘Class’ which will have a value of 1 if the categorical data in ‘Outcome’ column represents success of mission outcome landing and 0 if categorical data in in ‘Outcome’ column represents failure of mission outcome landing.
- Conclusion
  - We performed Exploratory data analysis and now have a dataset that can easily be used for training supervised models. We have saved dataset in ‘dataset_part_2.csv’ file.
- Github url: notebook 3

EDA with Data Visualization

- Objectives Completed
  - Performed exploratory Data Analysis and Feature Engineering using `Pandas` and `Matplotlib`
- Graphs
  - A category plot to observe relationship between flight number and payload mass of each Falcon9 rocket launch.
  - A category plot to observe relationship between flight number and launch site of each Falcon9 rocket launch.
  - A category plot to observe relationship between payload mass and launch site of each Falcon9 rocket launch.

EDA with Data Visualization

- Graphs
  - A bar plot to observe relationship between success rate and orbit type of each Falcon9 rocket launch.
  - A category plot to observe relationship between flight number and orbit type of each Falcon9 rocket launch.
  - A category plot to observe relationship between payload mass and orbit type of each Falcon9 rocket launch.
  - line chart to observe relationship between year and average success rate, to get the average launch success trend.

EDA with Data Visualization

- Features Engineering
  - After getting preliminary insights about how each important variable affects he success rate, feature engineering was performed in which some columns from the dataset that were not important in success prediction were omitted.
  - Then one hot encoding was performed on the columns: 'FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused', 'Legs', 'LandingPad', 'Block', 'ReusedCount', 'Serial'.
  - Finally the dataset was saved in 'dataset_part_3.csv' file.
- Github url: notebook 5

EDA with SQL

- Queries performed in SQLLite (1 to 6) Dataset used: Spacex.csv
  - Displaying unique launch sites in space mission
  - Displaying 5 records where launch sites starting with letters 'CCA'
  - Displaying total payload mass carried by boosters launched by NASA (CRS)
  - Displaying average payload mass carried by booster version F9 v1.1
  - Listing the date when the first successful landing outcome in ground pad was acheived.
  - Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

EDA with SQL

- Queries performed in SQLLite (7 to 10) Dataset used: Spacex.csv
  - Listing the total number of successful and failure mission outcomes
  - Listing the names of the booster versions which have carried the maximum payload mass using a subquery
  - Listing the records which will display the month names, failure landing outcomes in drone ship ,booster versions, launch site for the months in year 2015.
  - Ranking the count of landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order.
- Github url: notebook 4

Build an Interactive Map with Folium

- Objects used to build map
  - Map object from folium library that takes location coordinates to build the map at the needed position.
  - Circle object from folium library to add a highlighted text area at a specific coordinate.
  - Popup object from folium library to add a popup label on the highlighted text area of the Circle object.
  - Marker object from folium library to add a marker at a specific coordinate.
  - DivIcon object from folium library to add a custom icon as a text label on the marker object.

Build an Interactive Map with Folium

- Objects used to build map.
  - MarkerCluster object to simplify the map containing many markers at the same coordinate.
  - MousePosition object on the map to get coordinate for a mouse over a point on the map
- Github url: notebook 6

Build a Dashboard with Plotly Dash

- Graphs drawn
  - Pie chart to show the success launches for all sites to see which launch site has the highest success rate.
  - Pie chart to show success rate for each launch site to see the proportion of failed and successful launches.
  - Payload Mass vs success outcome scatter chart for all launch sites categorized by booster version category. This was done to see which booster version category has the highest success rate for all launch sites in a specific payload range.
  - Payload Mass vs success outcome scatter chart for each launch site categorized by booster version category. This was done to see which booster version category has the highest success rate for each launch sites in a specific payload range.

Predictive Analysis (Classification)

- Datasets used
  - Data in ‘dataset_part_2.csv’ file was used for getting data in ‘Class’ column which contains binary value 1 or 0 for whether launch was successful or not.
  - Data in ‘dataset_part_3.csv’ file was used for getting data that will be used to train machine learning models.
- Process for training machine learning models
  - The input data was standardized to be used for training machine learning models.
  - Then the input and output data was splitted into training and testing data.
  - GridSearchCV object was used to find the best parameters for our machine learning models

Predictive Analysis (Classification)

- Process for training machine learning models
  - SVM, logistic regression, K neighbors classifier , decision tree classifier models were trained.
- Github url: notebook 7

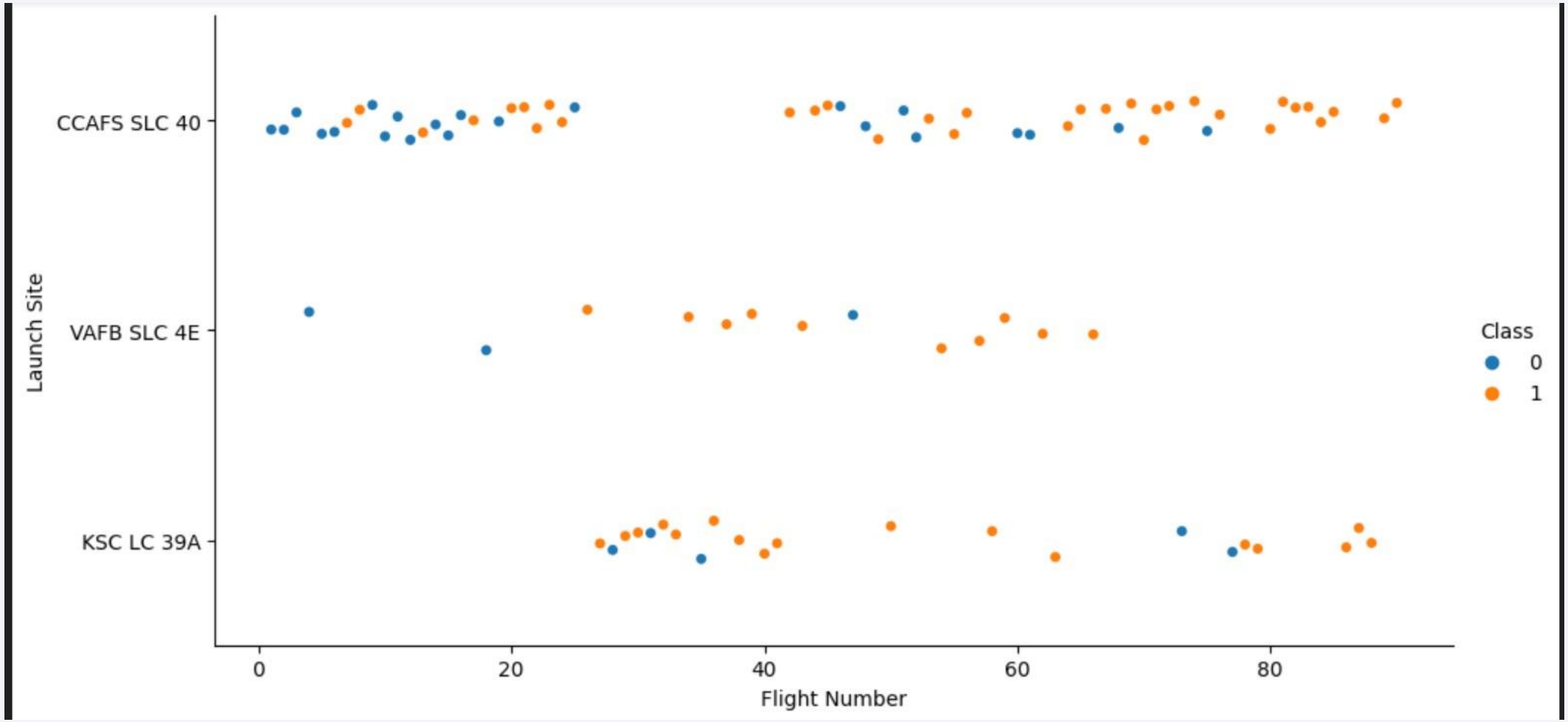
Results

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

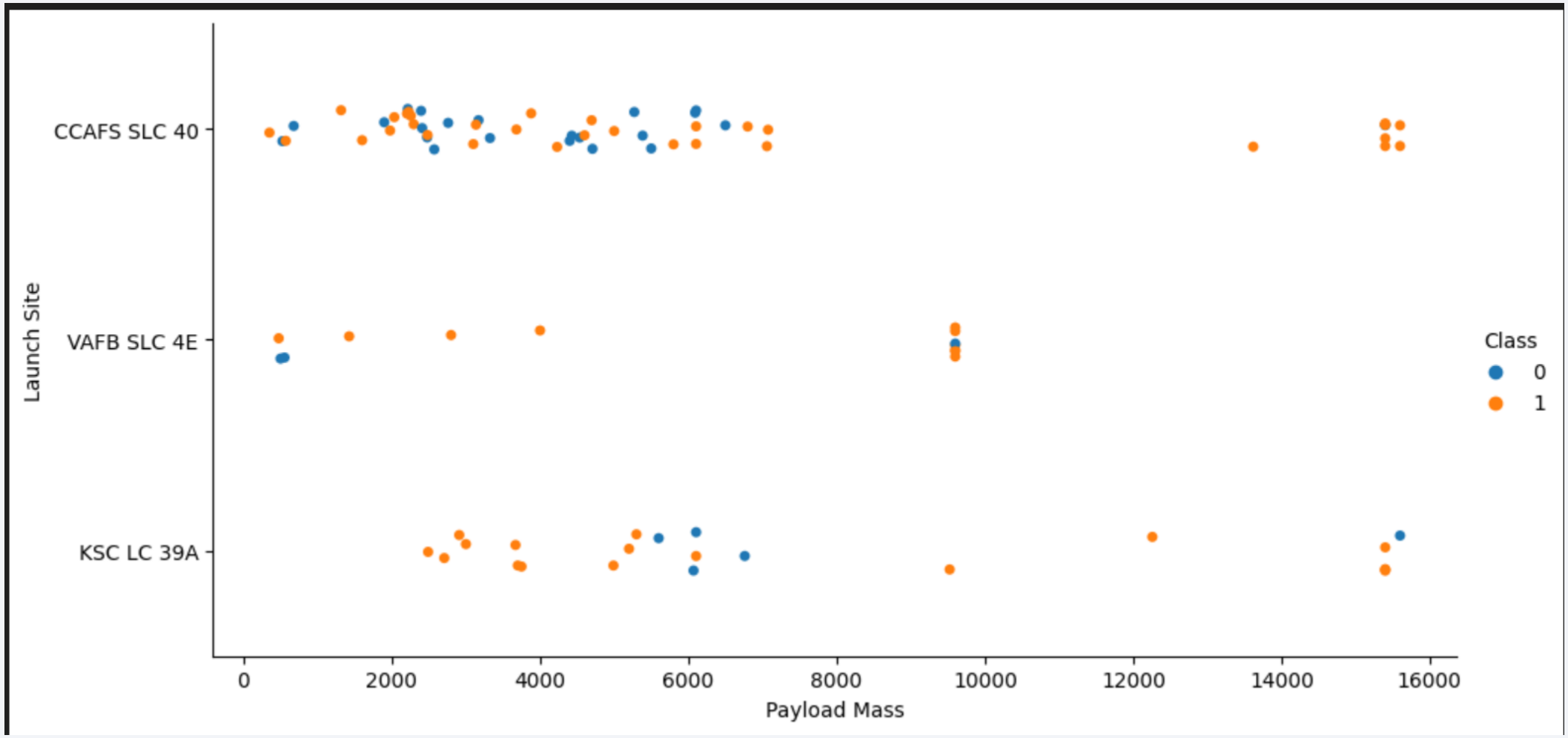
Section 2

Insights drawn from EDA

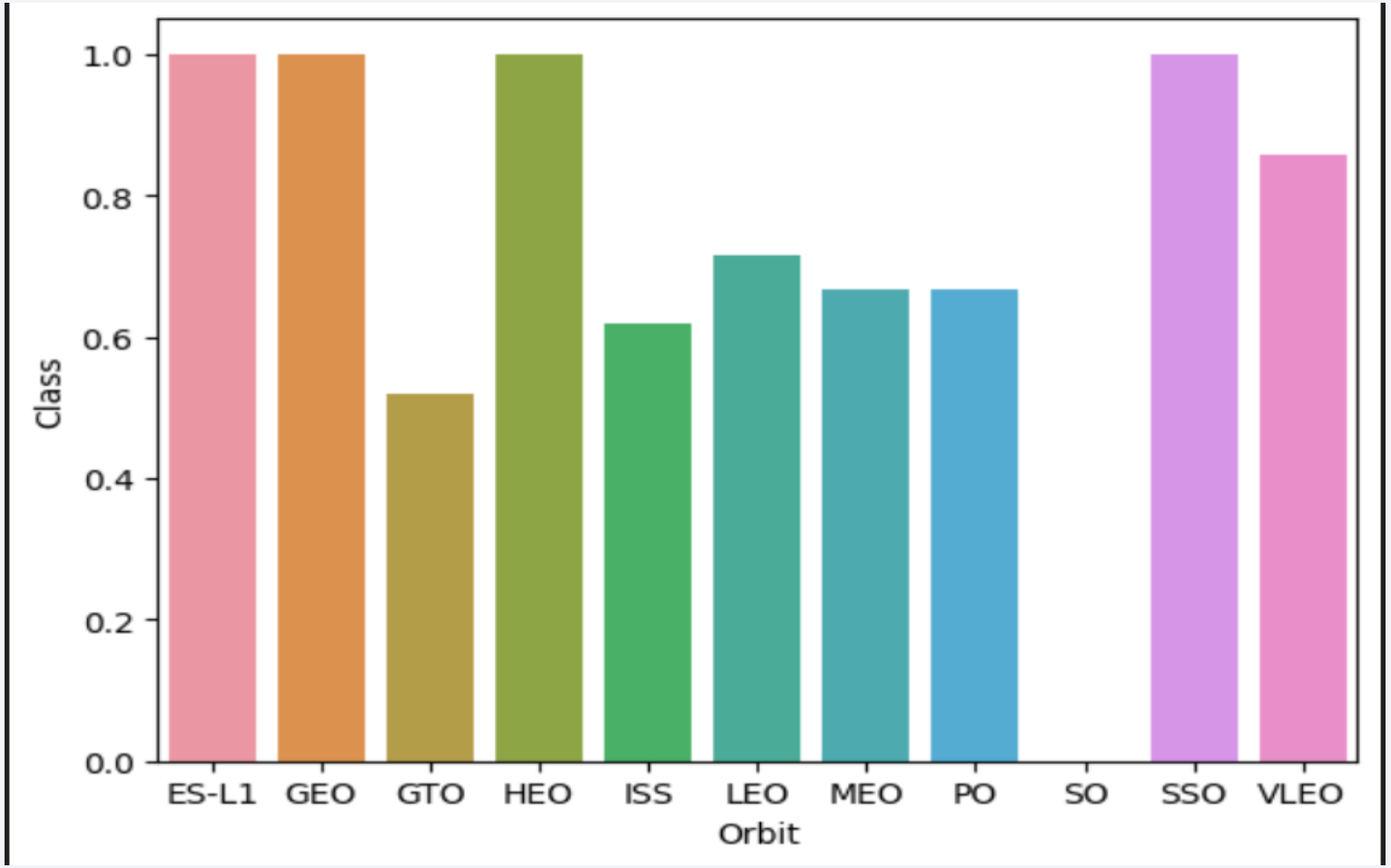
Flight Number vs. Launch Site



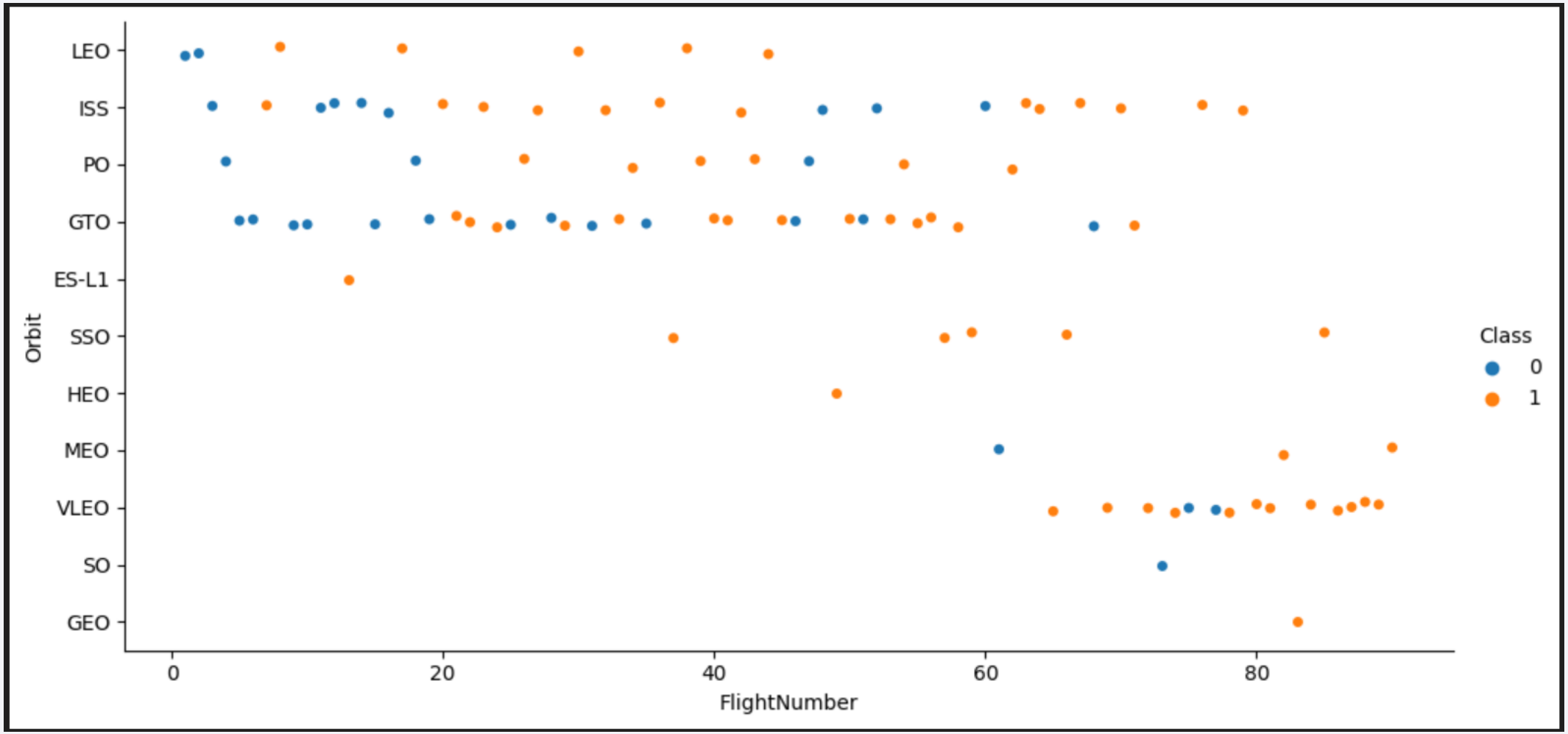
Payload vs. Launch Site



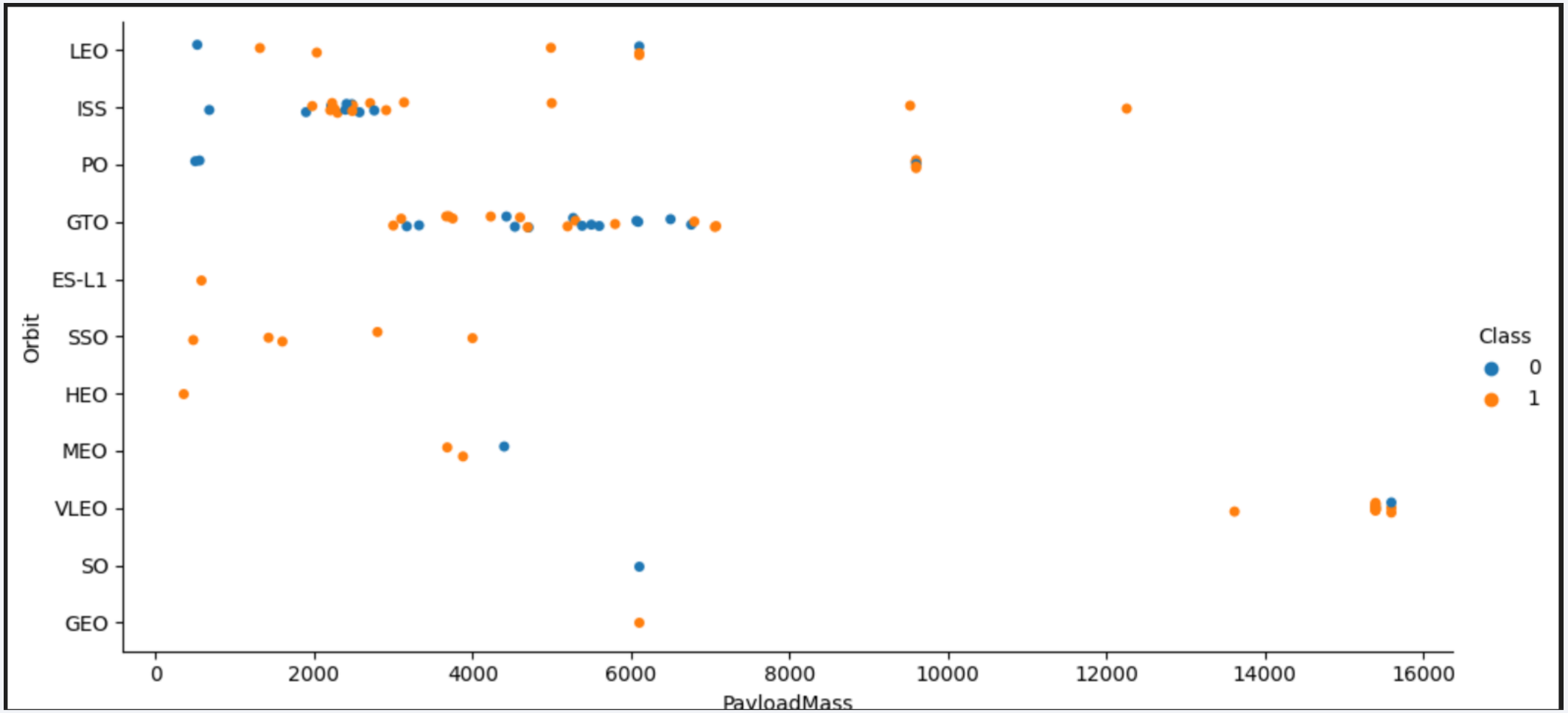
Success Rate vs. Orbit Type



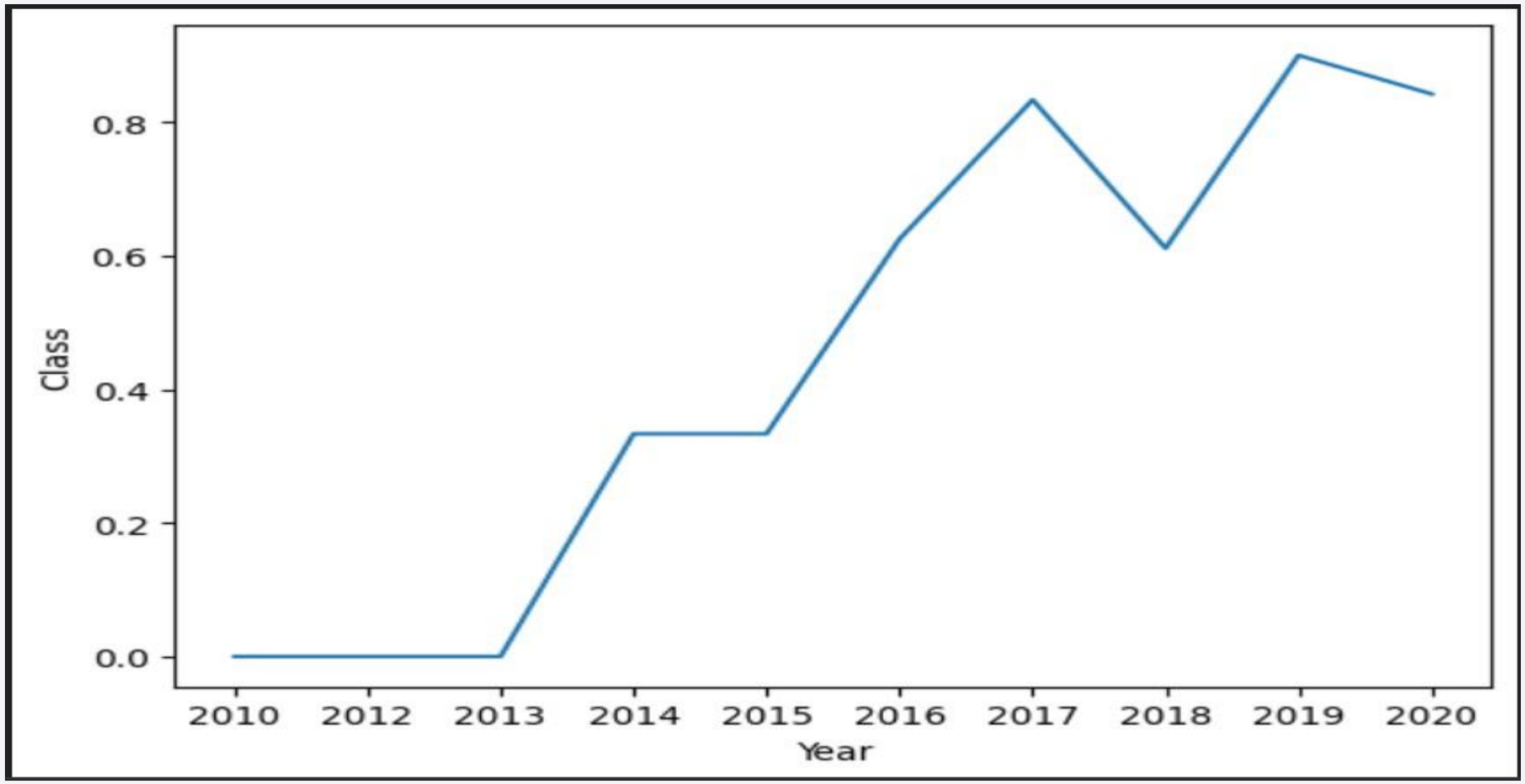
Flight Number vs. Orbit Type



Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

```
%sql select distinct(Launch_Site) from SPACEXTABLE
```

[9] ✓ 0.0s

```
 * 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'

```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' LIMIT 5
```

✓ 0.0s Python

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

```
%%sql SELECT SUM(PAYLOAD_MASS__KG_) as SUM
FROM SPACEXTABLE
WHERE Customer = 'NASA (CRS)';
```

✓ 0.0s

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

SUM
45596

Average Payload Mass by F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) as average FROM SPACEXTABLE where Booster_version like 'F9 v1.1%'
```

✓ 0.0s

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

average
2534.6666666666665

First Successful Ground Landing Date

```
%sql select min(Date) as FirstSuccessfulLandingDate from SPACEXTABLE where Mission_Outcome like 'Success%'
```

✓ 0.0s

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

FirstSuccessfulLandingDate
2010-04-06

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql select distinct(Booster_Version) from SPACEXTABLE
where (PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000)
and Landing_Outcome like 'Success%'
```

✓ 0.0s

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

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 B4 B1043.1
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1048.3

Total Number of Successful and Failure Mission Outcomes

```
%%sql select CASE when Mission_Outcome like 'Success%' then 'Success'
else 'Failure' END as Grouped_Outcome,count(*) as Total_Count
from SPACEXTABLE group by Grouped_Outcome
```

✓ 0.0s

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

Grouped_Outcome	Total_Count
Failure	1
Success	100

Boosters Carried Maximum Payload

```
%sql select distinct Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

✓ 0.0s

```
 * 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

```
%%sql SELECT
    CASE
        WHEN substr(Date, 6, 2) = '01' THEN 'January'
        WHEN substr(Date, 6, 2) = '02' THEN 'February'
        WHEN substr(Date, 6, 2) = '03' THEN 'March'
        WHEN substr(Date, 6, 2) = '04' THEN 'April'
        WHEN substr(Date, 6, 2) = '05' THEN 'May'
        WHEN substr(Date, 6, 2) = '06' THEN 'June'
        WHEN substr(Date, 6, 2) = '07' THEN 'July'
        WHEN substr(Date, 6, 2) = '08' THEN 'August'
        WHEN substr(Date, 6, 2) = '09' THEN 'September'
        WHEN substr(Date, 6, 2) = '10' THEN 'October'
        WHEN substr(Date, 6, 2) = '11' THEN 'November'
        WHEN substr(Date, 6, 2) = '12' THEN 'December' END AS MonthName,Landing_Outcome,Booster_Version,Launch_Site
FROM SPACEXTABLE
WHERE Landing_Outcome like 'Failure%' AND substr(Date,0,5)='2015'
```

✓ 0.0s

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

MonthName	Landing_Outcome	Booster_Version	Launch_Site
October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

```
%%sql SELECT Landing_Outcome, COUNT(Landing_Outcome)
FROM (
    SELECT * FROM SPACEXTABLE
    WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
) AS subquery
GROUP BY Landing_Outcome
ORDER BY COUNT(Landing_Outcome) DESC
```

✓ 0.0s

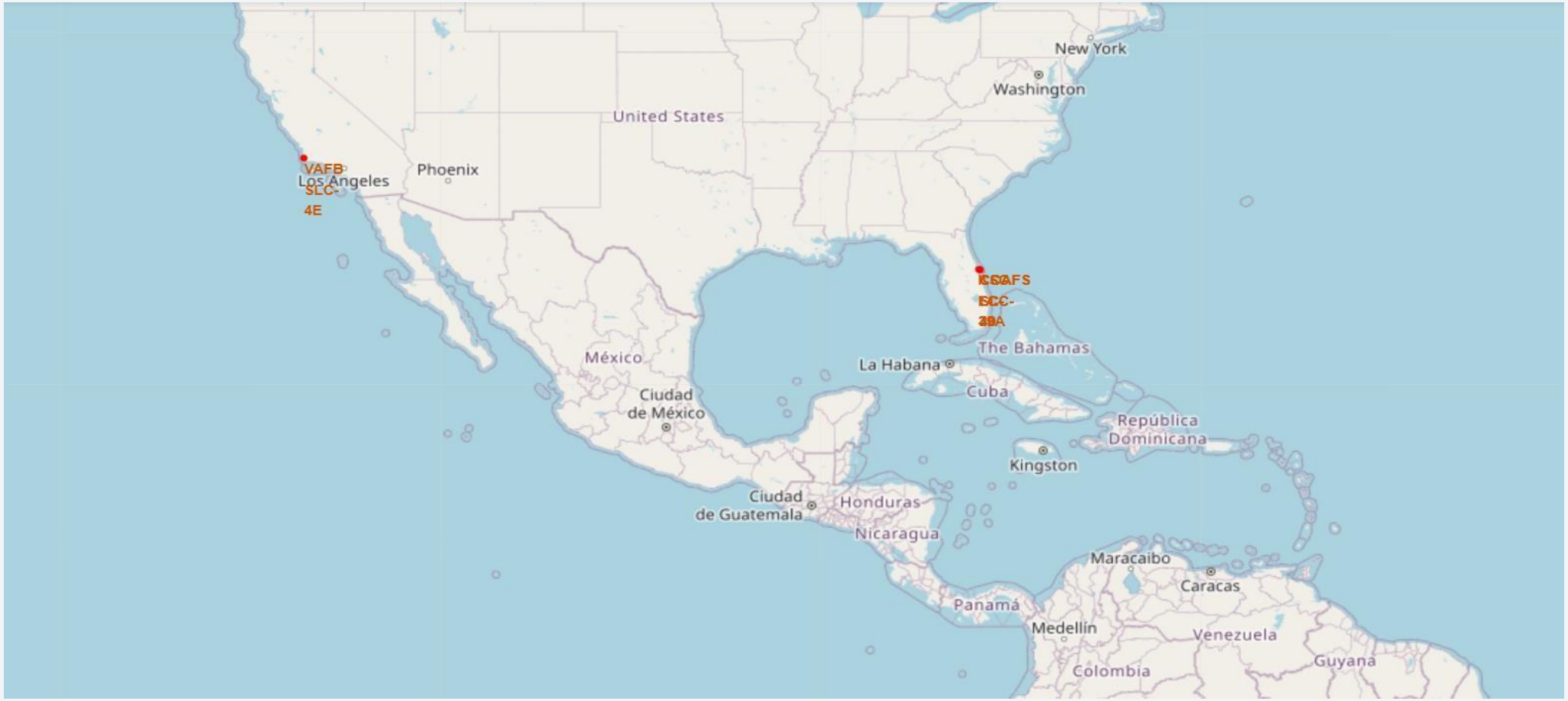
```
 * sqlite:///my_data1.db
Done.
```

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

Section 3

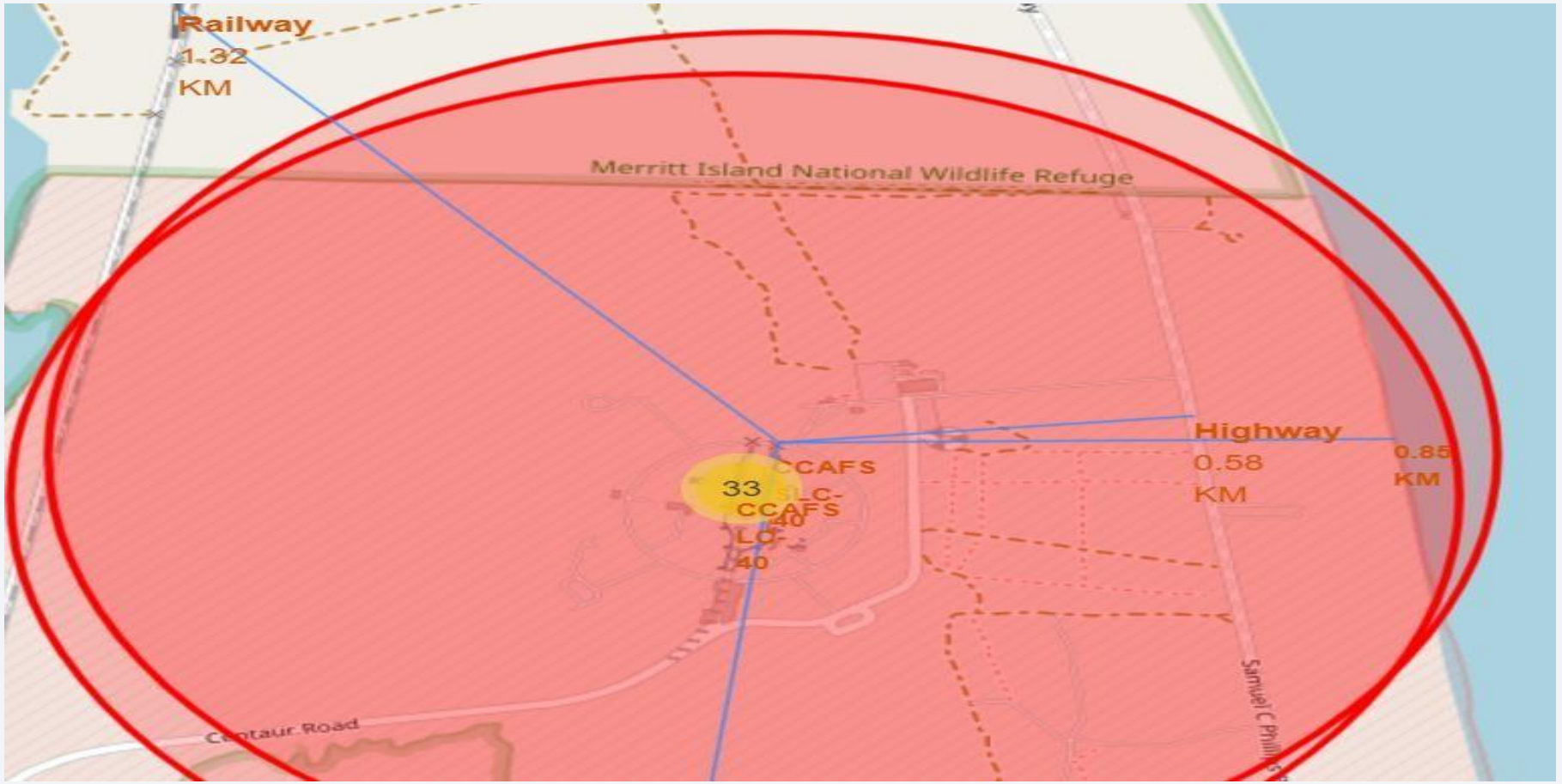
Launch Sites Proximities Analysis

Map Showing SpaceX Launch Sites Marked on the Globe

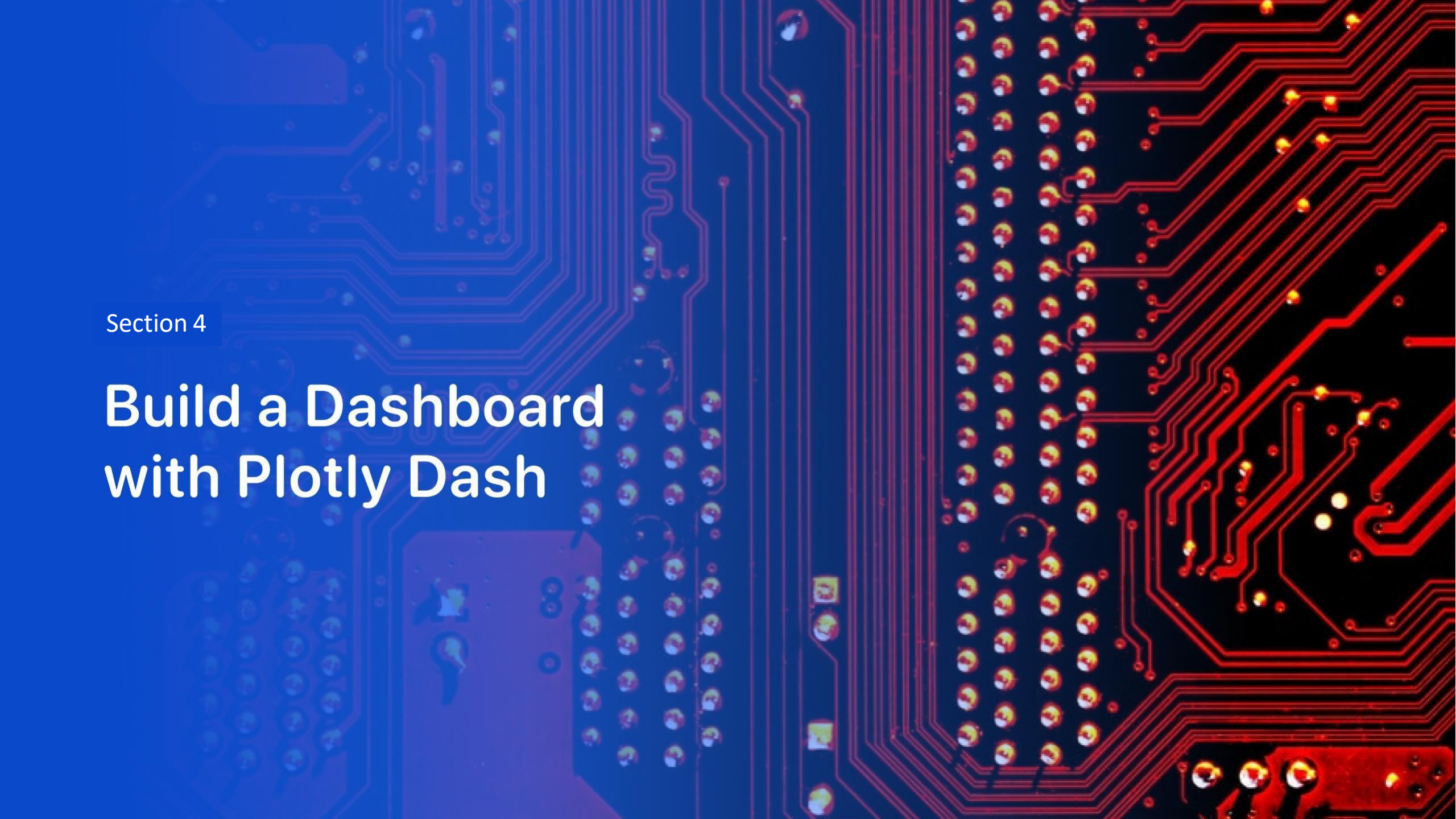


Interactive Folium Map: SpaceX Launch Sites and Launch Outcomes Worldwide

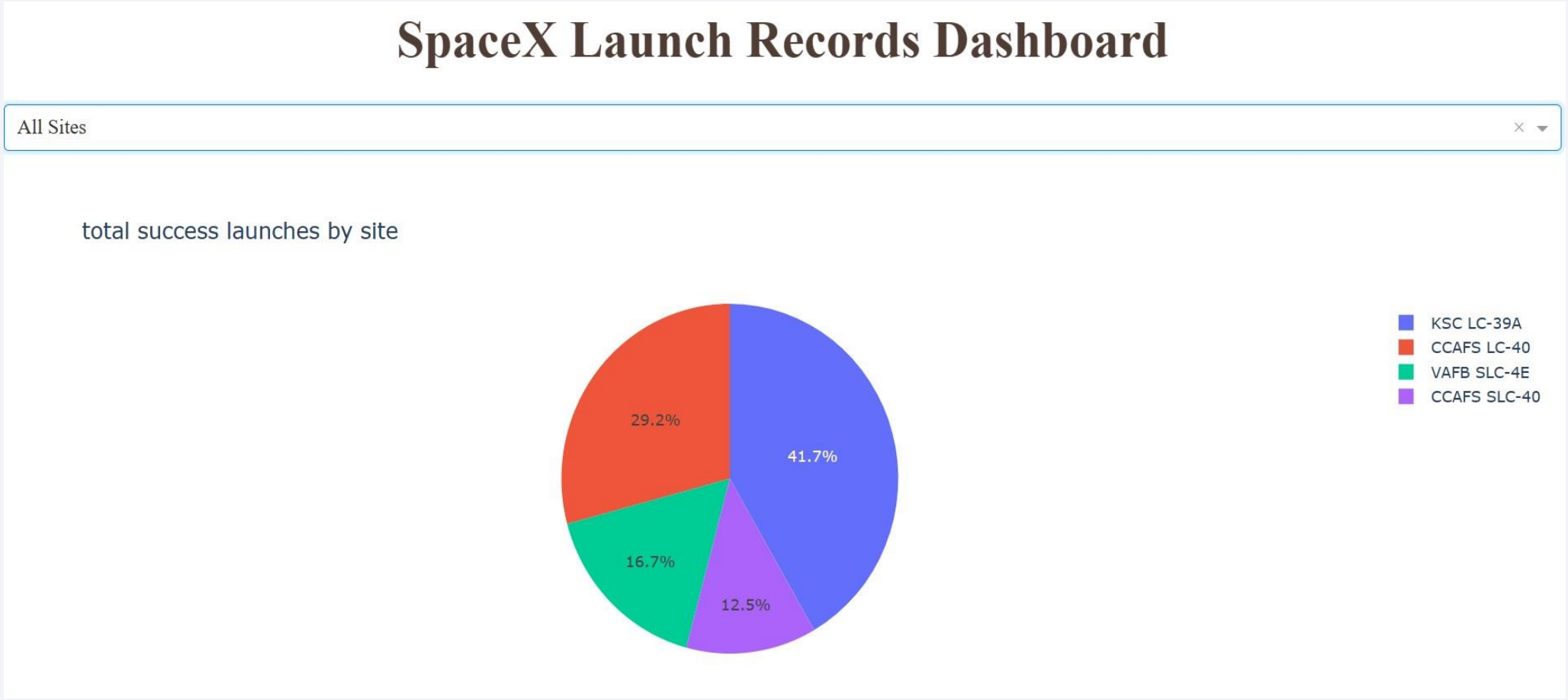
Folium Map: Distances of SpaceX Launch Site to Nearby Locations (City, Highway, Railway, Coastline)



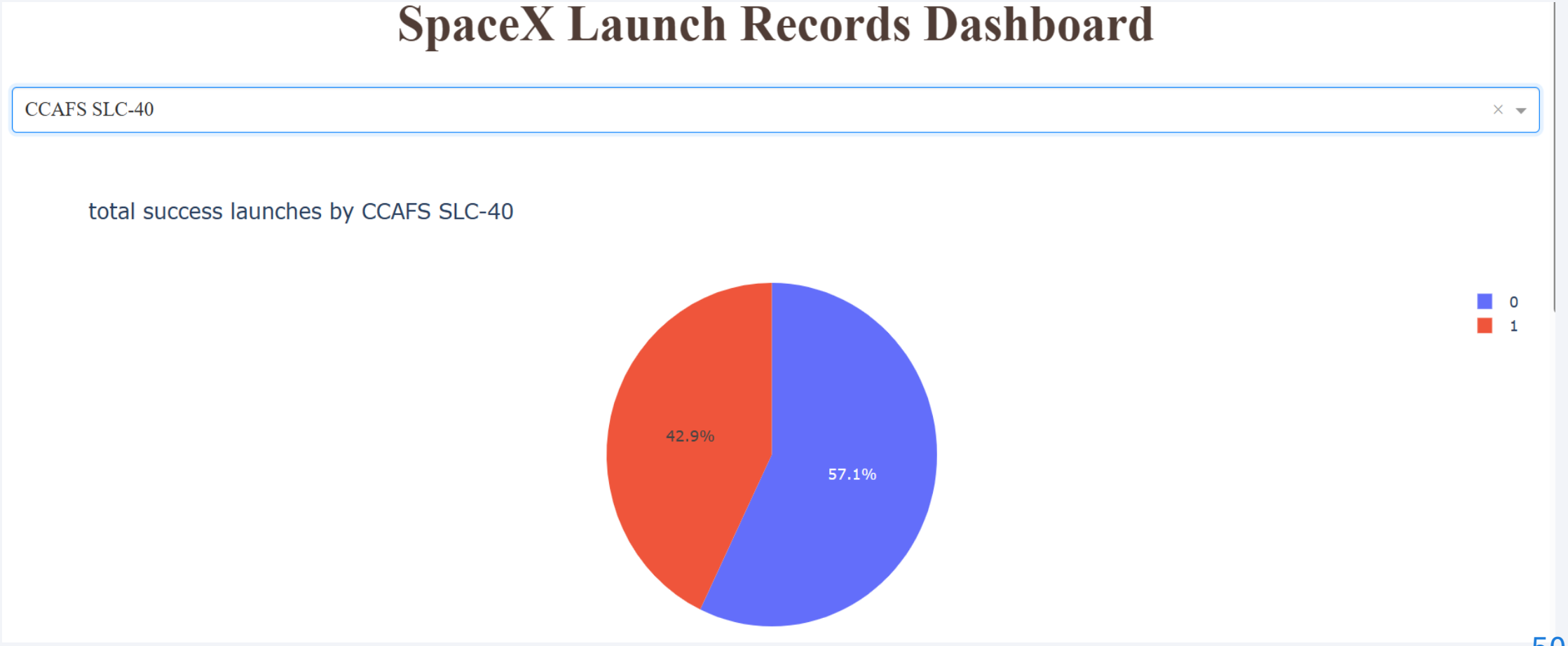
City is not visible here because of large distance from launch site


Section 4
Build a Dashboard with Plotly Dash

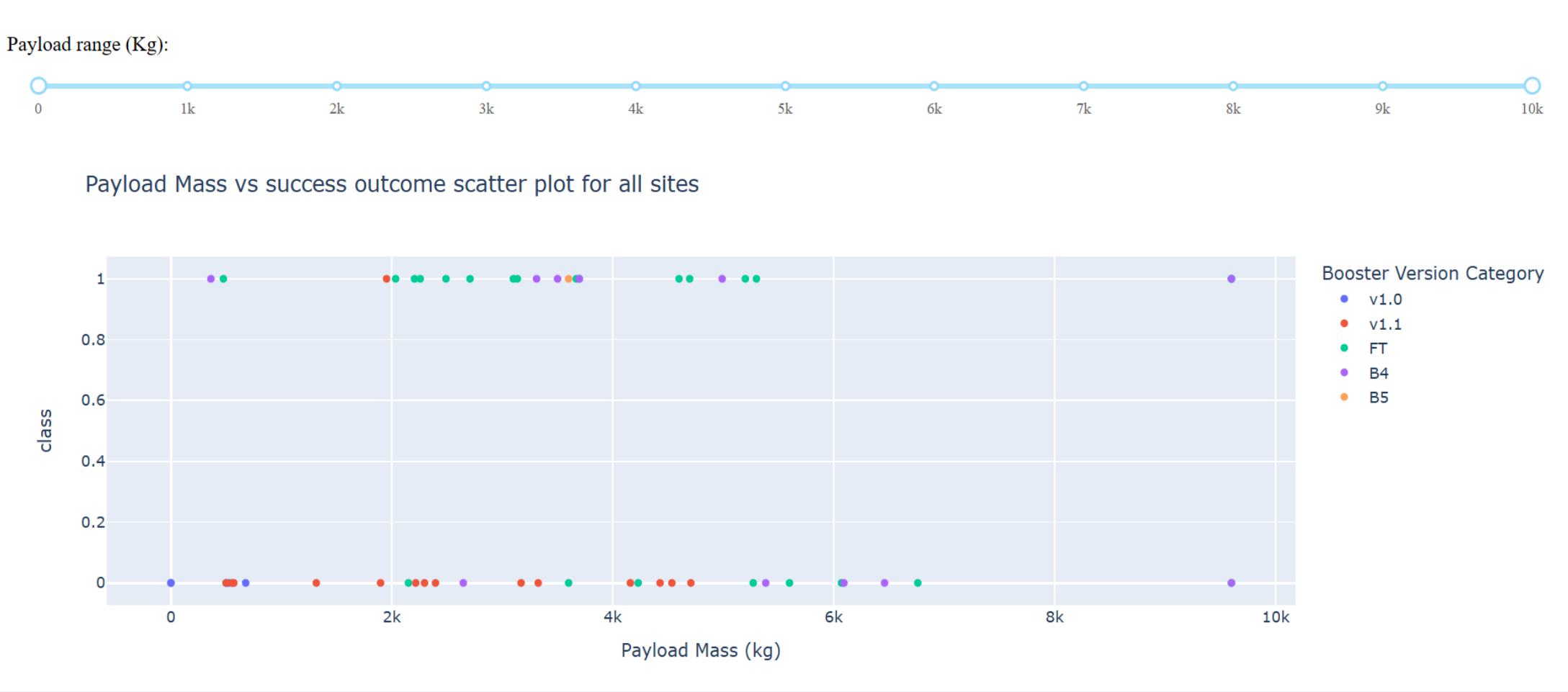
Pie chart for showing success rate of all rocket launches



launch site with highest success rate

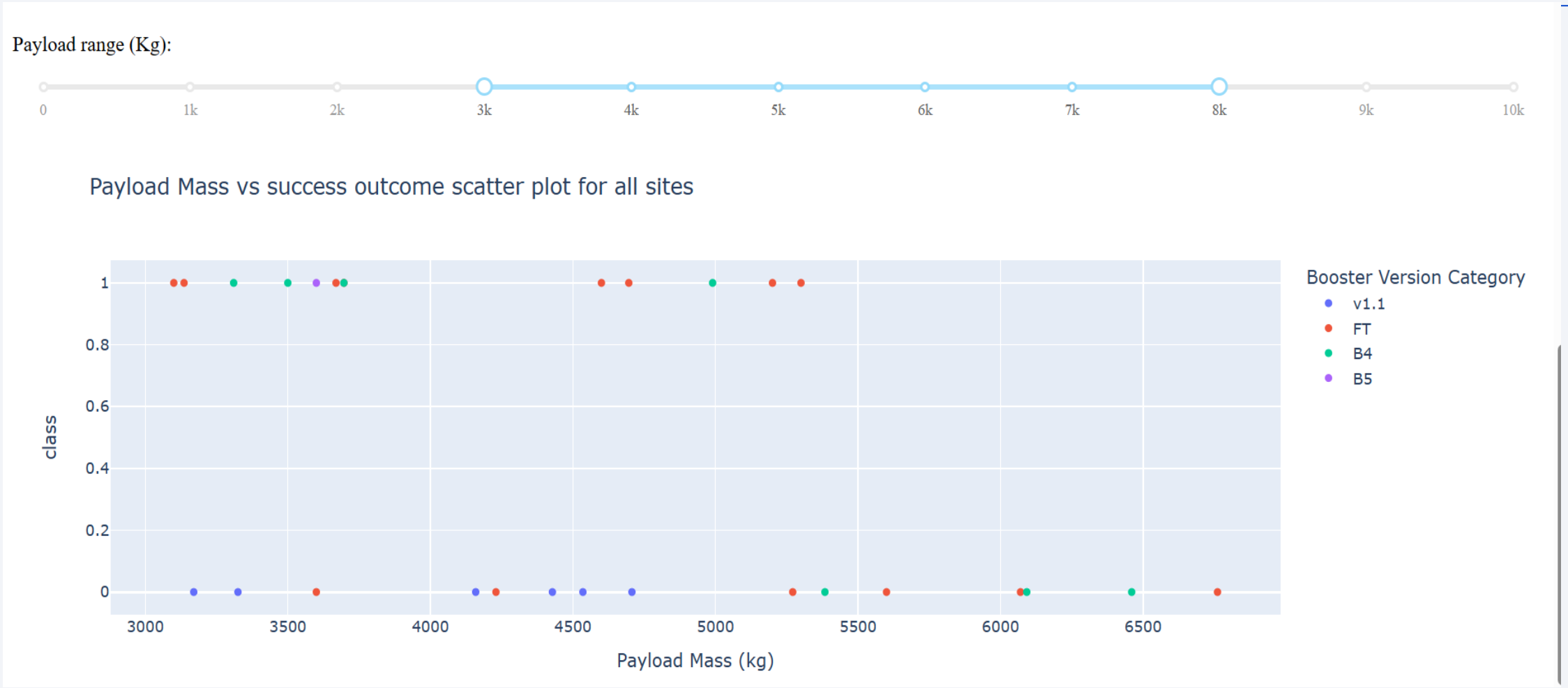


Payload mass vs success outcome chart for all sites (Range [0,10])



We observe that Booster version category FT has the highest success rate when the payload range is between 0 and 10k inclusive.

Payload mass vs success outcome chart for all sites (Range [3,8])

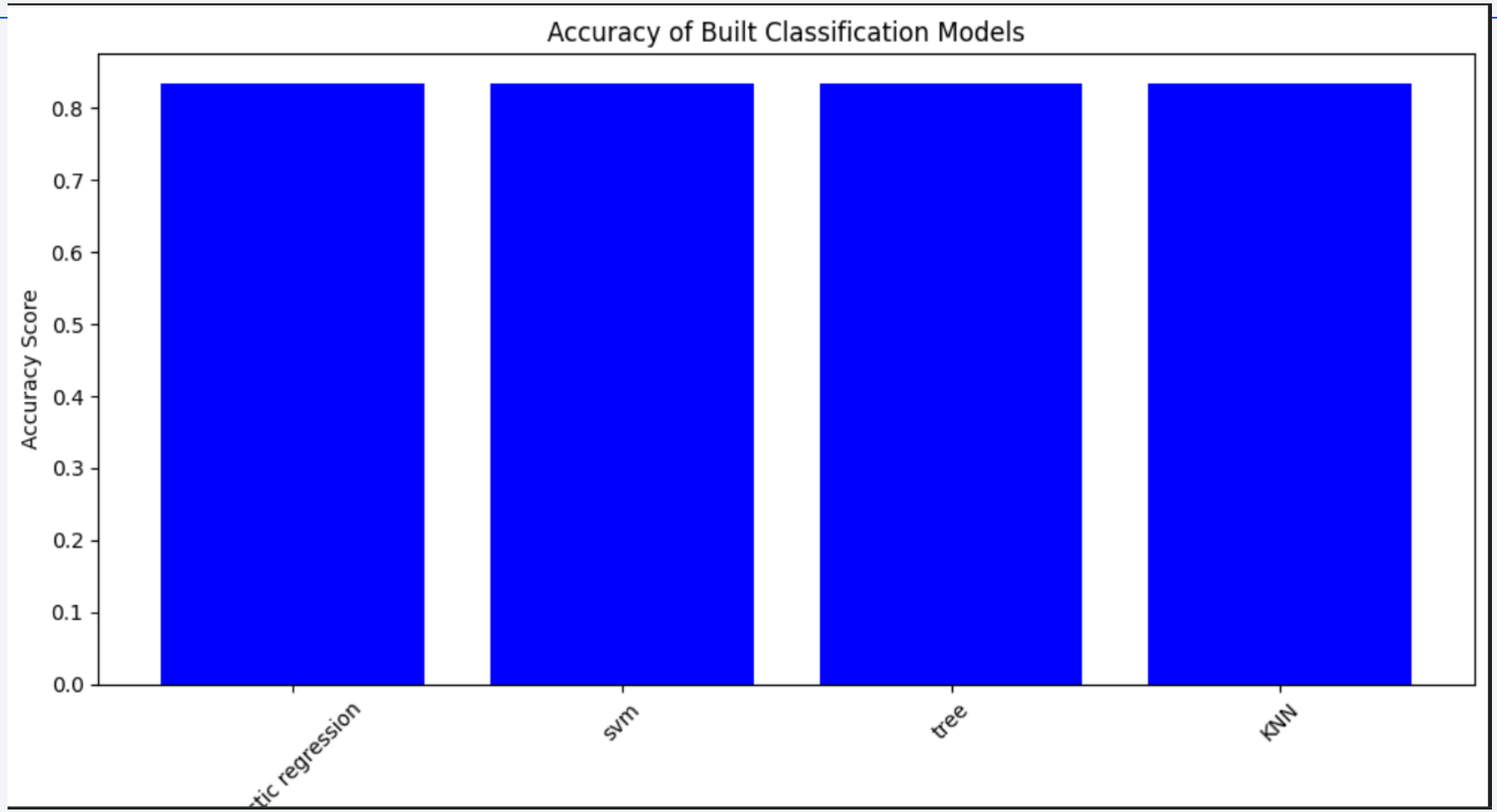


We observe that Booster version category FT has the highest success rate when the payload range is between 0 and 10k inclusive.

Section 5

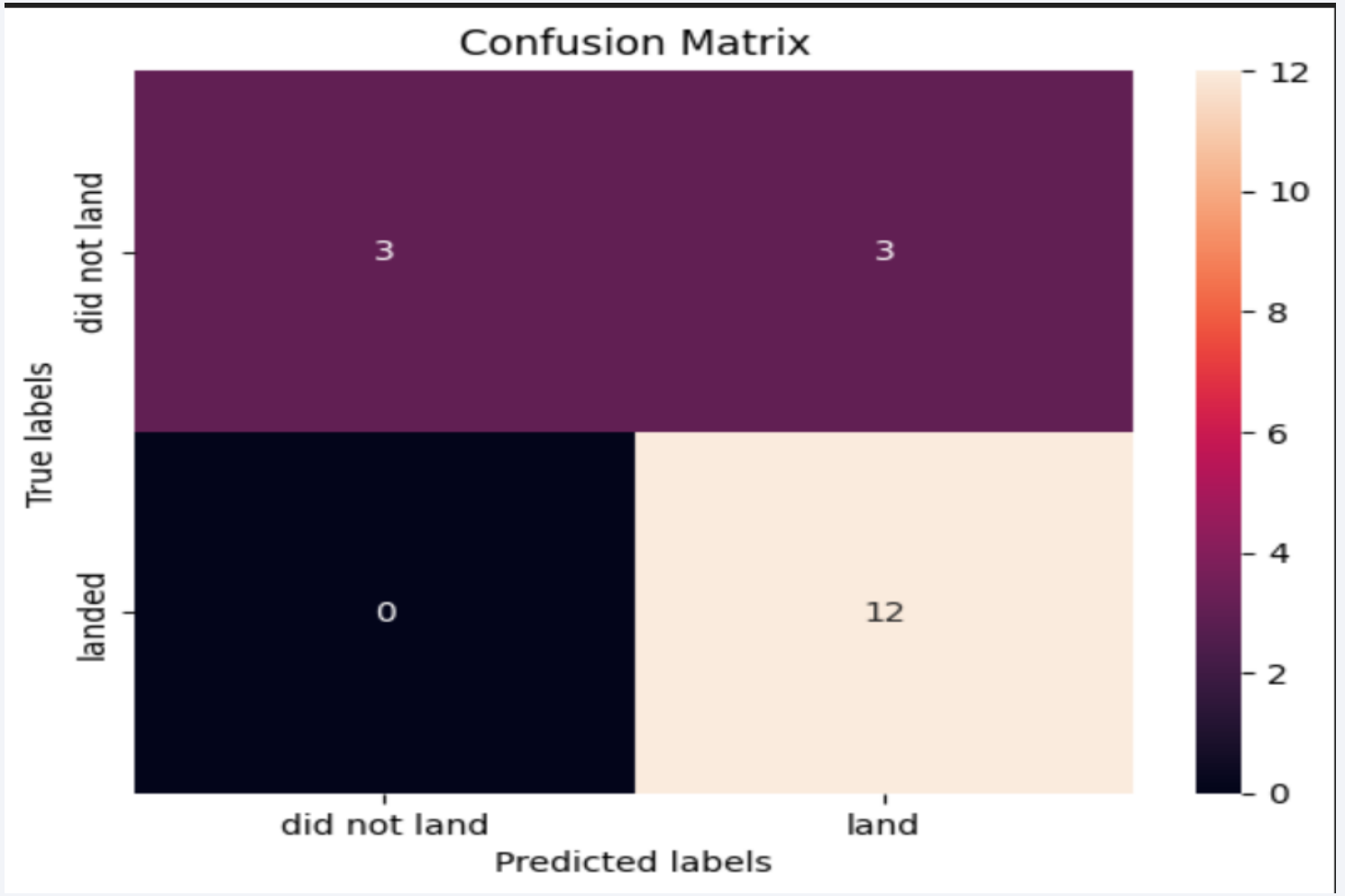
Predictive Analysis (Classification)

Classification Accuracy



All models have same accuracy but for our problem we select decision tree classifier

Confusion Matrix



Conclusions

- We have finally developed a proper machine learning model (decision tree classifier) that will correctly predict if future spaceX Falcon9 rocket launch ‘s first stage will land successfully or not.
- Analyzing launch sites on maps using folium, we see that launch sites are kept far away from city areas but distance from highways, railways and coastlines is not that much
- We observe through dashboards that Booster version category FT has the highest success rate for most launch sites.

Appendix

- Python code for dash url : dash code

Thank you!