IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nguyen Van Duc Long
Aug 24th 2024

Outline

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

Executive Summary

- Methodology
  - Data Collection
  - Data Analysis & Data Wrangling
  - EDA
  - Machine Learning
  - Insights

Introduction

In this capstone, I took the role of a data scientist working for a new rocket company **SpaceY** that would like to compete with **SpaceX** founded by Billionaire industrialist **Allon Mask**.

This project is to determine the price of each launch. It will do this by gathering information about SpaceX and create dashboards for your team. It will also determine if SpaceX will reuse the first stage. Instead of using Rocket Science to determine if the first stage will land successfully, it will train a machine learning model and use public information to predict if SpaceX will reuse the first stage.

Section 1

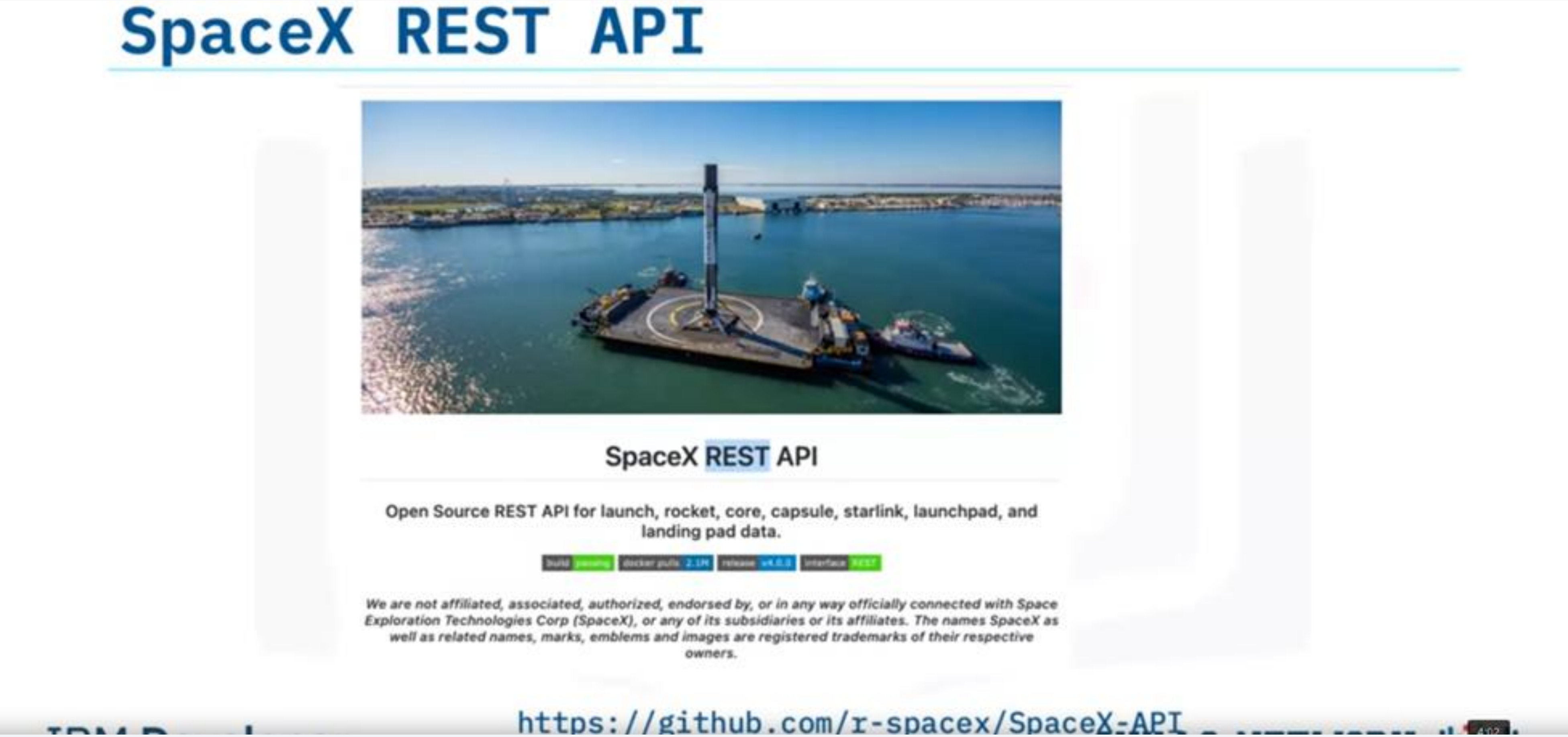
Methodology

Methodology

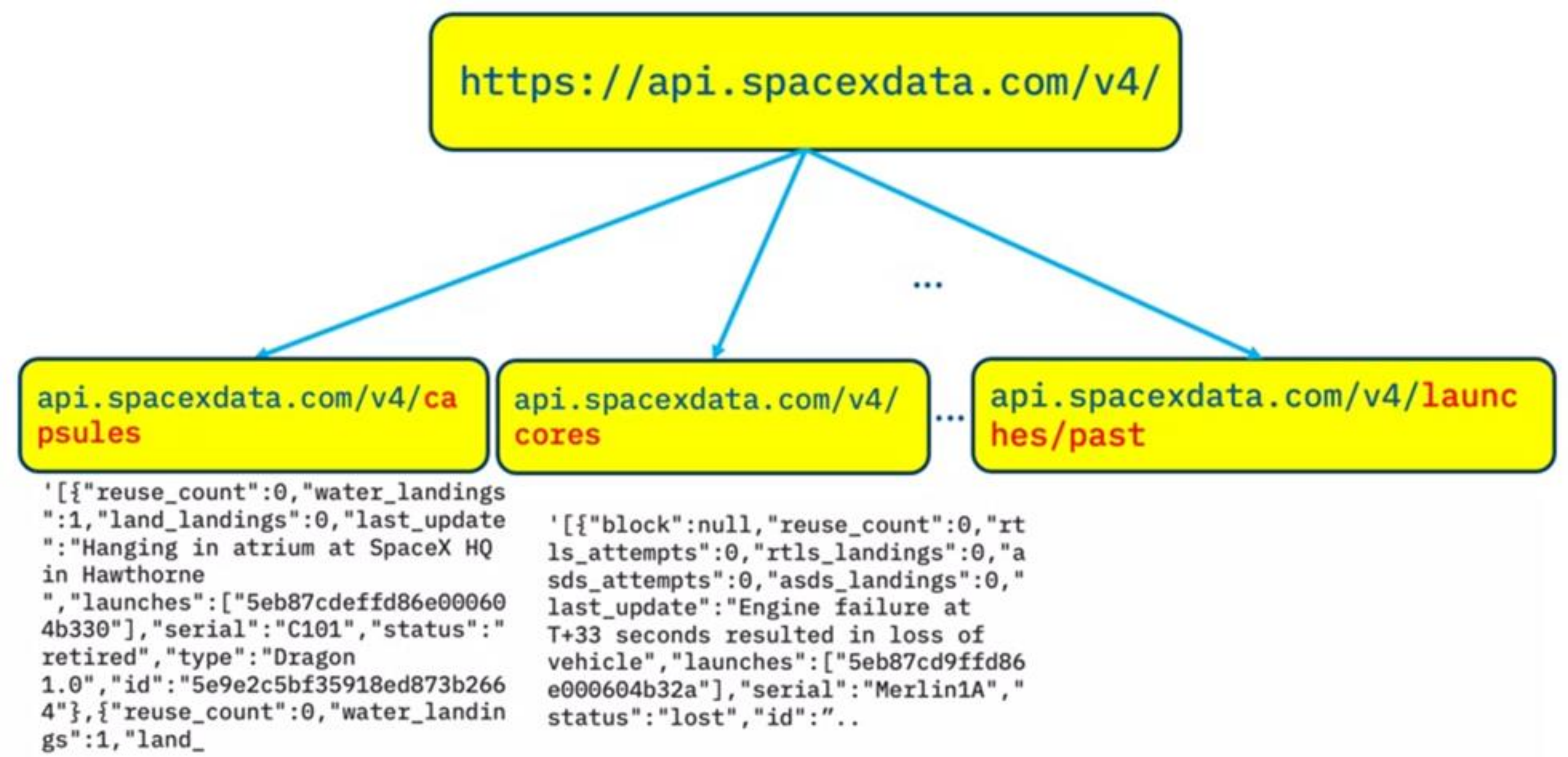
Executive Summary

- Data collection
- Perform data analysis and data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

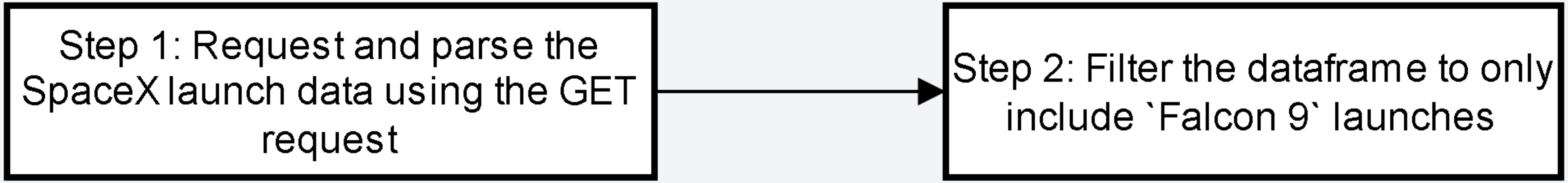
Data Collection



Data Collection

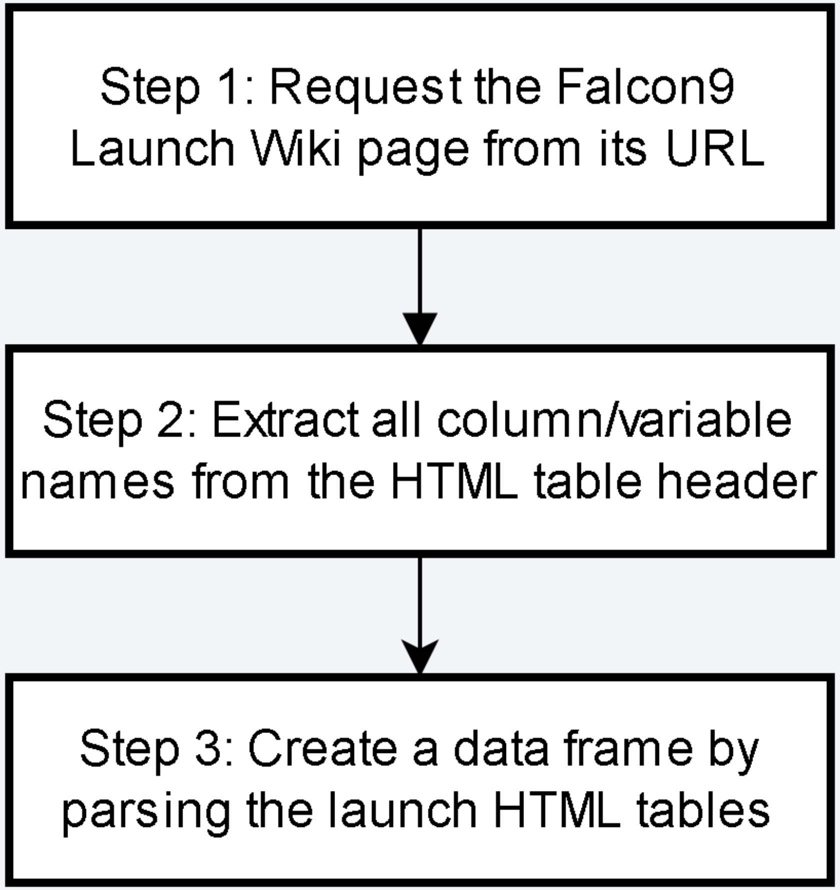


Data Collection – SpaceX API



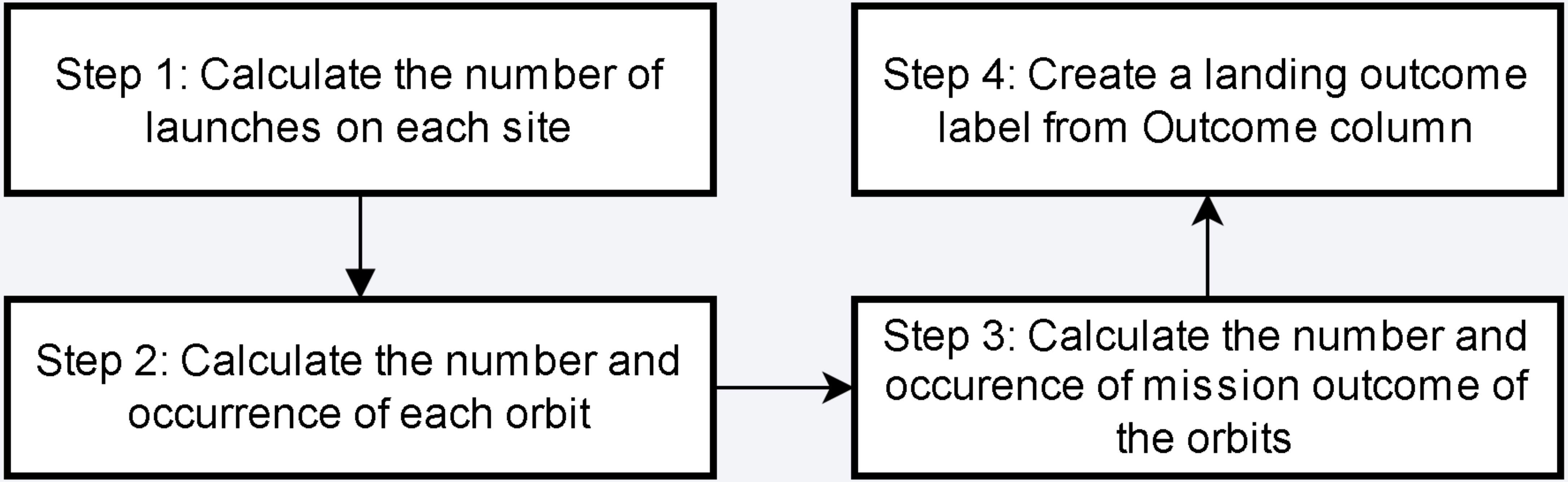
- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/1_jupyter-labs-spacex-data-collection-api.ipynb

Data Collection – Scraping



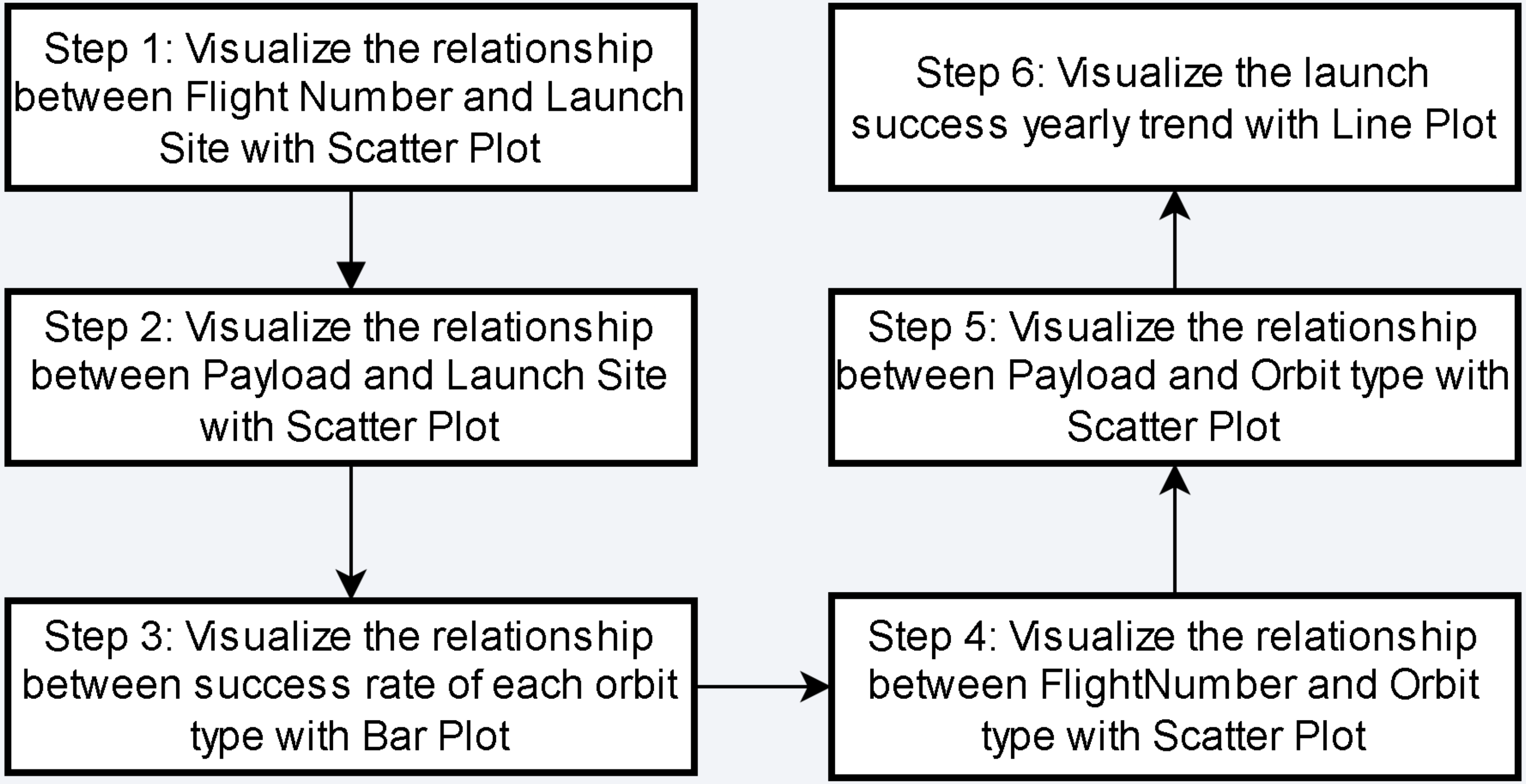
- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/2_jupyter-labs-webscraping.ipynb

Data Analysis and Data Wrangling



- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/3_jupyter-spacex-Data_wrangling.ipynb

EDA with Data Visualization



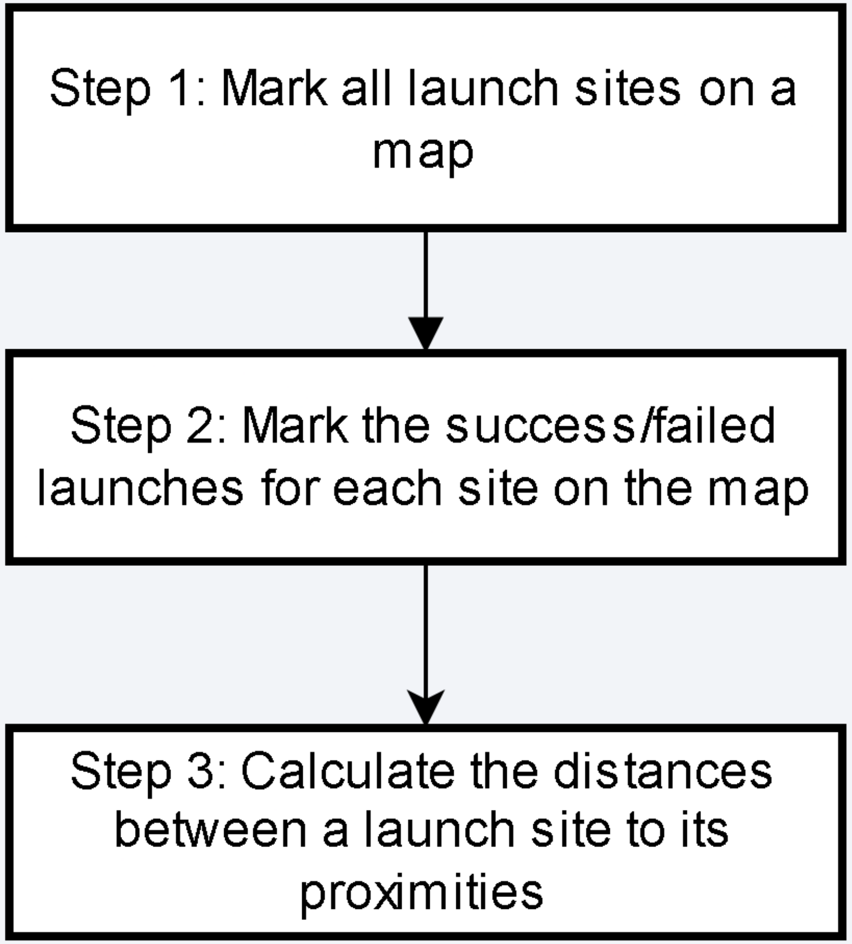
- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/5_jupyter-labs-eda-dataviz.ipynb

EDA with SQL

Step 1: Explore Launch Sites and Missions	Step 2: Analyze Payload Mass and Booster Versions	Step 3: Evaluate Mission Outcomes
• Display the names of the unique launch sites in the space mission. • Display 5 records where launch sites begin with the string 'CCA'.	• Display the total payload mass carried by boosters launched by NASA (CRS). • Display the average payload mass carried by booster version F9 v1.1. • List the names of the boosters that succeeded in drone ship landings and carried a payload mass between 4000 and 6000. • List the names of the booster versions that carried the maximum payload mass using a subquery.	• List the total number of successful and failed mission outcomes. • List the date when the first successful landing outcome on a ground pad was achieved. • List the records displaying month names, failed landing outcomes in drone ships, booster versions, and launch sites for the months in the year 2015. • Rank the count of landing outcomes (e.g., Failure on drone ship or Success on ground pad) between the dates 2010-06-04 and 2017-03-20, in descending order.

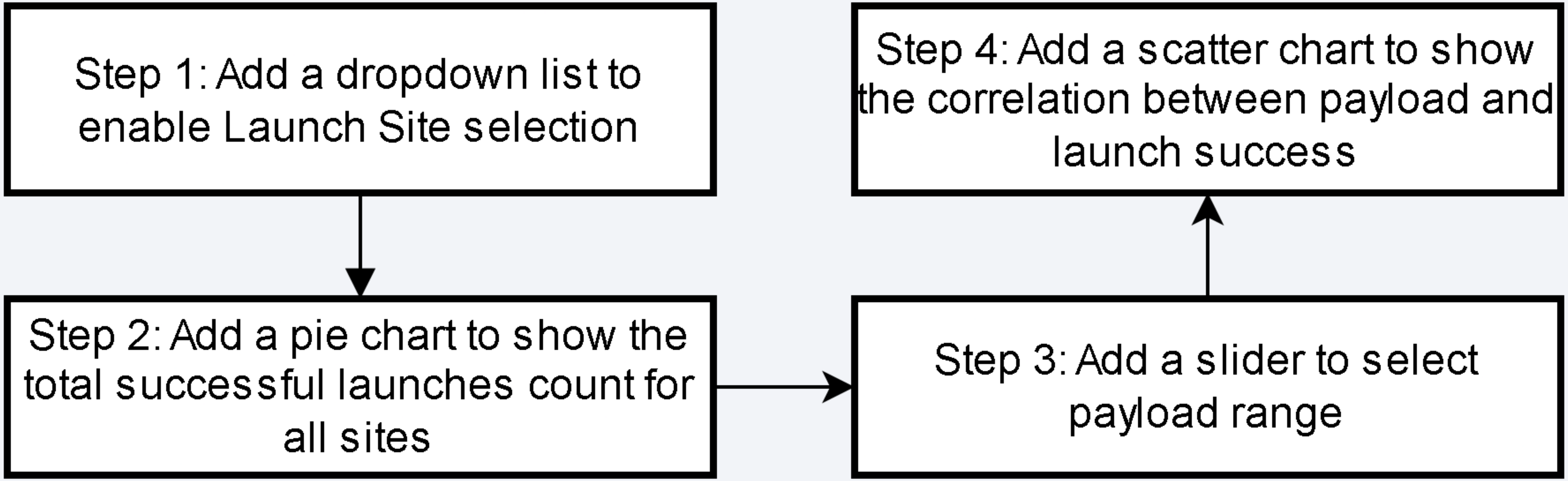
- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/4_jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium



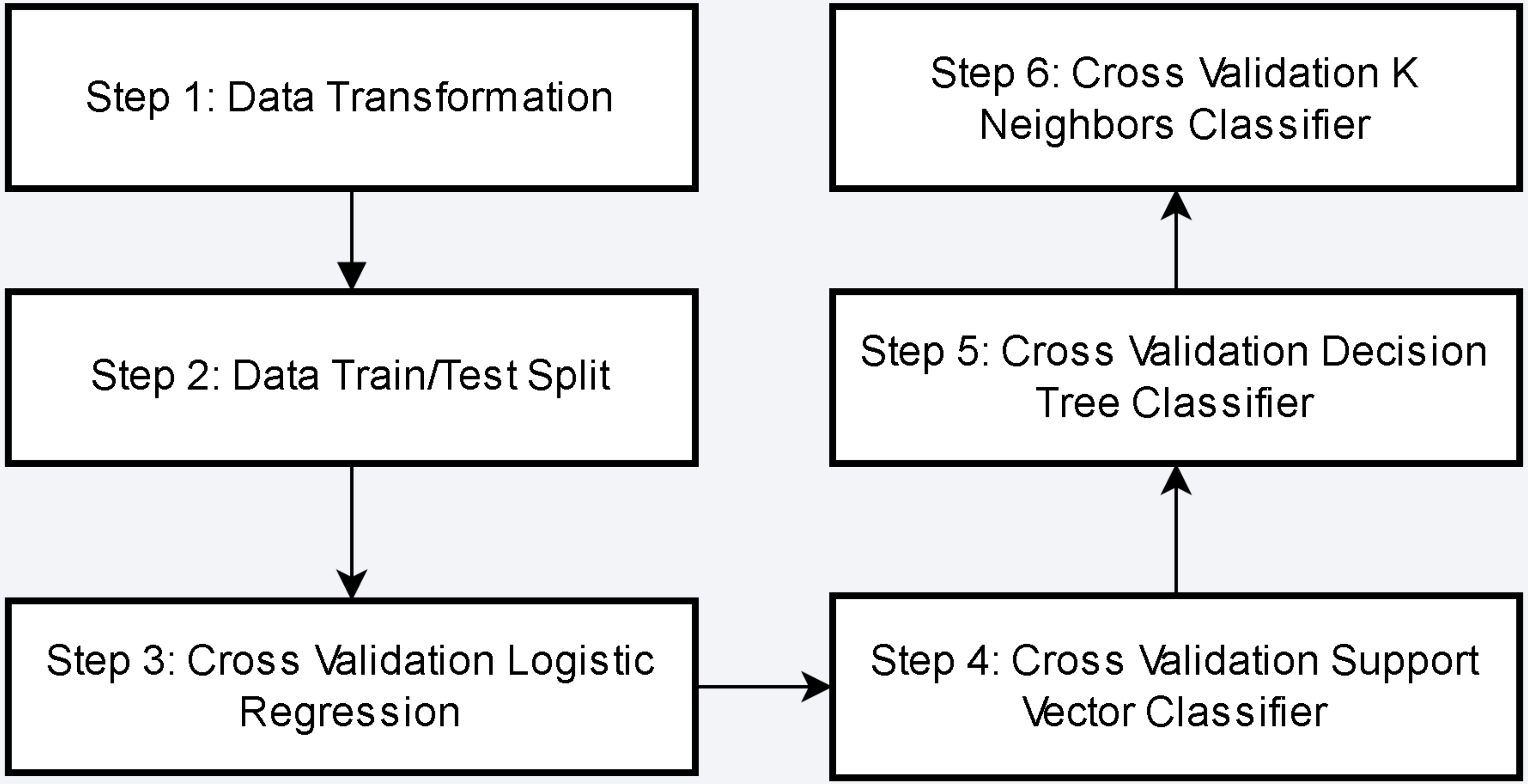
- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/6_lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash



- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/7_dash_interactivity.py

Predictive Analysis (Classification)



- Github URL: https://github.com/longnguyencbct/IBM-DS-Course-Repo/blob/main/Applied_Data_Science_Capstone/8_SpaceX_Machine_Learning_Prediction.ipynb

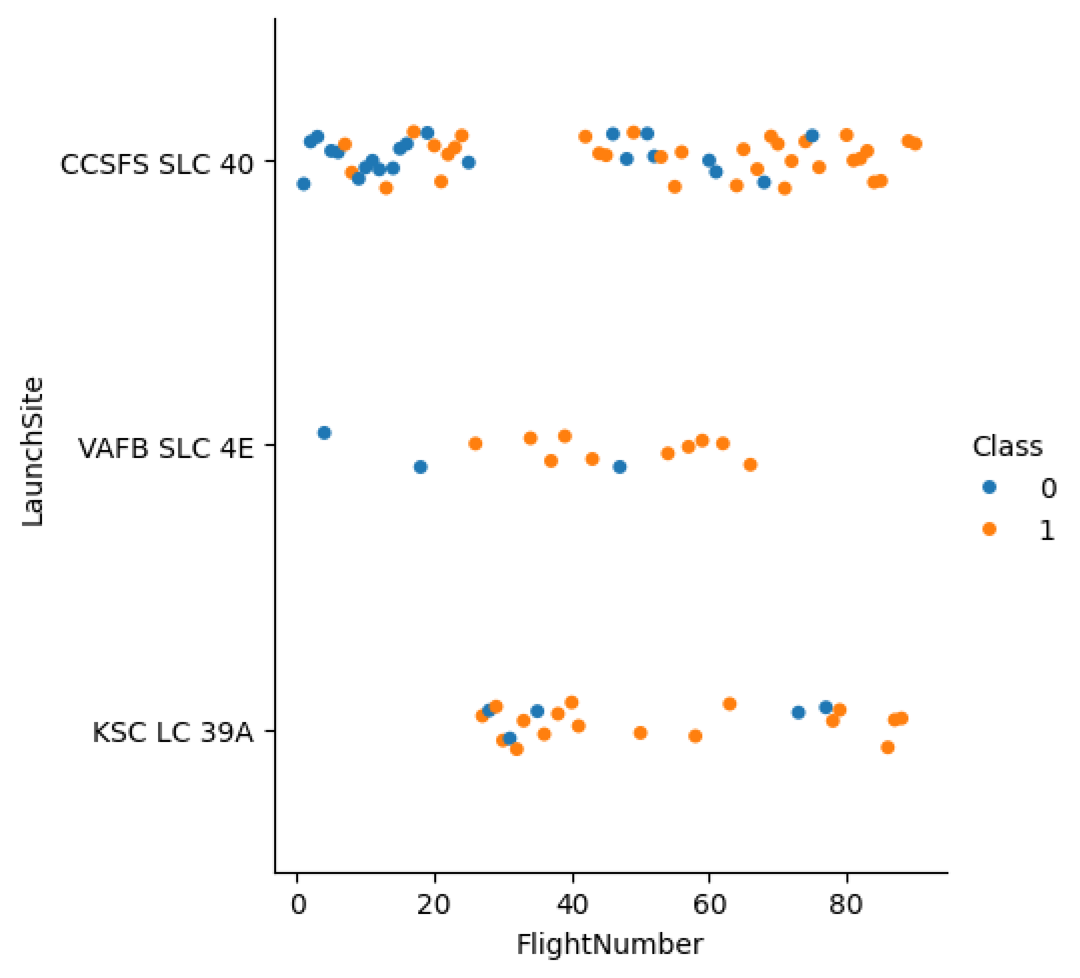
Results

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

Section 2

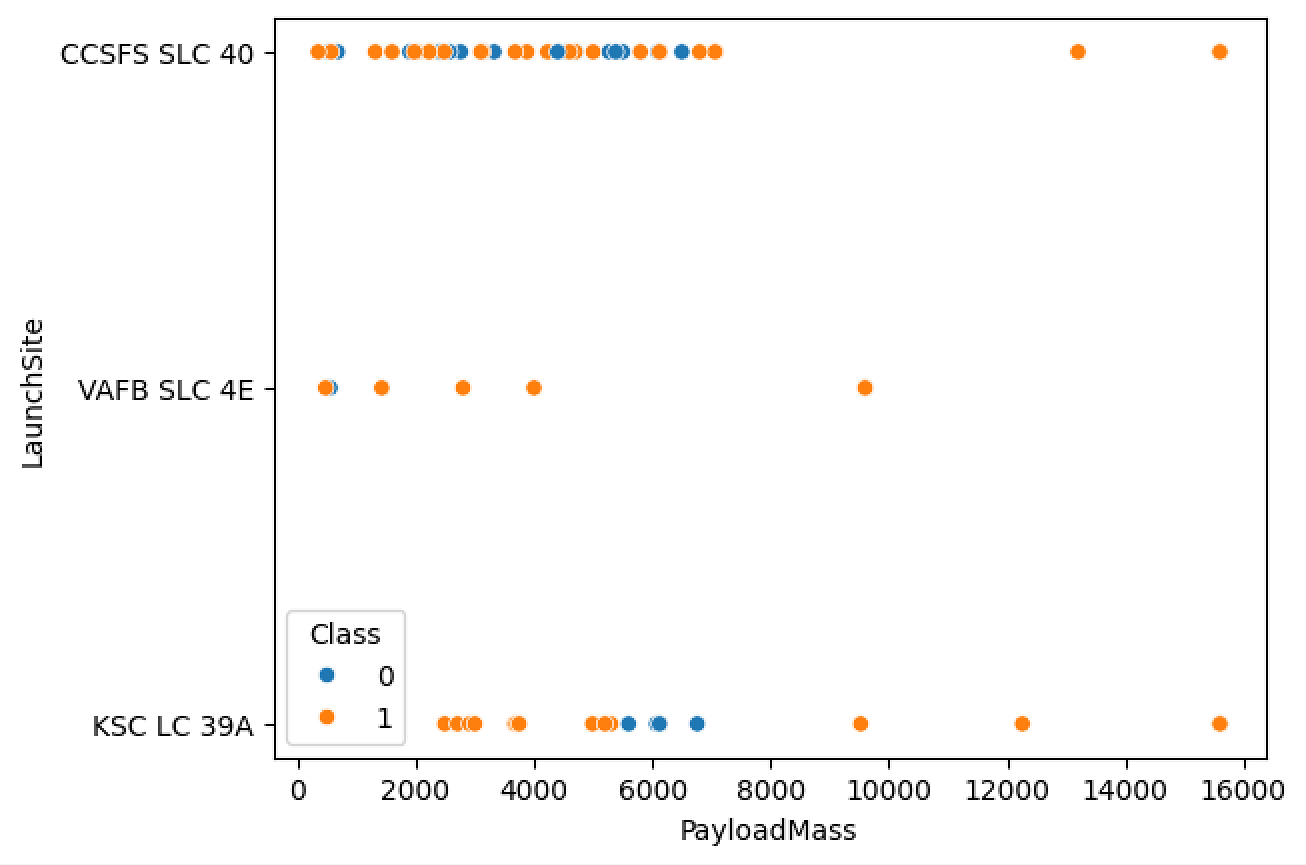
Insights drawn from EDA

Results – EDA: Visualize the relationship between Flight Number and Launch Site



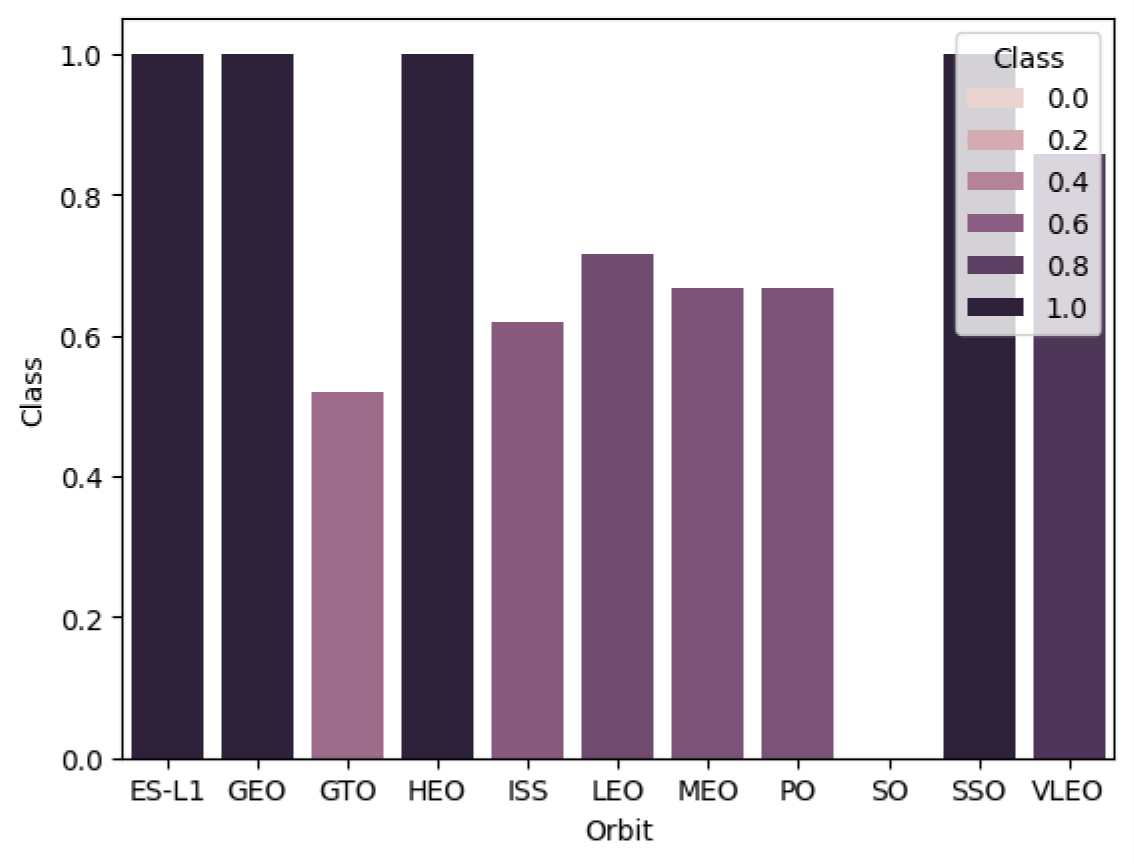
Insight: VAFB SLC 4E and KSC LC 39A Launch Sites have high success rates

Results – EDA:
Visualize the relationship between Payload and Launch Site



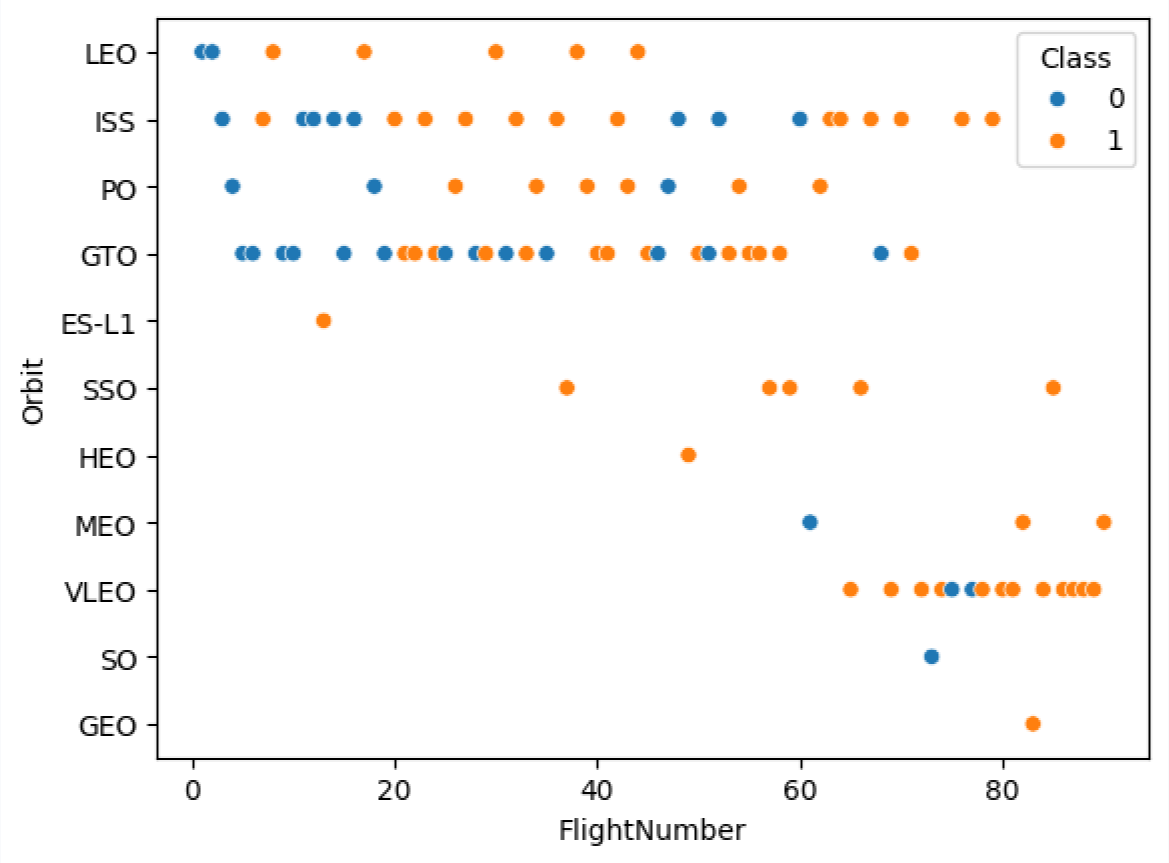
Insight: Very high success rate when Payload Mass is between [2000,5000]. Payload Mass above 8000 also has high success rate, but low statistical significance.

Results – EDA:
Visualize the relationship between success rate of each orbit type



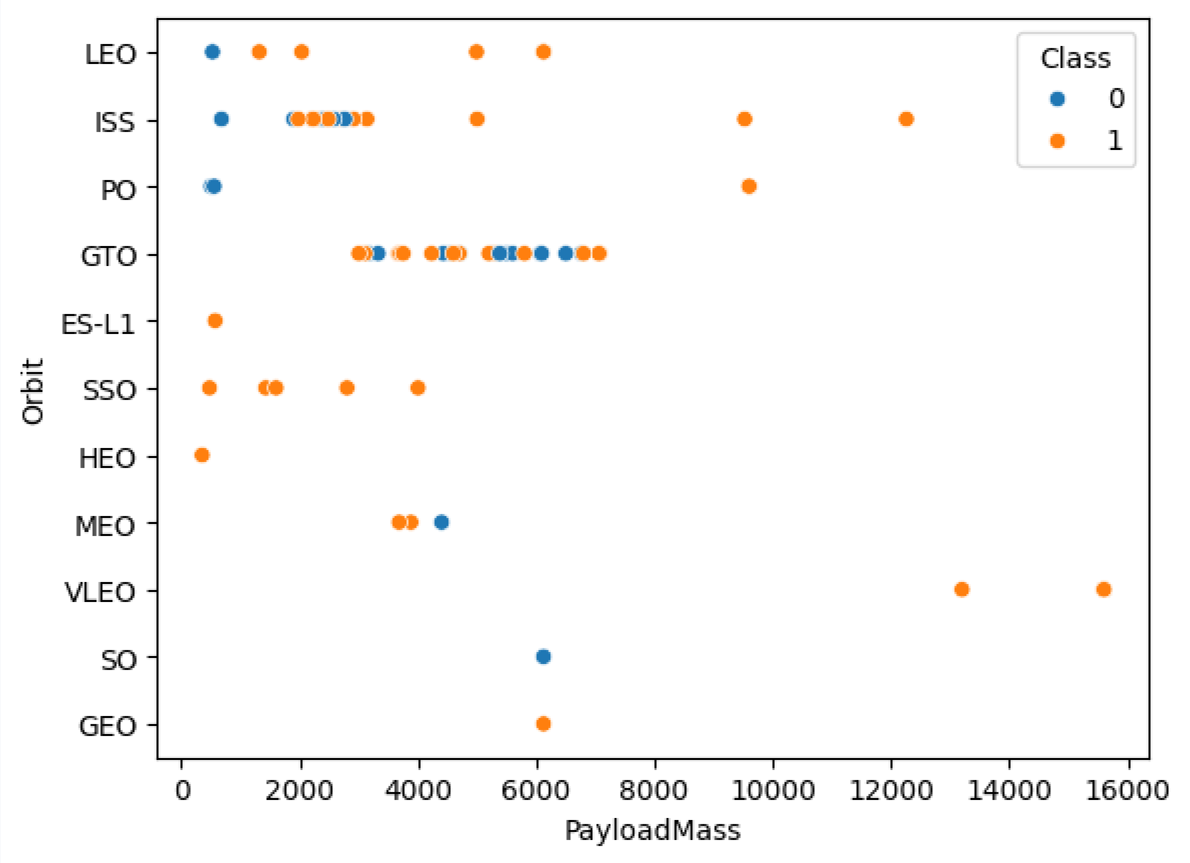
Insight: ES-L1, GEO, HEO, SSO, VLEO Orbits has high success rate. But this has not considered statistical significance.

Results – EDA:
Visualize the relationship between Flight Number and Orbit type



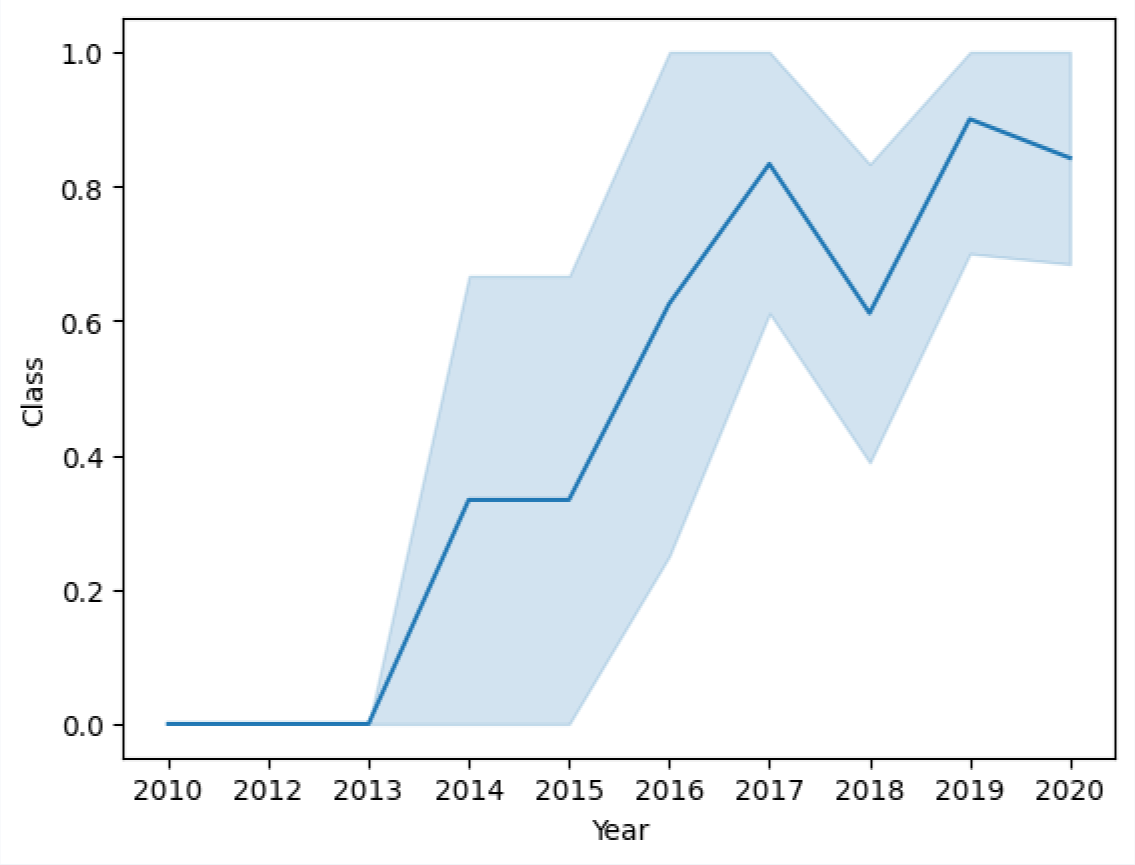
Insight: LEO Orbit option has improved success rate over time.

Results – EDA: Visualize the relationship between Payload and Orbit type



Insight: With heavy payloads the successful landing or positive landing rate are more for PO, 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.

Results – EDA:
Visualize the launch success yearly trend



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

All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

```
%sql Select distinct Launch_Site from SPACEXTBL
```

[12] Python

```
 * 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'

Task 2

Display 5 records where launch sites begin with the string 'CCA'

[17]

```
%sql select * from spacextbl where Launch_Site like "CCA%" limit 5
```

Python

```
 * sqlite:///my_data1.db
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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

Task 3

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

```
%sql select sum(PAYLOAD_MASS__KG_) from spacextbl where Customer = "NASA (CRS)"
```

[18]

Python

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

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

Task 4

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

```
%sql select avg(PAYLOAD_MASS__KG_) from spacextbl where Booster_Version = "F9 v1.1"
```

[26] Python

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql select min(Date) from spacextbl where Landing_Outcome = "Success (ground pad)"
```

[41] Python

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

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

[46]

```
%sql select Payload from spacextbl where Landing_Outcome = "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

Python

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

Payload
JCSAT-14
JCSAT-16
SES-10
SES-11 / EchoStar 105

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

[47]

```
%sql select count(*) from spacextbl where Landing_Outcome like "Succ%" or Landing_Outcome like "Fail%"
```

Python

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

count(*)
71

Boosters Carried Maximum Payload

Task 8

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

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

[53] Python

```
 * 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

Task 9

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

[69]

```
%sql select substr(Date,6,2) as Month,Booster_Version,Launch_Site, Landing_Outcome from spacextbl where substr(Date,0,5)='2015' and Landing_Outcome = "Failure (drone ship)"
```

Python

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

Month	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

Task 10

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.

[81]

```
%sql select Landing_Outcome, count(*) as Count from spacextbl where Date between "2010-06-04" and "2017-03-20" group by Landing_Outcome order by Date desc
```

Python

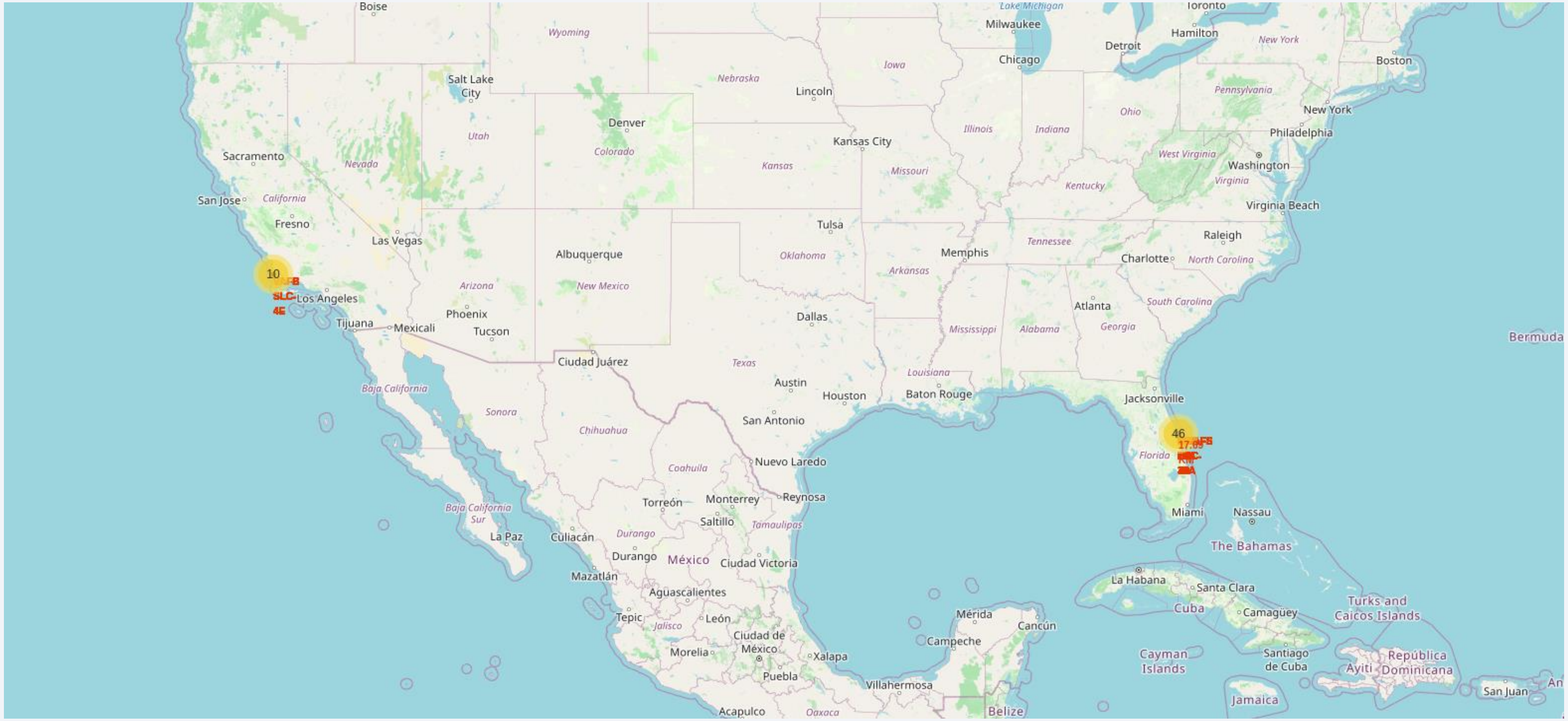
```
 * sqlite:///my_data1.db
Done.
```

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

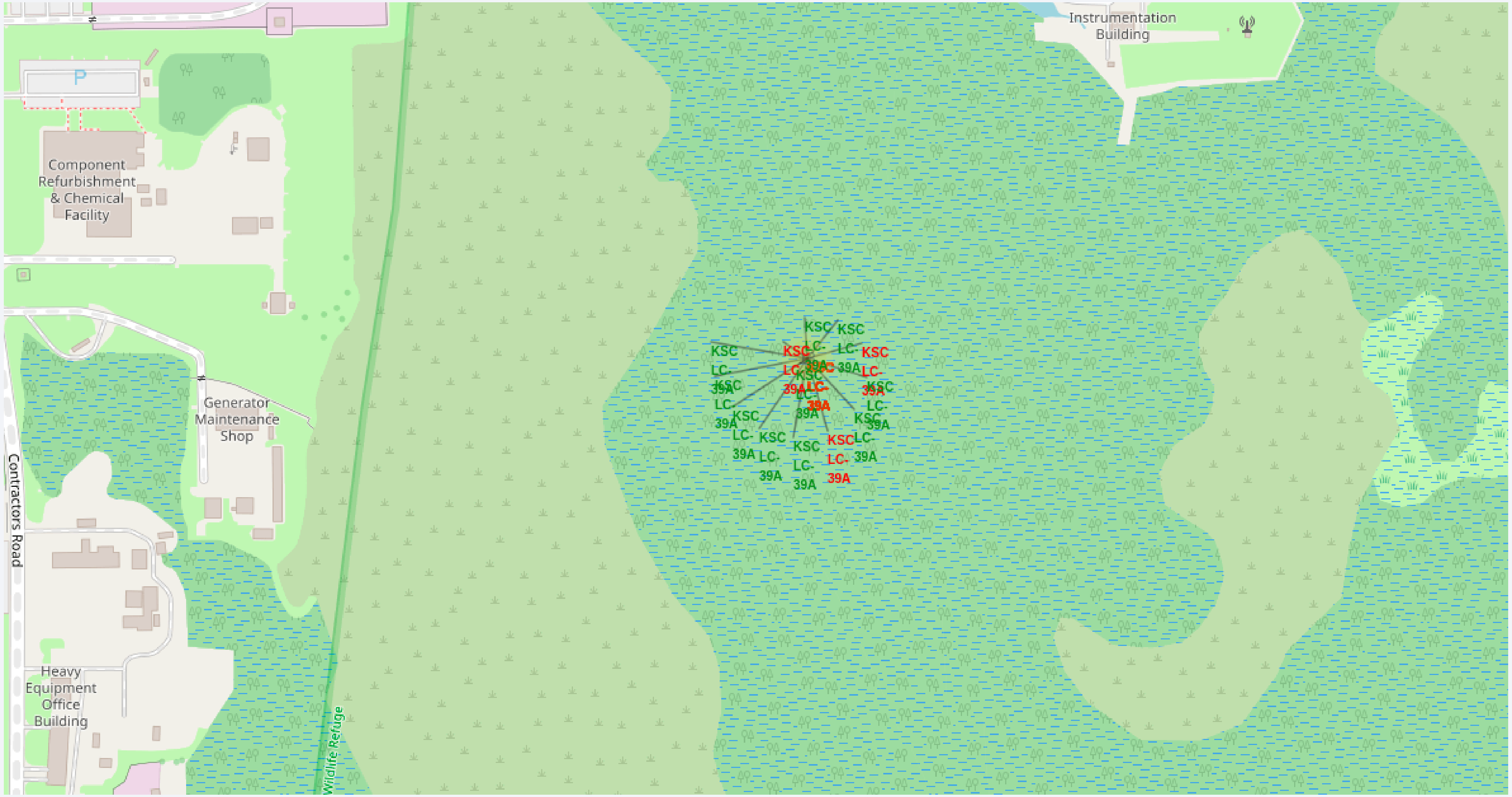
Section 3

Launch Sites Proximities Analysis

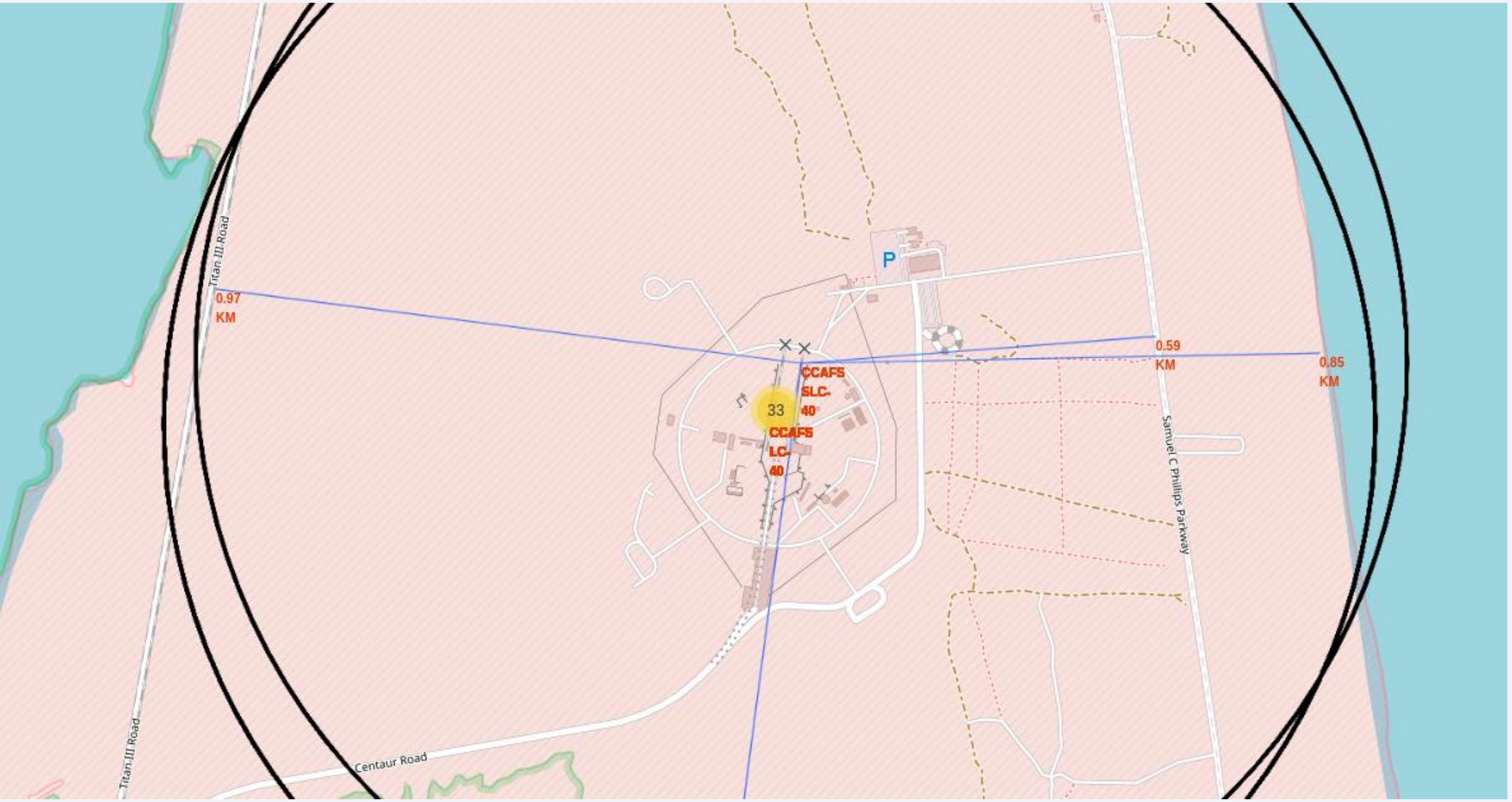
Result – Folium Map: All launch sites



Result – Folium Map: Launch Outcomes

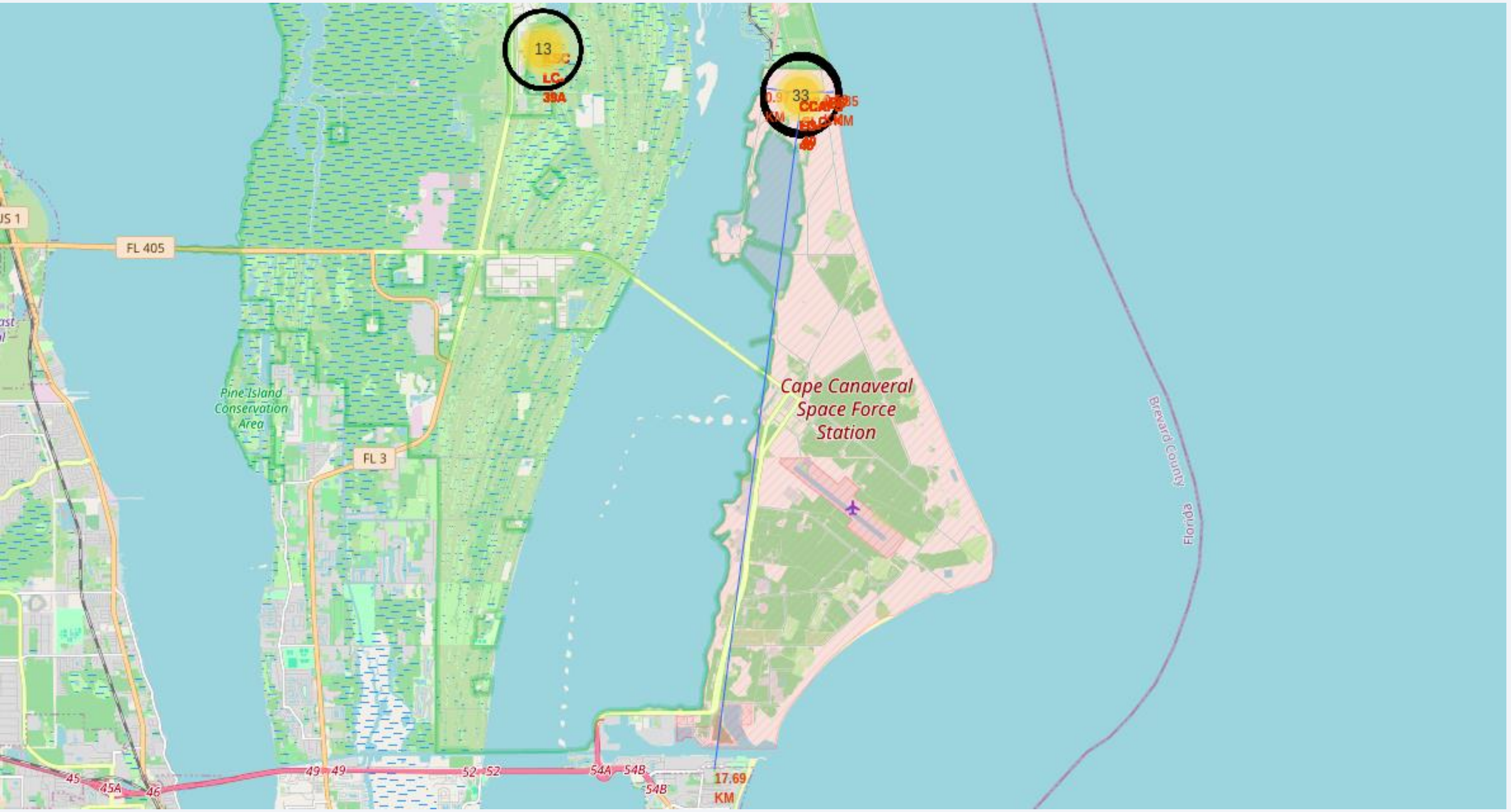


Result – Folium Map: Distance from proximities



- Distance from nearest railway: 0.97KM.
- Distance from nearest highway: 0.59KM.
- Distance from nearest coastline: 0.85 KM.

Result – Folium Map: Distance from proximities

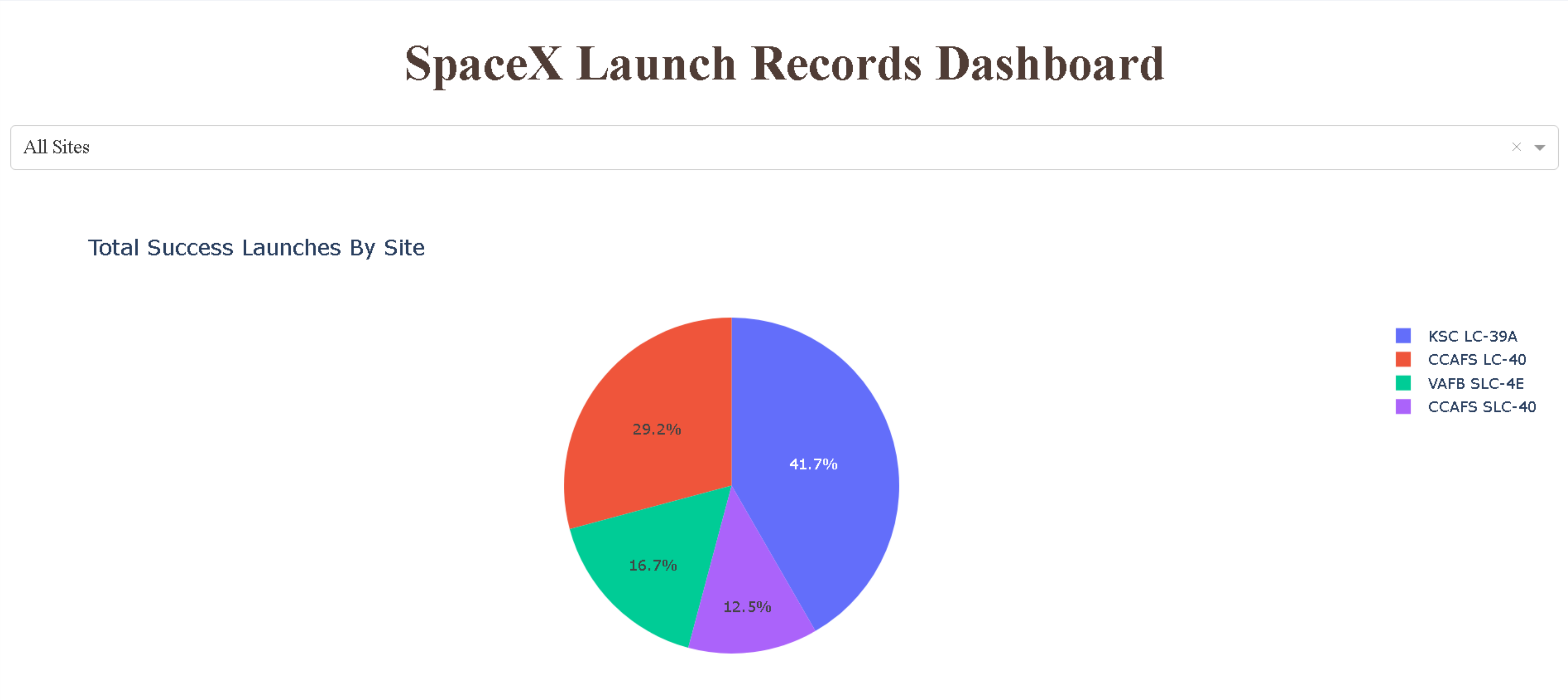


- .Distance from nearest city: 17.69 KM.

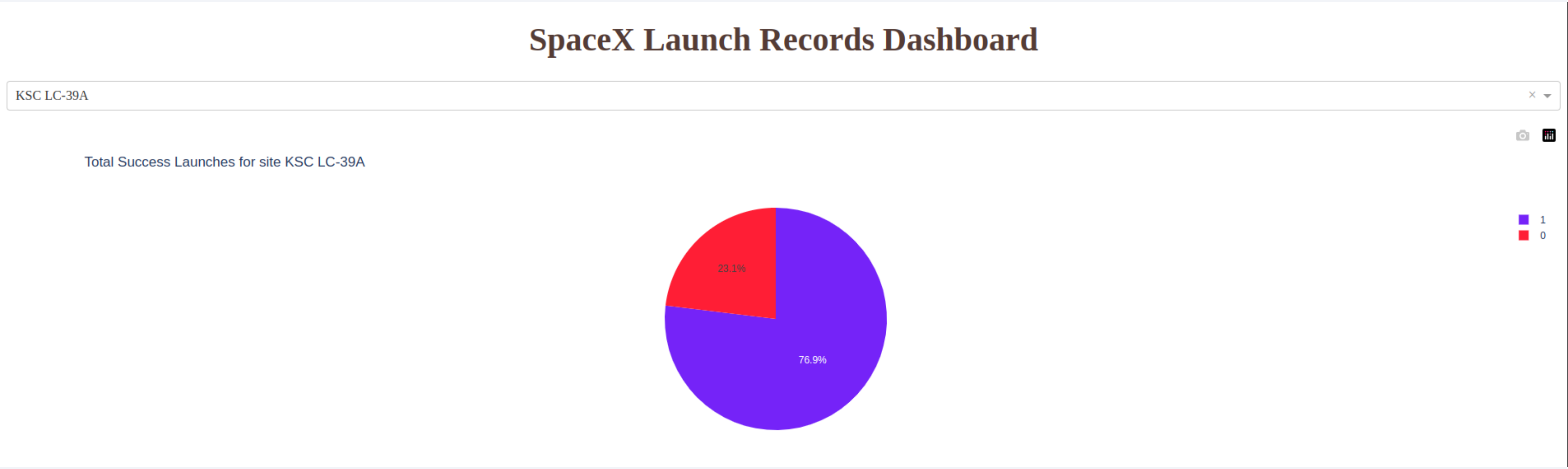
Section 4

Build a Dashboard with Plotly Dash

Results – Interactive: Pie Chart



Results – Interactive: Highest success rate site pie chart



Results – Interactive: Slider & Scatter Plot

Section 5

Predictive Analysis (Classification)

Results – Predictive Analysis

In-Sample (80%): 10-Fold Cross Validation Out-of-Sample (20%): Retest using best hyperparameters			
Logistic Regression	**Support Vector Classifier**	**Decision Tree Classifier**	**K Neighbors Classifier**
• Train Score: 0.8464 • Test Score: 0.8333 • TP: 12 • FN: 0 • FP: 3 • TN: 3	• Train Score: 0.8482 • Test Score: 0.8333 • TP: 12 • FN: 0 • FP: 3 • TN: 3	• Train Score: 0.8768 • Test Score: 0.6666 • TP: 9 • FN: 3 • FP: 3 • TN: 3	• Train Score: 0.8482 • Test Score: 0.8333 • TP: 12 • FN: 0 • FP: 3 • TN: 3

Conclusions

- **Launch Success**: High success rates are linked to specific launch sites and optimal payload ranges (2000-5000 kg).
- **Orbit Types**: Certain orbits like ES-L1 and GEO show higher successes
- **Predictive Analysis:** 3 out of 4 machine learning models effectively predict SpaceX's first stage reuse, demonstrating valuable forecasting potential.

The insights gained from this analysis can help SpaceY optimize its launch strategies, focusing on more successful launch sites and payload configurations. By leveraging similar machine learning models, SpaceY could improve its predictive capabilities, thereby enhancing decision-making and operational efficiency.

Appendix

- Decision Tree Classifier gives different scores when re-running the jupyter notebook.
- Besides Decision Tree Classifier, every other models are equally suitable for landing success prediction with test precision of 0.8333

Thank you!