IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Collect data using API or web scraping with BeautifulSoup.

. Exploratory Data analysis

- Visualization
- Modeling (build, evaluate, and refine predictive models for discovering more exiting insights)

Introduction

- Companies like Virgin Galactic, Rocket Lab, Blue Origin, SpaceX are making space travel affordable.
- SpaceX sends spacecraft to the international Space Station du to inexpensive rocket lunches.
- In this studying we want to getter better understanding of the Falcon 9 launch.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - using an API, specifically with the SpaceX REST API.
  - Using the Python BeautifulSoup package to web scrape some HTML table that contain valuable Falcon 9 launch.
- Perform data wrangling
  - Exploit deeply the data and try to describe the meaning of each attribute.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Preprocessing, train_test_split, Grid Search, accuracy, confusion matrix.

Data Collection

- Data sets were collected by using:
  - ➢ using an API, specifically with the SpaceX REST API.
  - ➢ Using the Python BeautifulSoup package to web scrape some HTML table that contain valuable Falcon 9 launch.

- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- GitHub URL of the completed SpaceX API calls notebook: test/jupyter-labs-spacex-data-collection-api.ipynb at main · mouhamedlaminembaye/test (github.com)

SpaceX REST API URL:

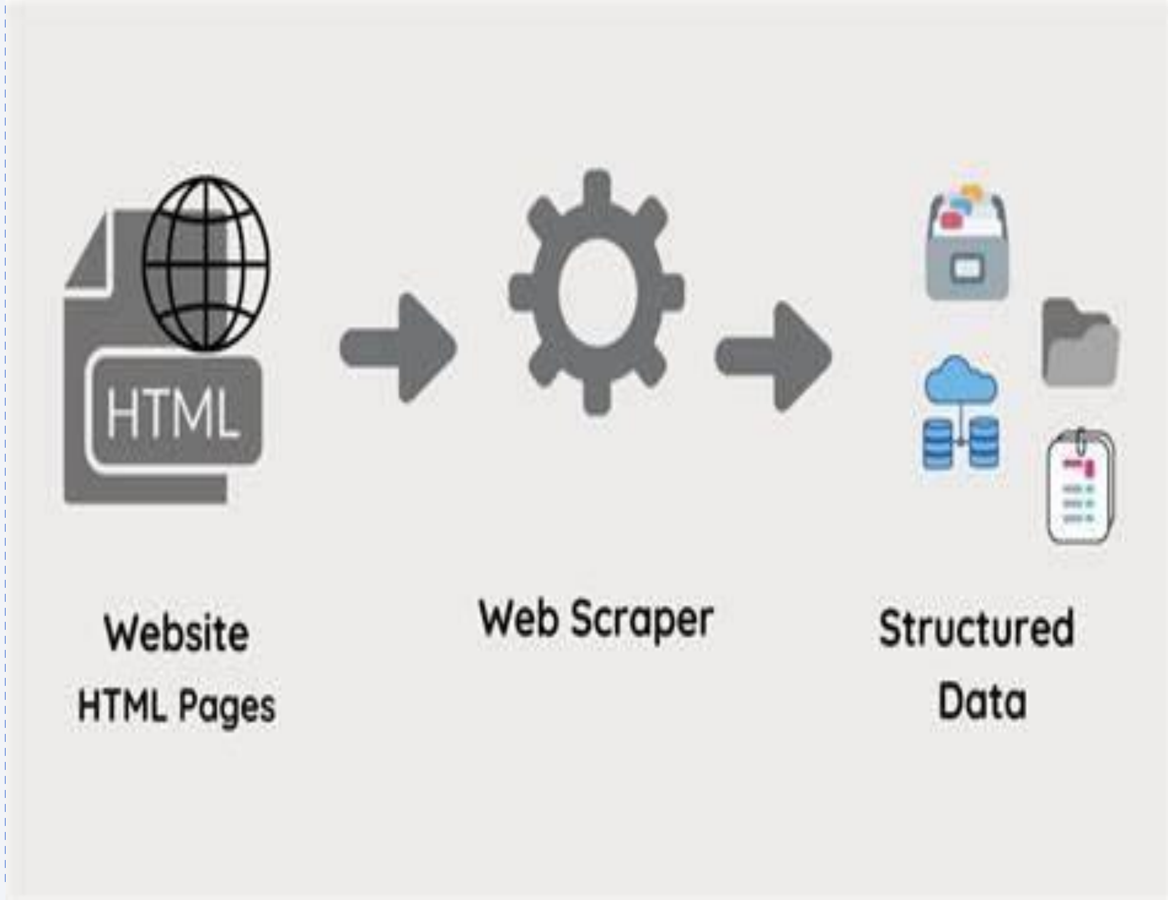
- url=“api.spacexdata.com/v4/launches/past”
- response = requests().get(url)
- data = pd.json_normalize(response.json())

Place your flowchart of SpaceX API calls here

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



Data Wrangling

- Load the SpaceX data with   the pd.read_csv() function.
- Do Exploratory Data analysis
- GitHub URL for our completed data wrangling related notebooks:

test/labs-jupyter-spacex-Data wrangling.ipynb at main · mouhamedlaminembaye/test (github.com)

EDA with Data Visualization

- Generally, we use a scatterplot to showcase the relationship between to variables
- Add the GitHub URL of our completed EDA with data visualization notebook:

test/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb at main · mouhamedlaminembaye/test (github.com)

EDA with SQL

- SQL queries performed:

➢select

➢Select distinct()

➢Where clause

➢Limit clause

➢groupby

- GitHub URL of our completed EDA with SQL notebook:

test/jupyter-labs-eda-sql-coursera_sqllite.ipynb at main · mouhamedlaminembaye/test (github.com)

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- These objects enables users to explore and manipulate data in an ineractive and real-time way
- GitHub URL of our completed interactive map with Folium map:

test/lab_jupyter_launch_site_location.jupyterlite.ipynb at main · mouhamedlaminembaye/test (github.com)

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

- For each model, we create an object and then a GridSearch model
- We find the best parameters and calculate the accuracy using best_score_ and the method score().
- To find the model that performs the bets, we compare the accuracies of different models.
- GitHub URL of our completed predictive analysis lab:
  test/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite (1).ipynb at main · mouhamedlaminembaye/test (github.com)

Results

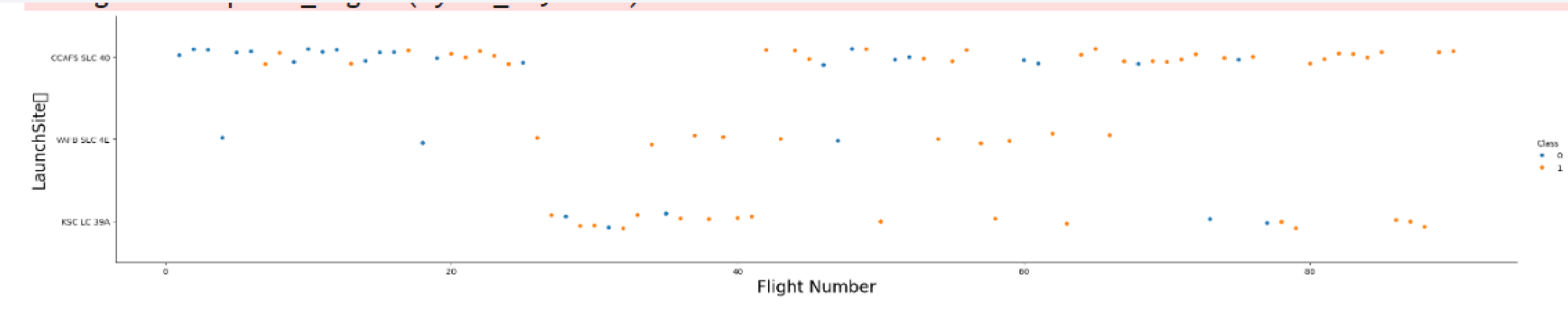
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

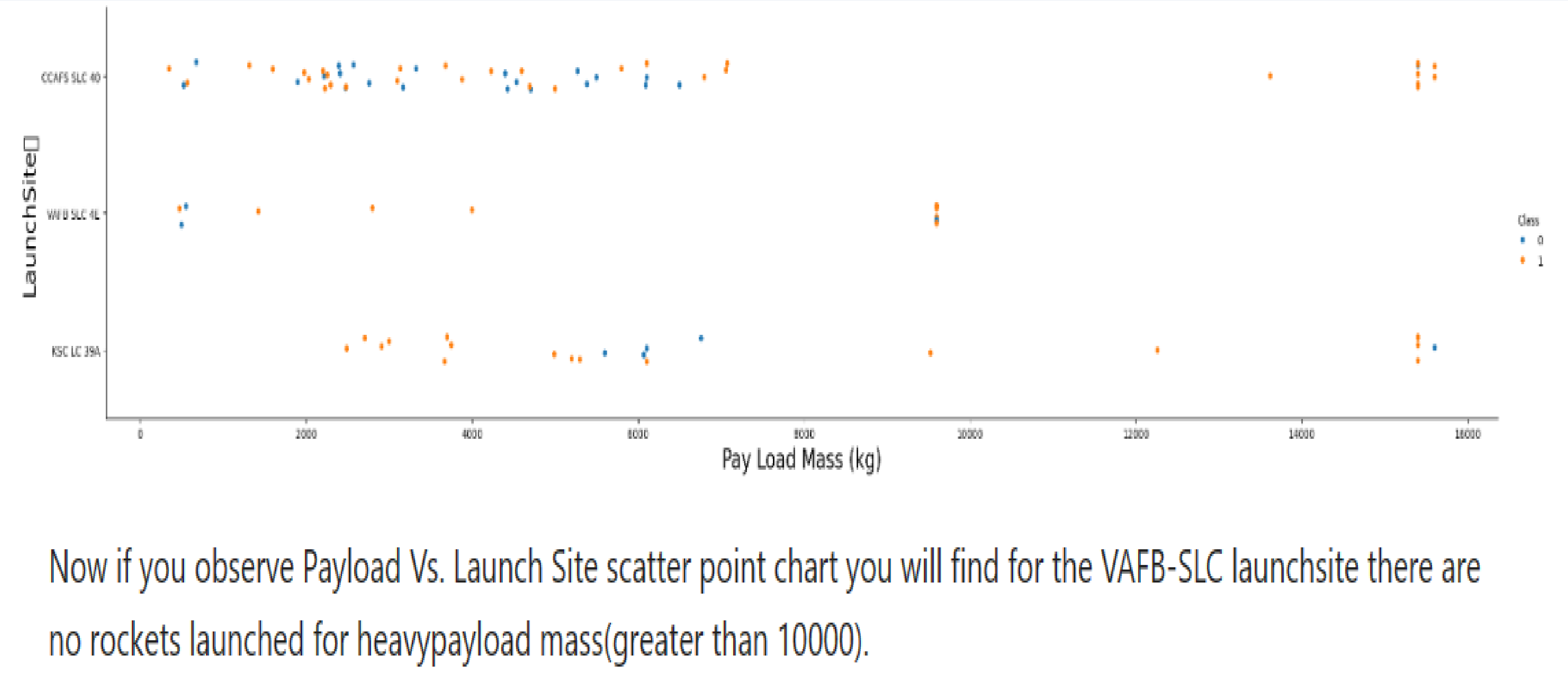
Flight Number vs. Launch Site

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



Payload vs. Launch Site

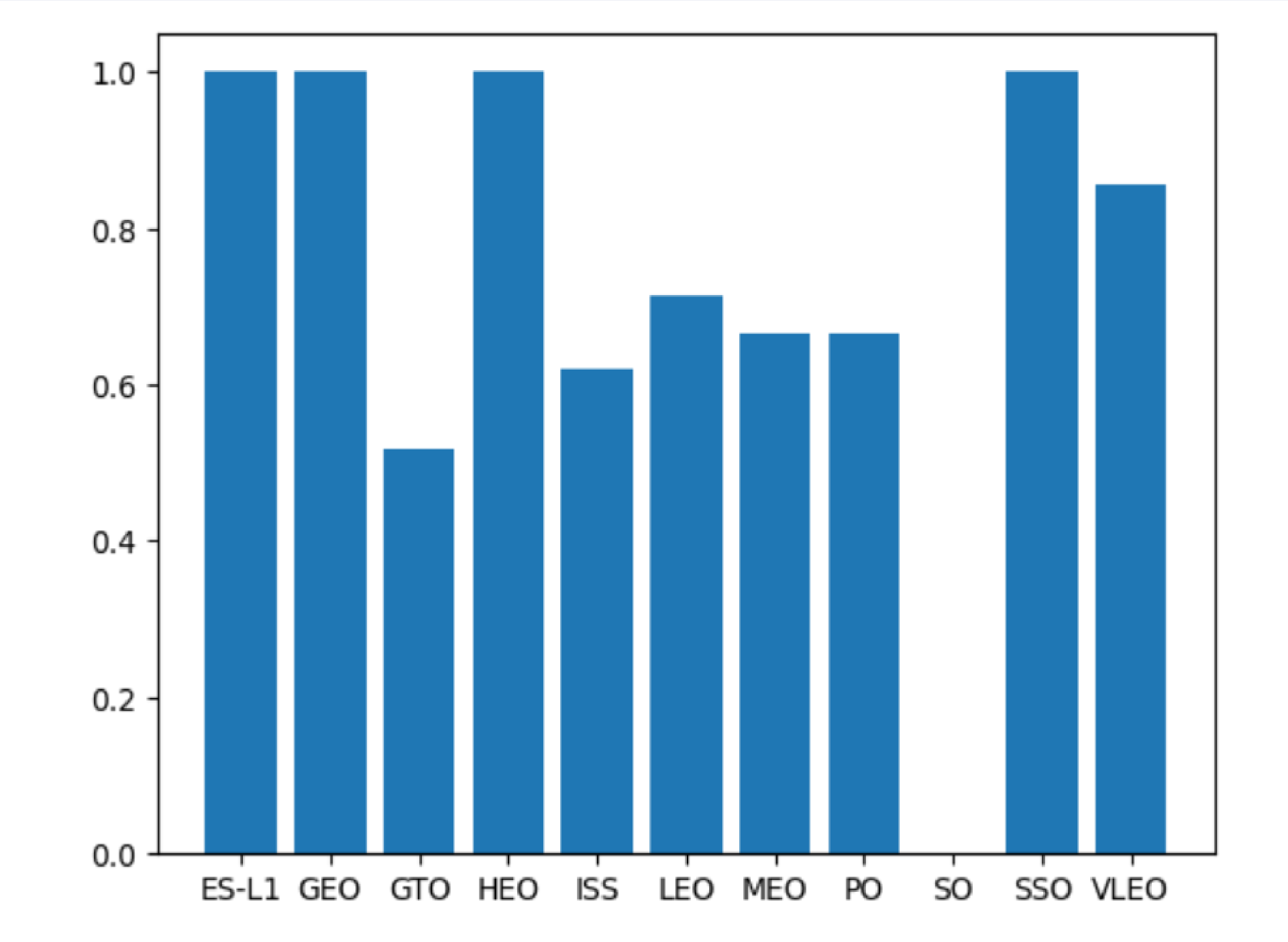
- Show a scatter plot of Payload vs. Launch Site



Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

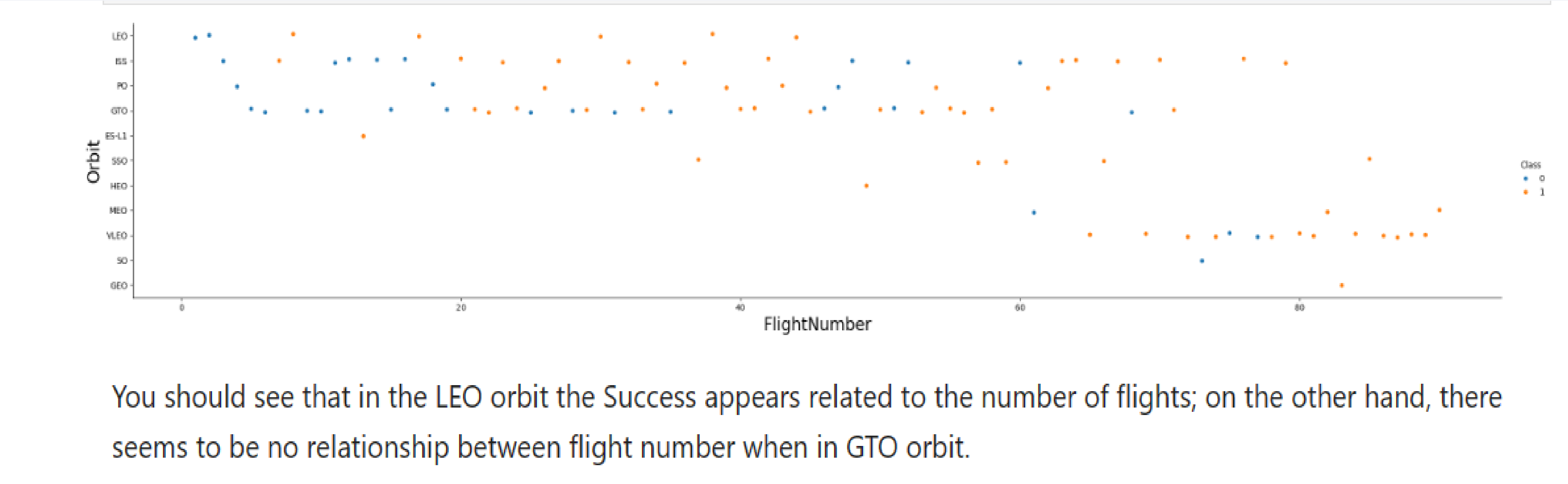
Success Rate vs. Orbit Type

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



Flight Number vs. Orbit Type

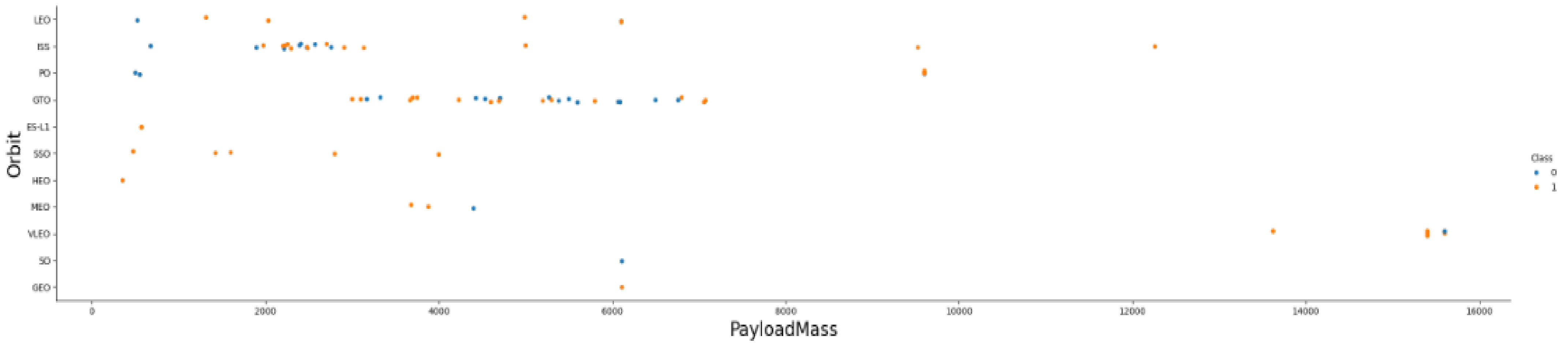
- Show a scatter point of Flight number vs. Orbit type



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.

Payload vs. Orbit Type

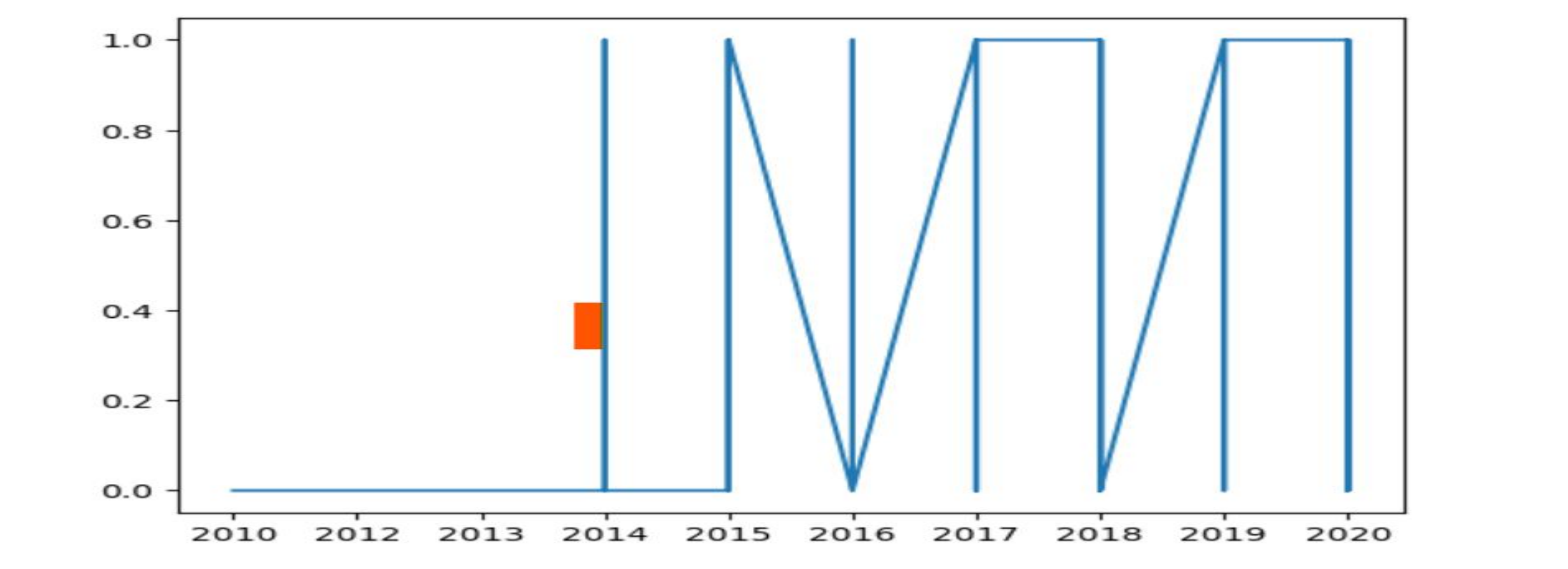
- Show a scatter point of payload vs. orbit type



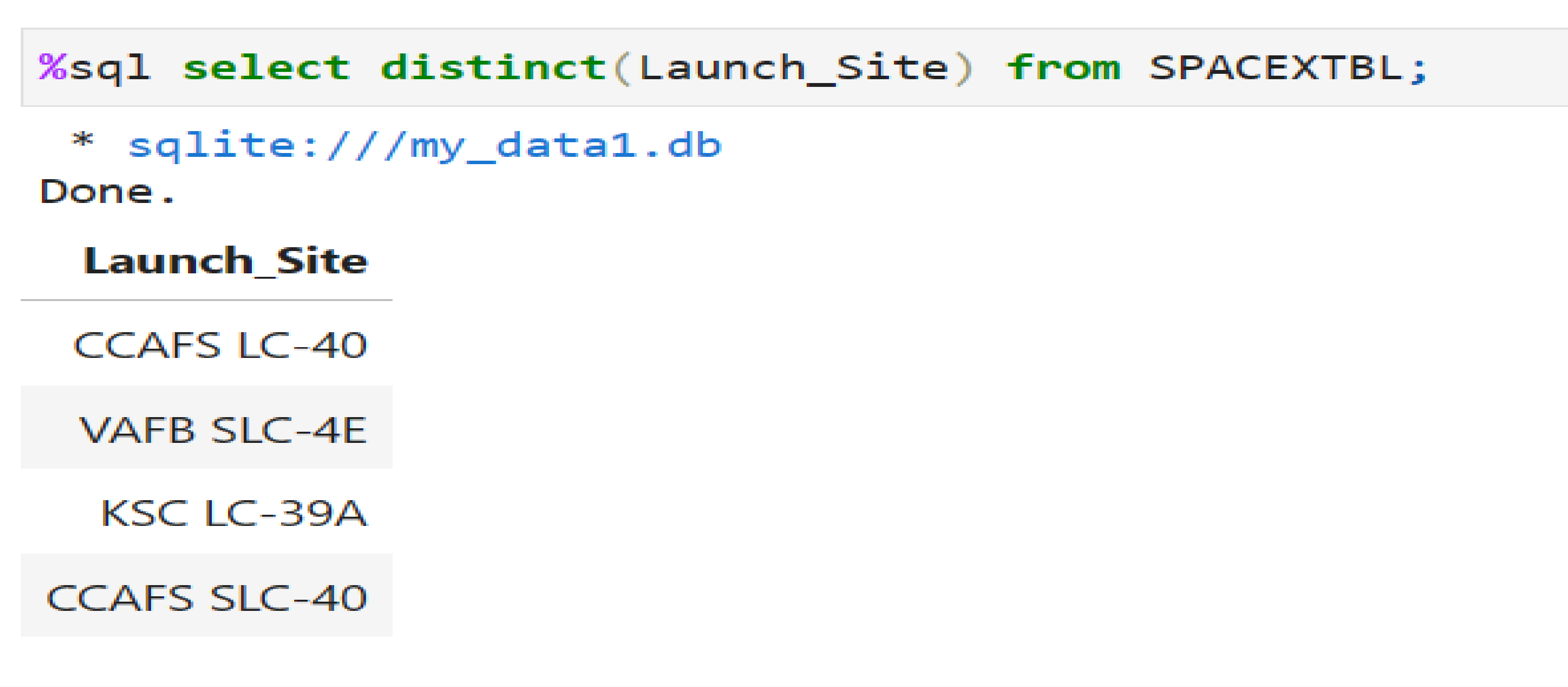
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

Launch Success Yearly Trend

- Show a line chart of yearly average success rate



All Launch Site Names



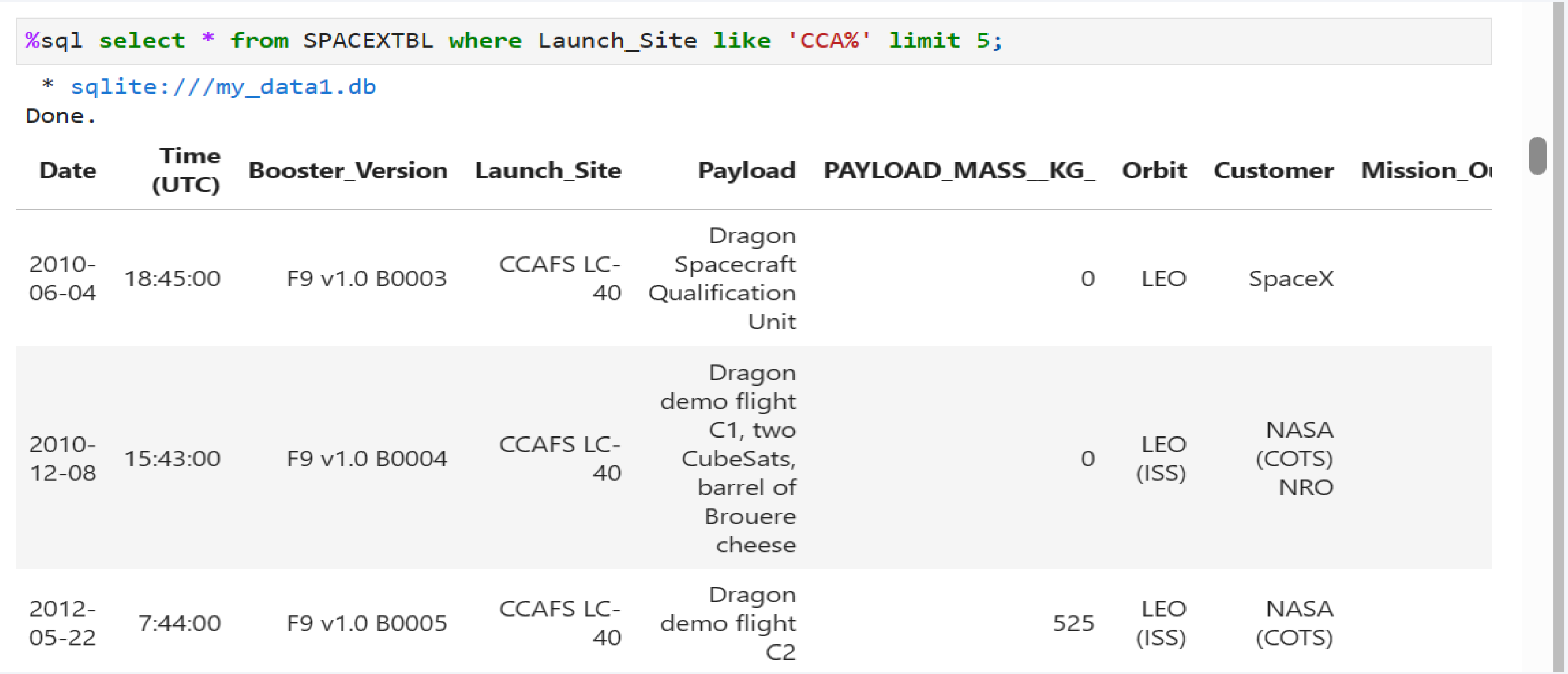
```
%sql select distinct(Launch_Site) from SPACEXTBL;
```

```
 * 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'

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



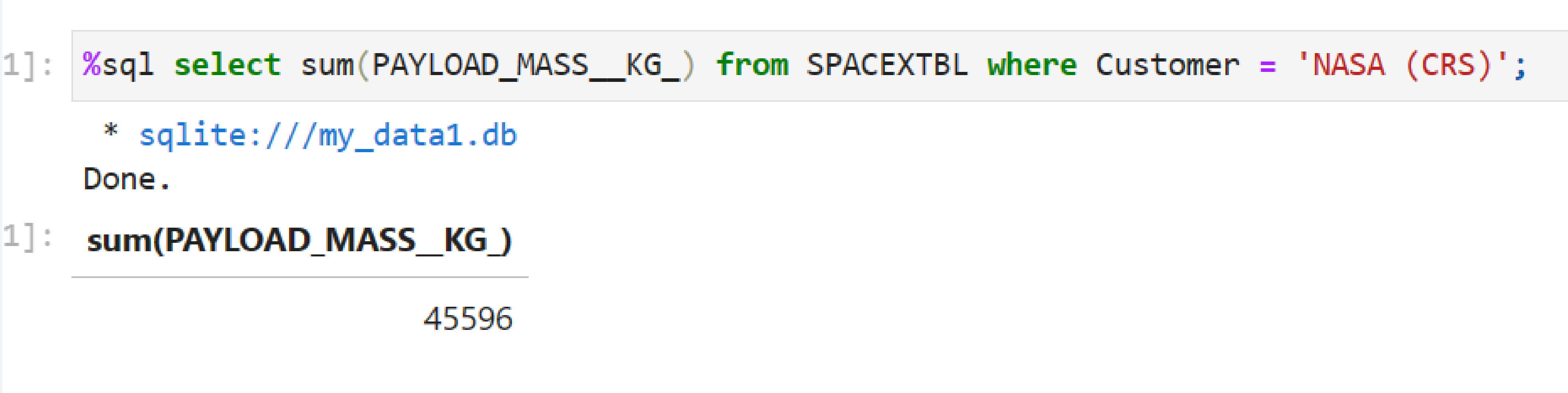
```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Ou
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	
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	
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	

Total Payload Mass

- Calculate the total payload carried by boosters from NASA



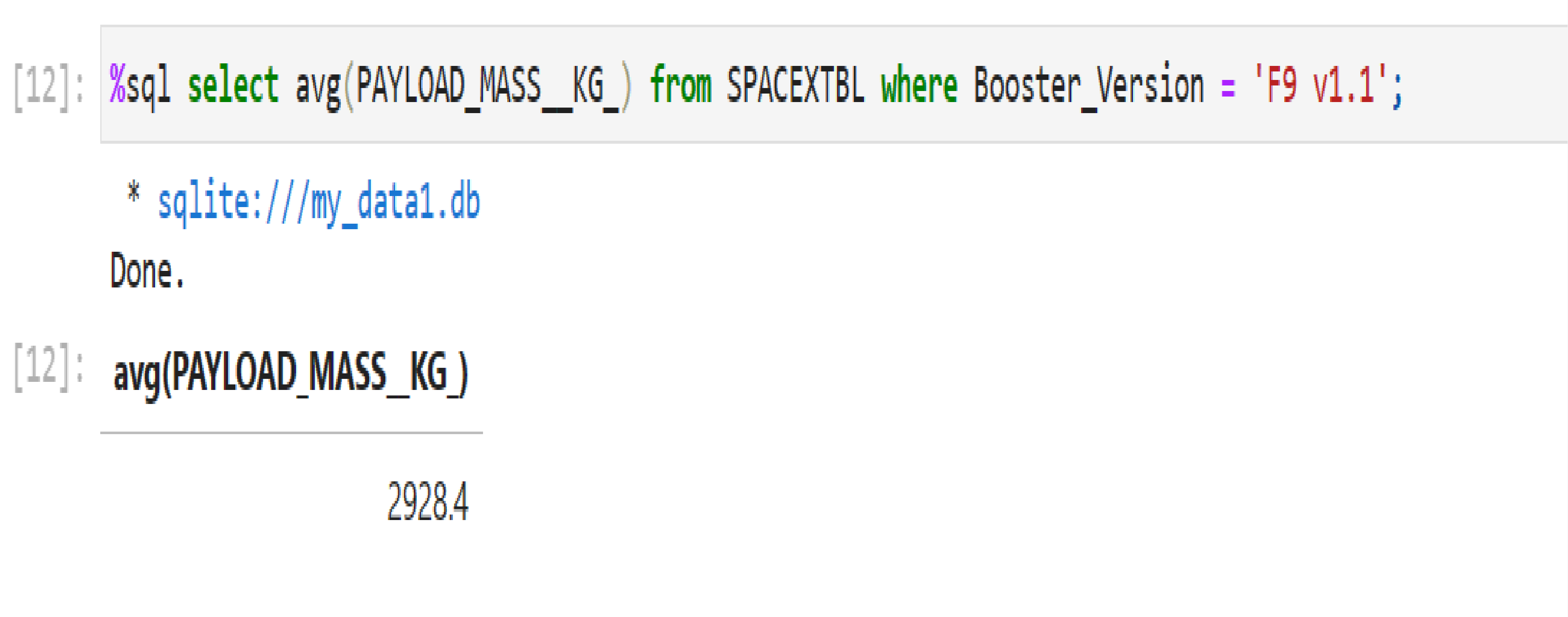
```
1]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer = 'NASA (CRS)';
```

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

```
1]: sum(PAYLOAD_MASS__KG_)
                     45596
```

Average Payload Mass by F9 v1.1

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



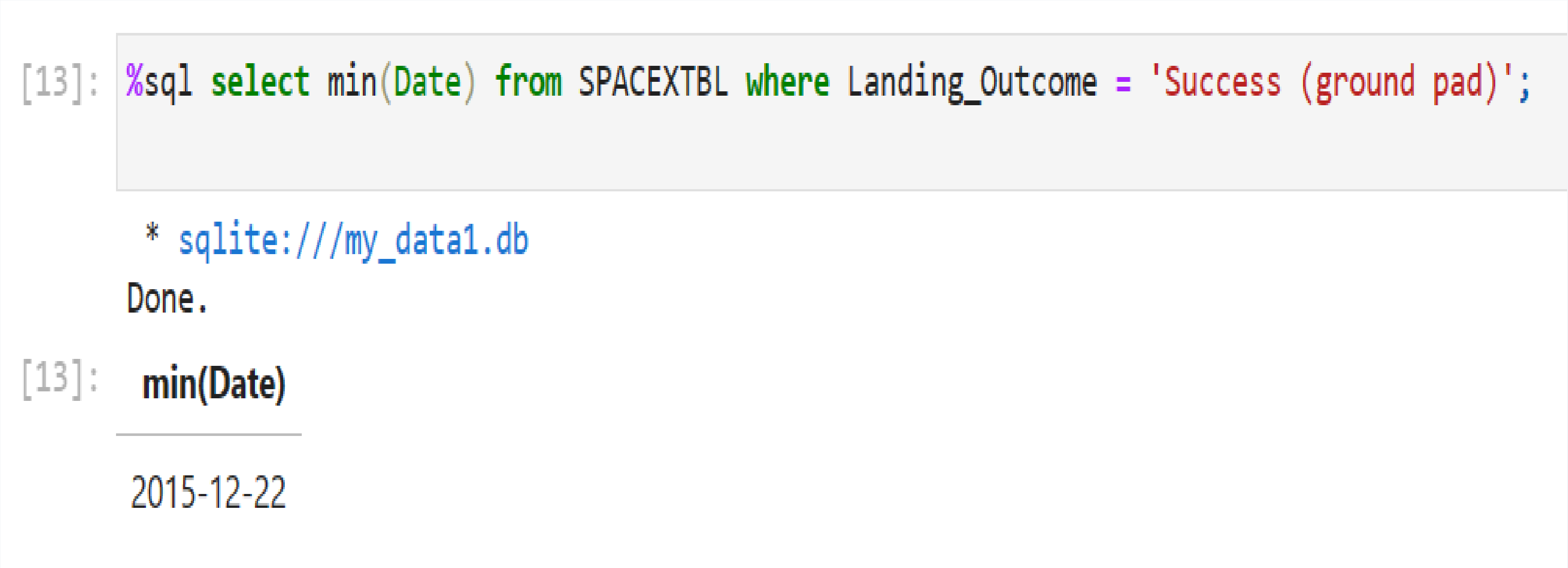
```
[12]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version = 'F9 v1.1';

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

[12]:

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

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



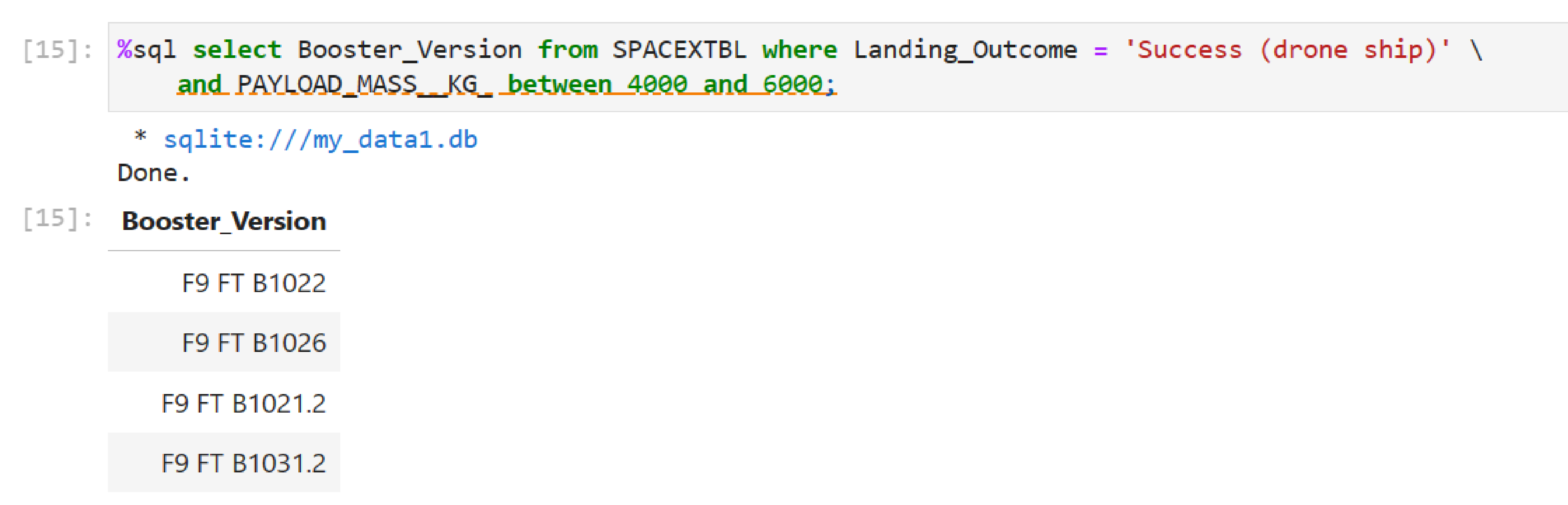
```
[13]: %sql select min(Date) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)';
```

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

```
[13]:  min(Date)
       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



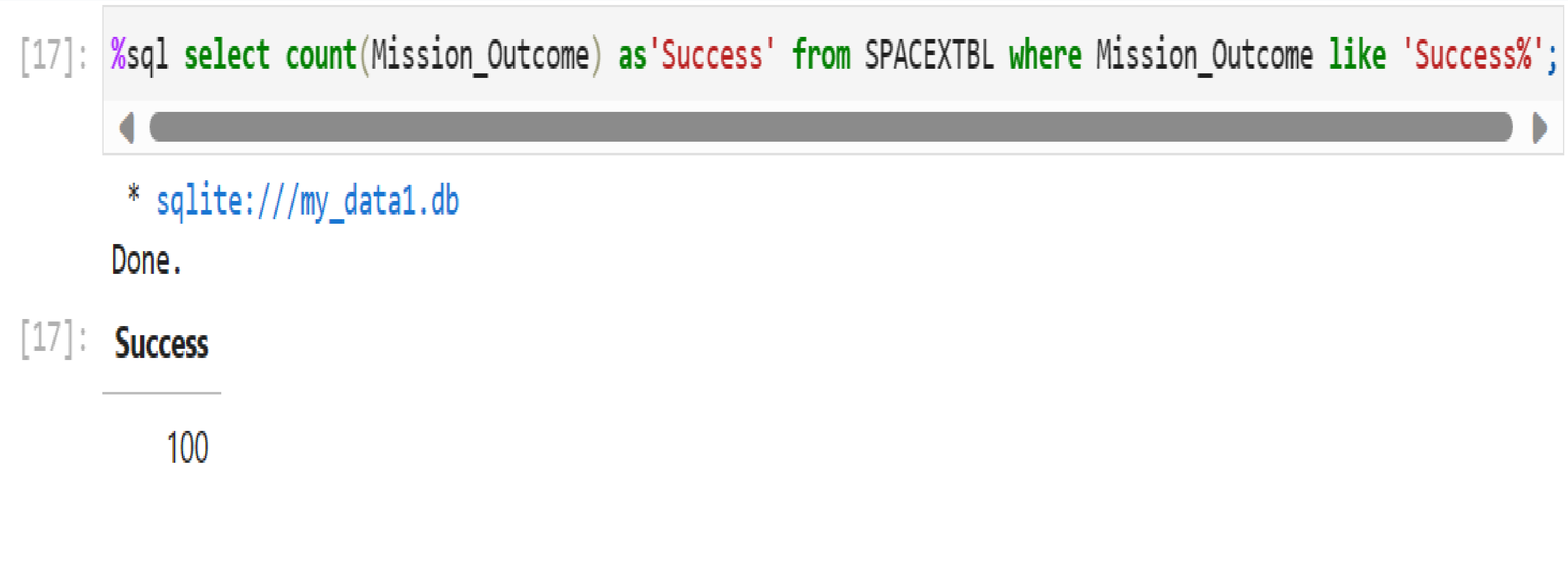
```
[15]: %sql select Booster_Version from SPACEXTBL where Landing_Outcome = 'Success (drone ship)' \
          and PAYLOAD_MASS__KG_ between 4000 and 6000;

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

[15]:

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes



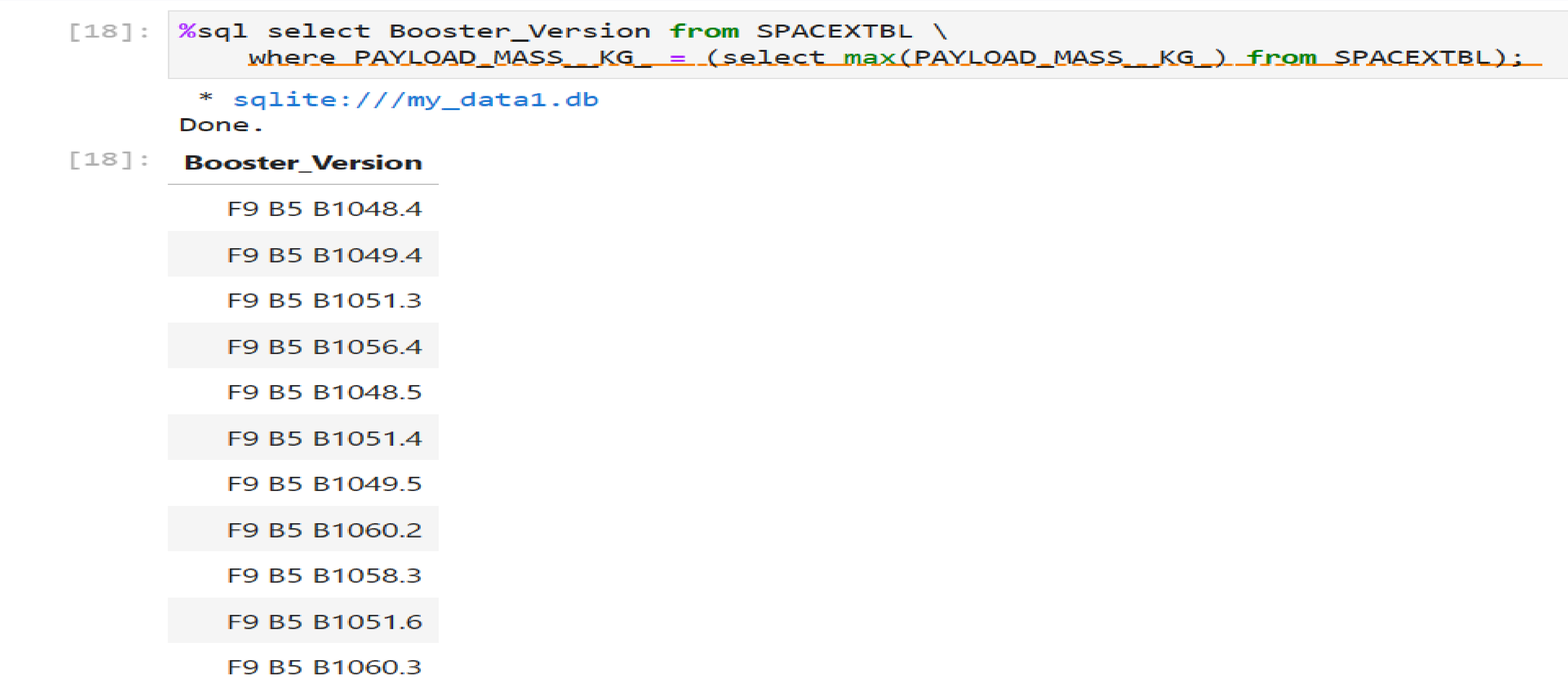
```
[17]: %sql select count(Mission_Outcome) as'Success' from SPACEXTBL where Mission_Outcome like 'Success%';
```

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

```
[17]: Success
      
          100
```

Boosters Carried Maximum Payload

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



```
[18]: %sql select Booster_Version from SPACEXTBL \
          where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
```

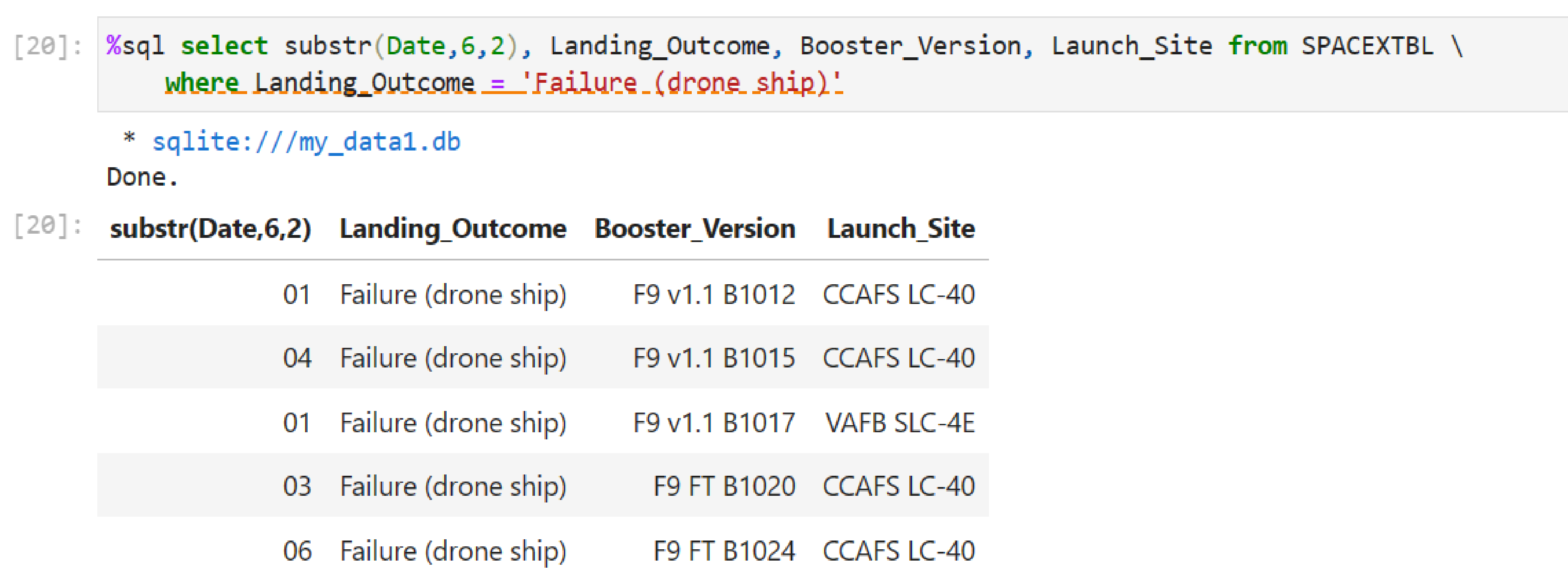
```
 * sqlite:///my_data1.db
Done.
```

[18]:

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

2015 Launch Records

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



```
[20]: %sql select substr(Date,6,2), Landing_Outcome, Booster_Version, Launch_Site from SPACEXTBL \
          where Landing_Outcome = 'Failure (drone ship)'

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

[20]:

substr(Date,6,2)	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
01	Failure (drone ship)	F9 v1.1 B1017	VAFB SLC-4E
03	Failure (drone ship)	F9 FT B1020	CCAFS LC-40
06	Failure (drone ship)	F9 FT B1024	CCAFS LC-40

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

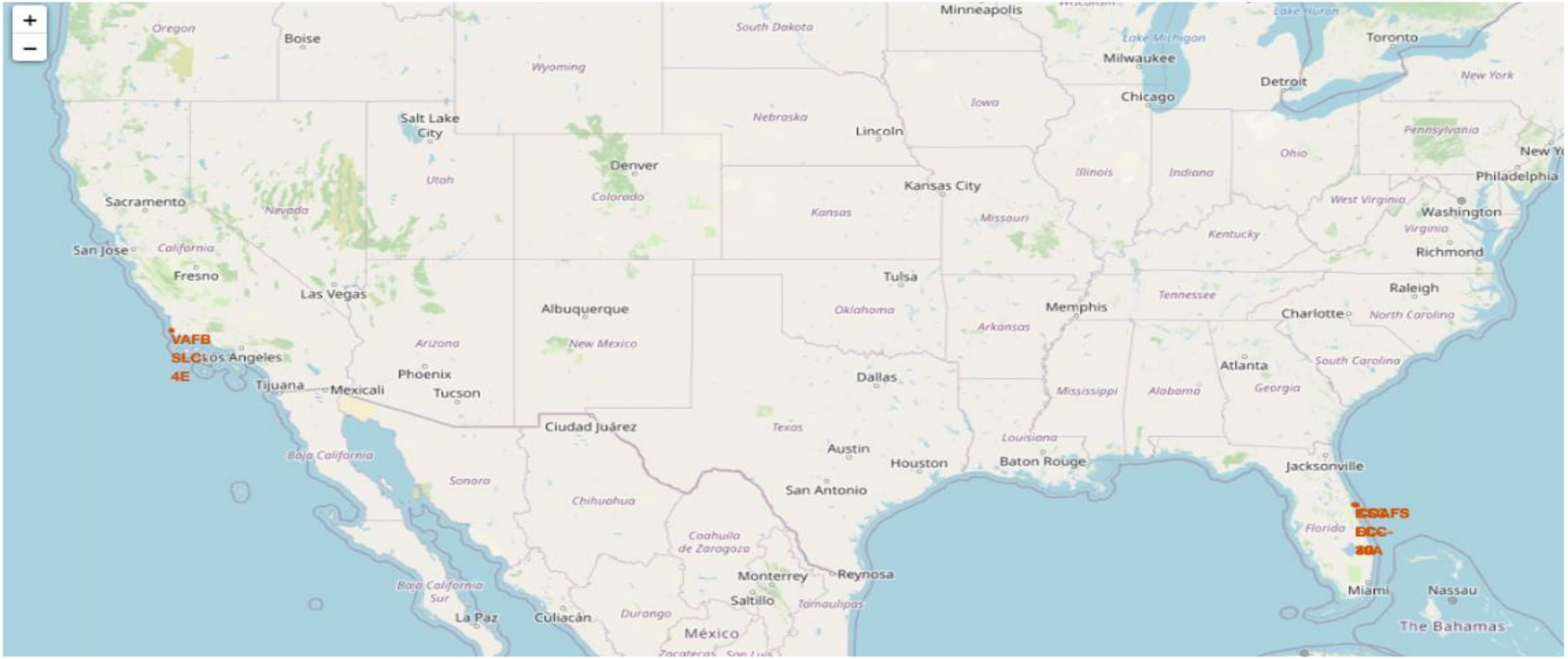
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

- Present your query result with a short explanation here

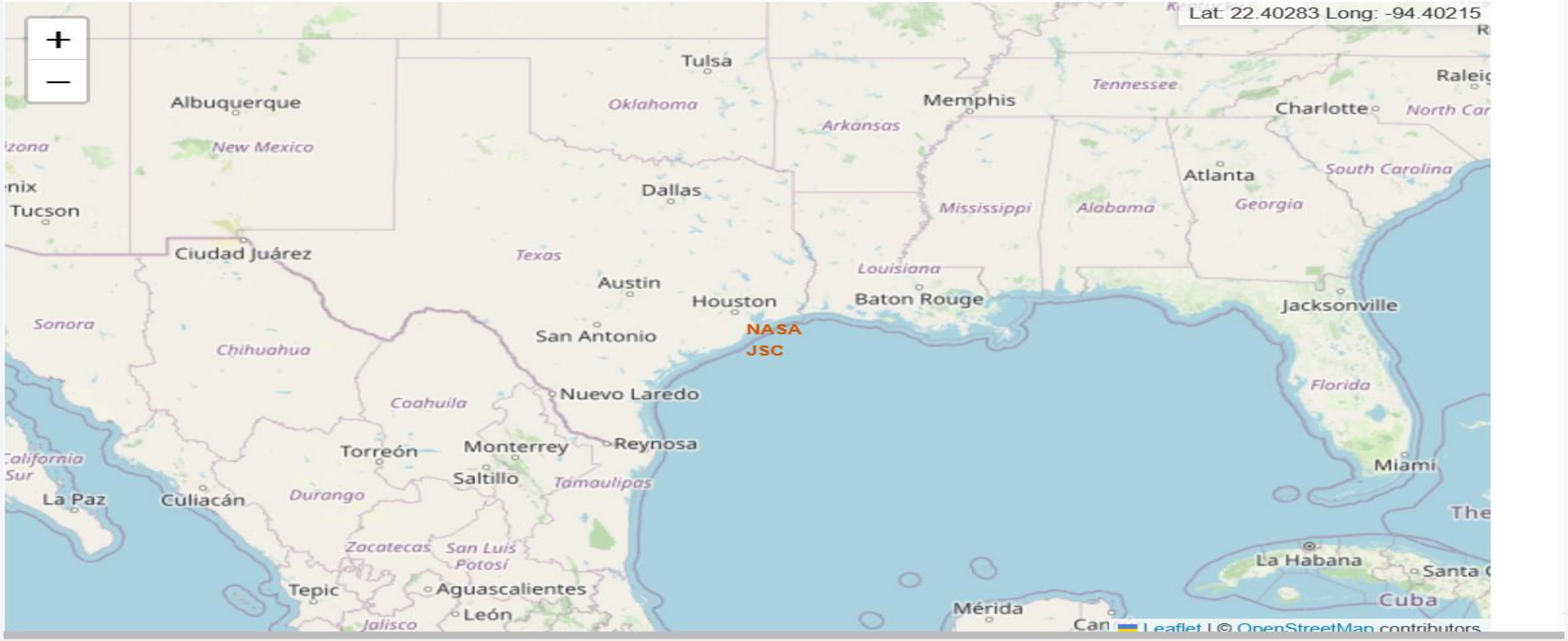
Section 3

Launch Sites Proximities Analysis

Launch Sites Locations Analysis with Folium



Site map


+
—
Lat: 22.40283 Long: -94.40215
NASA
JSC
Tulsa
Oklahoma
Albuquerque
New Mexico
Memphis
Arkansas
Tennessee
Charlotte
North Car
Raleig
South Carolina
Atlanta
Georgia
Alabama
Mississippi
Louisiana
Baton Rouge
Dallas
Texas
Austin
Houston
San Antonio
Nuevo Laredo
Reynosa
Ciudad Juárez
izona
enix
Tucson
Sonora
Chihuahua
Coahuila
Torreón
Monterrey
Saltillo
Tamaulipas
Durango
Culiacán
California
Sur
La Paz
Zacatecas
San Luis
Potosí
Tepic
Aguascalientes
León
Jalisco
Jacksonville
Florida
Miami
The
La Habana
Santa C
Cuba
Mérida
Can
Leaflet | © OpenStreetMap contributors

<Folium Map Screenshot 3>

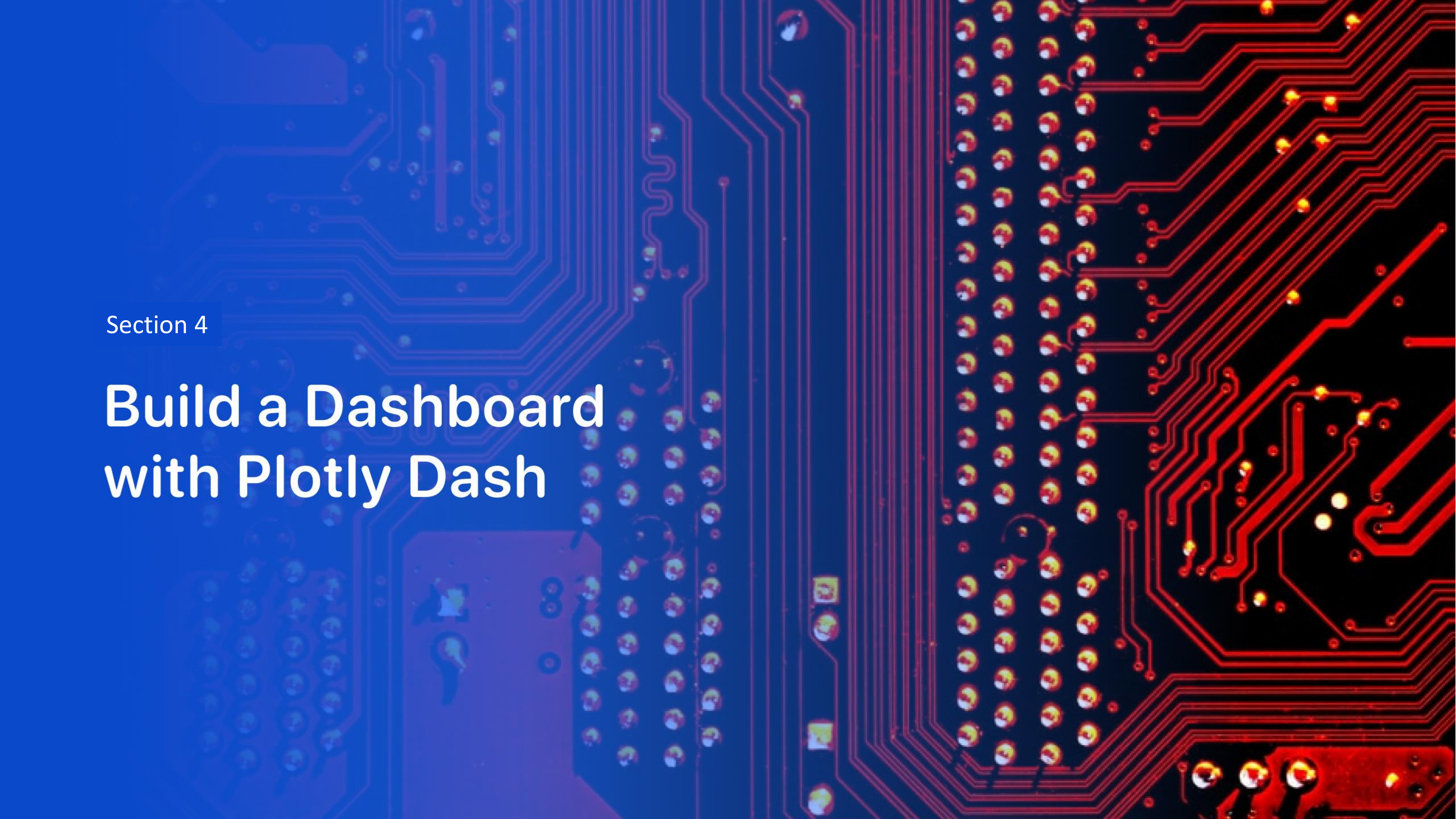
- Replace <Folium map screenshot 3> title with an appropriate title

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

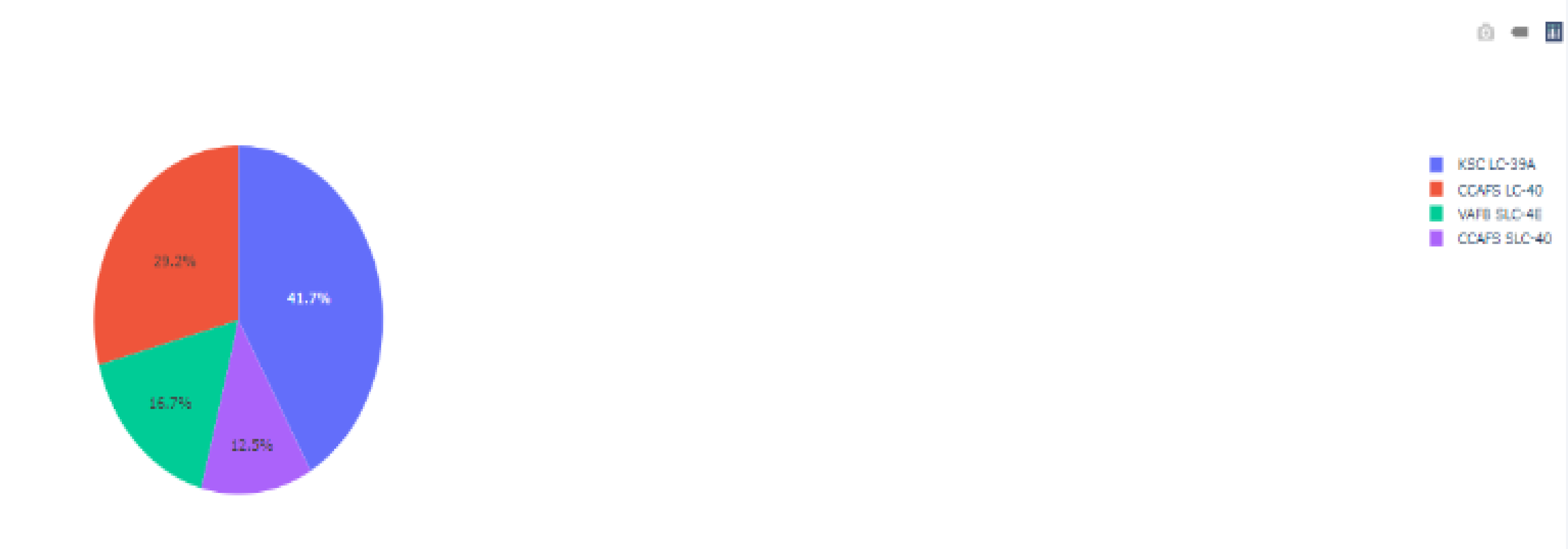
- Explain the important elements and findings on the screenshot

Section 4

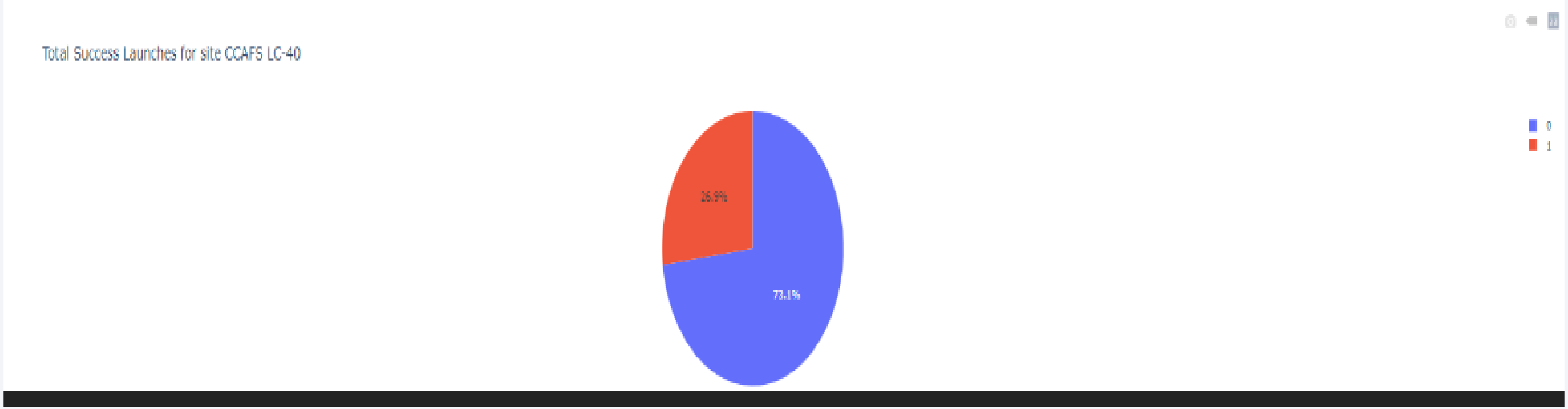
Build a Dashboard with Plotly Dash



spaceX Launch Records Dashboard



SpaceX highest launch success ration



<Dashboard Screenshot 3>

- Replace <Dashboard screenshot 3> title with an appropriate title

- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider

- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart

- Find which model has the highest classification accuracy

Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation

Conclusions

- Collecting data using API or web scraping
- Data Exploratory Analysis
- Using visulization
- Using sql queries
- Using Folium
- Using Dashbords
- Modeling

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!