IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Marta En
2025-05-10

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

In this project, we will learn to predict if the Falcon 9 first stage will land successfully. 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. Therefore if we can determine if the first stage will land, we can determine the cost of a launch.

Section 1

Methodology

Methodology

Executive Summary

Following steps were taken:

- Data collection from SpaceX API.
- Data wrangling to extract and prepare necessary data.
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using four classification models.

As a result, a model is trained that showed 94.4% accuracy on the test set.

Data Collection

- Retrieving data from SpaceX API
- Scraping
- Data wrangling to clean up the dataset

Data Collection – SpaceX API

Data collection included the following steps:

- First, data was collected from SpaceX API in JSON format
- Next, JSON data was turned it into a Pandas dataframe
- Next, we worked on data to leave columns and records we will be focusing on

See the GitHub URL of the completed data collection notebook:

https://github.com/MartaEN/jupiter_sandbox/blob/main/01_jupyter-labs-spacex-data-collection-api.ipynb

Data Wrangling

In this part, we performed some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.

We analyzed the 'Outcome' column which contained such values as 'True ASDS', 'False ASDS', 'True Ocean', 'False Ocean' etc.

We then created a new column 'Class' and filled it with 1 for successful outcomes and 0 for failures.

See the GitHub URL of the completed data wrangling notebook:

https://github.com/MartaEN/jupiter_sandbox/blob/main/02_labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

In visualization part, we plotted success / failure outcomes in relation with launch sites, rocket payload and orbit type. We also looked into the historic trend of success / failure outcomes over time.

See the GitHub URL of EDA with data visualization notebook:

https://github.com/MartaEN/jupiter_sandbox/blob/main/04_edadataviz.ipynb

EDA with SQL

- Following points were investigated with SQL:
  - Launch sites (2 queries)
  - Payload carried by boosters (3 queries)
  - Details of successful and failure outcomes (5 queries)

See the GitHub URL of EDA with SQL notebook:
https://github.com/MartaEN/jupiter_sandbox/blob/main/03_jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

In this part, we built an interactive map showing launch locations with numbers of launches.

See the GitHub URL of the interactive Folium map notebook: https://github.com/MartaEN/jupiter_sandbox/blob/main/05_lab_jupyter_launch_site_location.ipynb

Predictive Analysis (Classification)

Here, we build four models with the following accuracy rates on the test set:

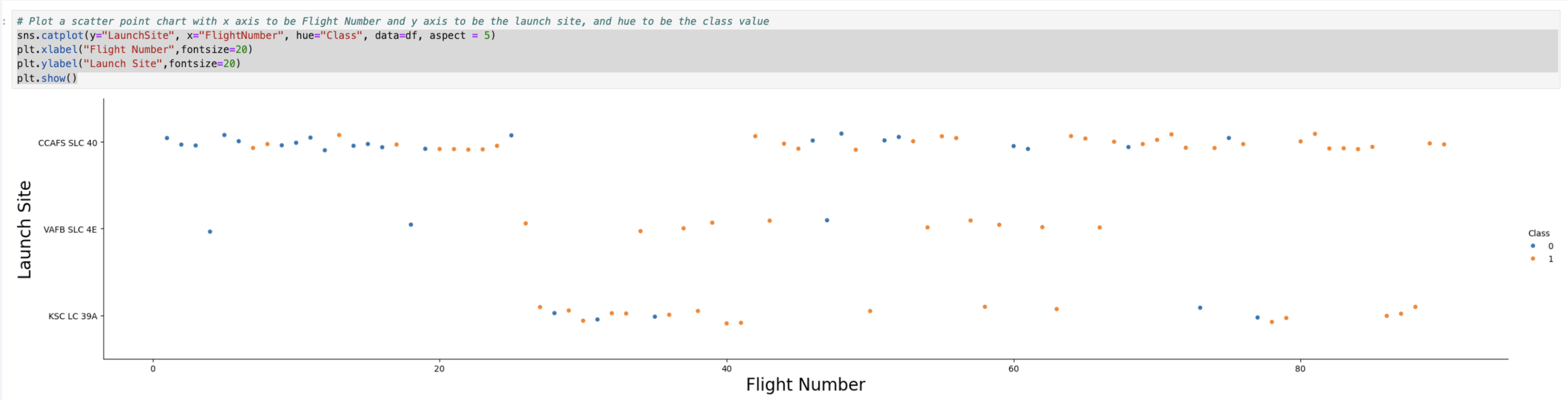
- Decision tree classifier: 94.4% (the winner)
- Logistic regression: 83.3%
- Support vector machine: 83.3%
- K nearest neighbors: 83.3%

See the GitHub URL of the completed predictive analysis notebook : https://github.com/MartaEN/jupiter_sandbox/blob/main/06_SpaceX_Machine%20Learning%20Prediction.ipynb

Section 2

Insights drawn from EDA

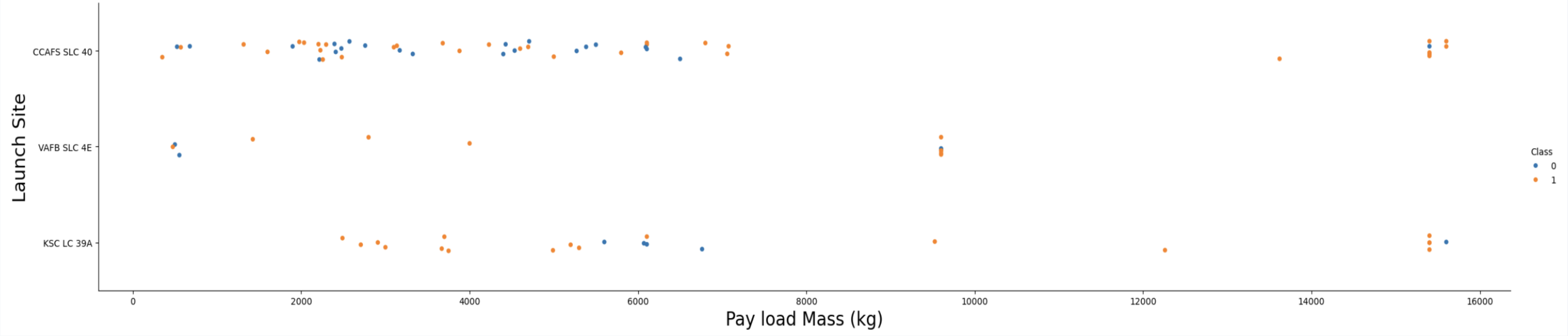
Flight Number vs. Launch Site



```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```

We can see that most launches were made from CCAFS SLC 40 site, followed by KSC LC 39A site. We can also note that success rate is increasing over time for all the three lauch sites.

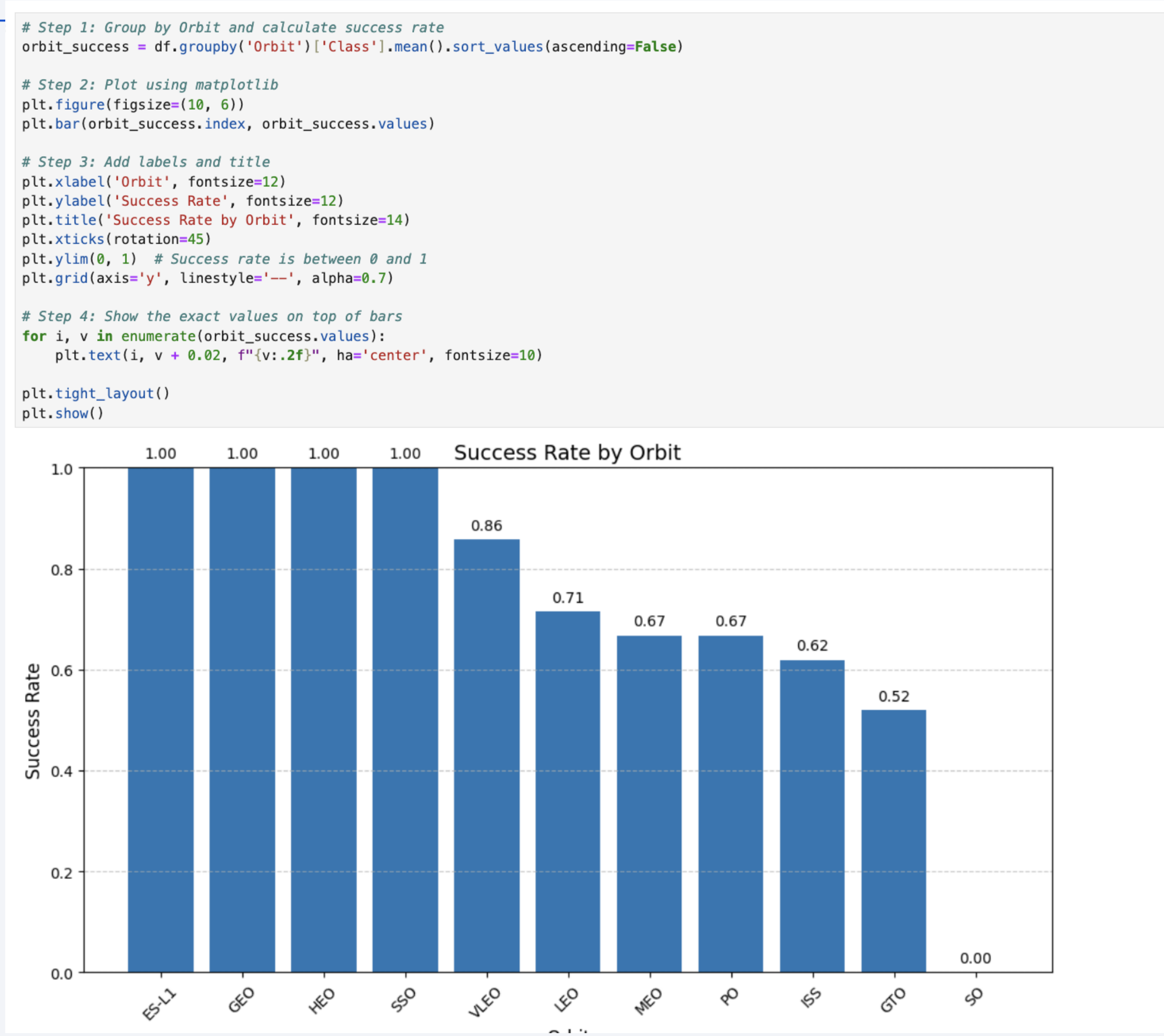
Payload vs. Launch Site



We can note that:

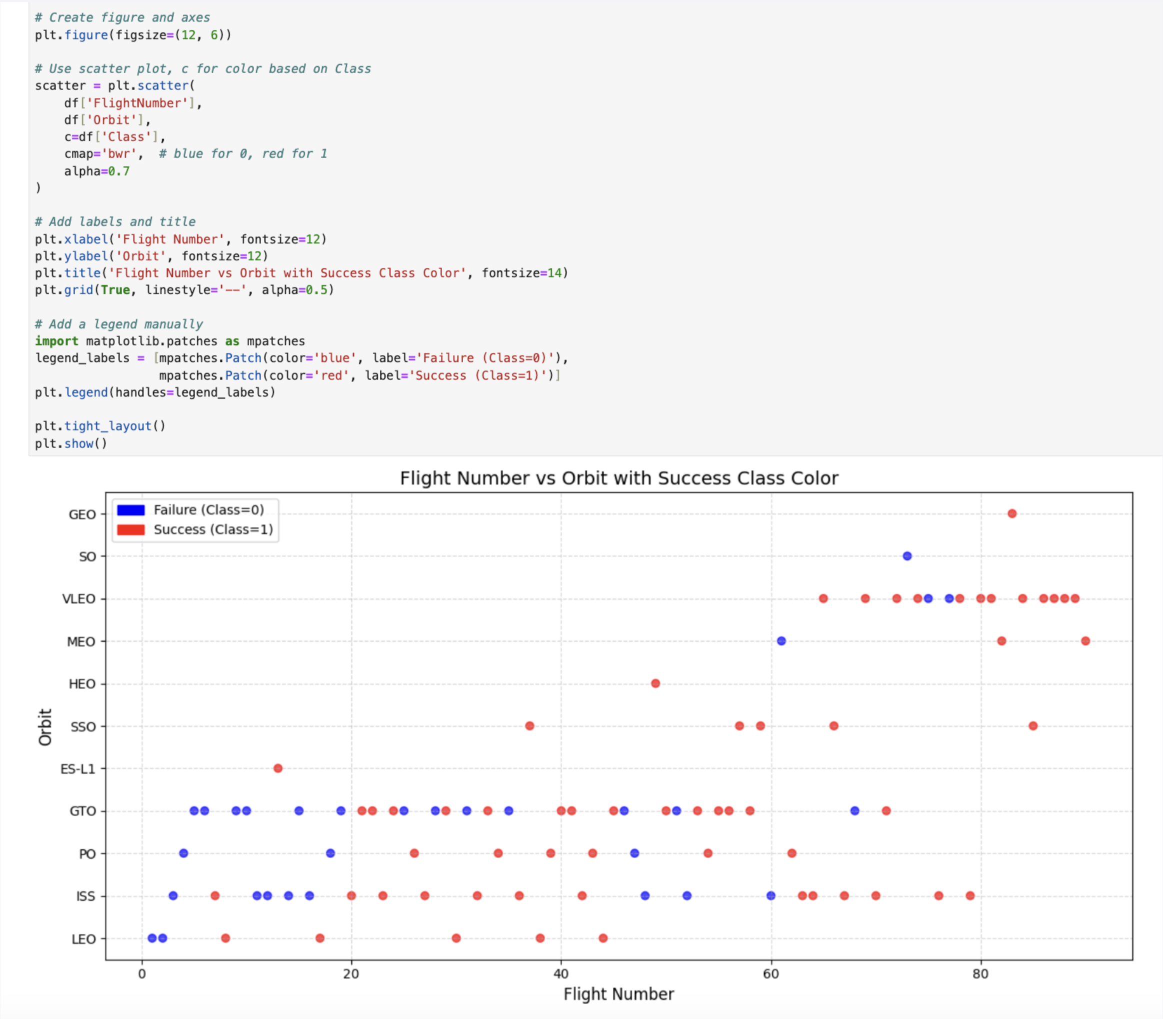
- Most launches were made with payload below 8000 kg.
- VAFB SLC 4E didn't launch rockets over 10000 kg.

Success Rate vs. Orbit Type



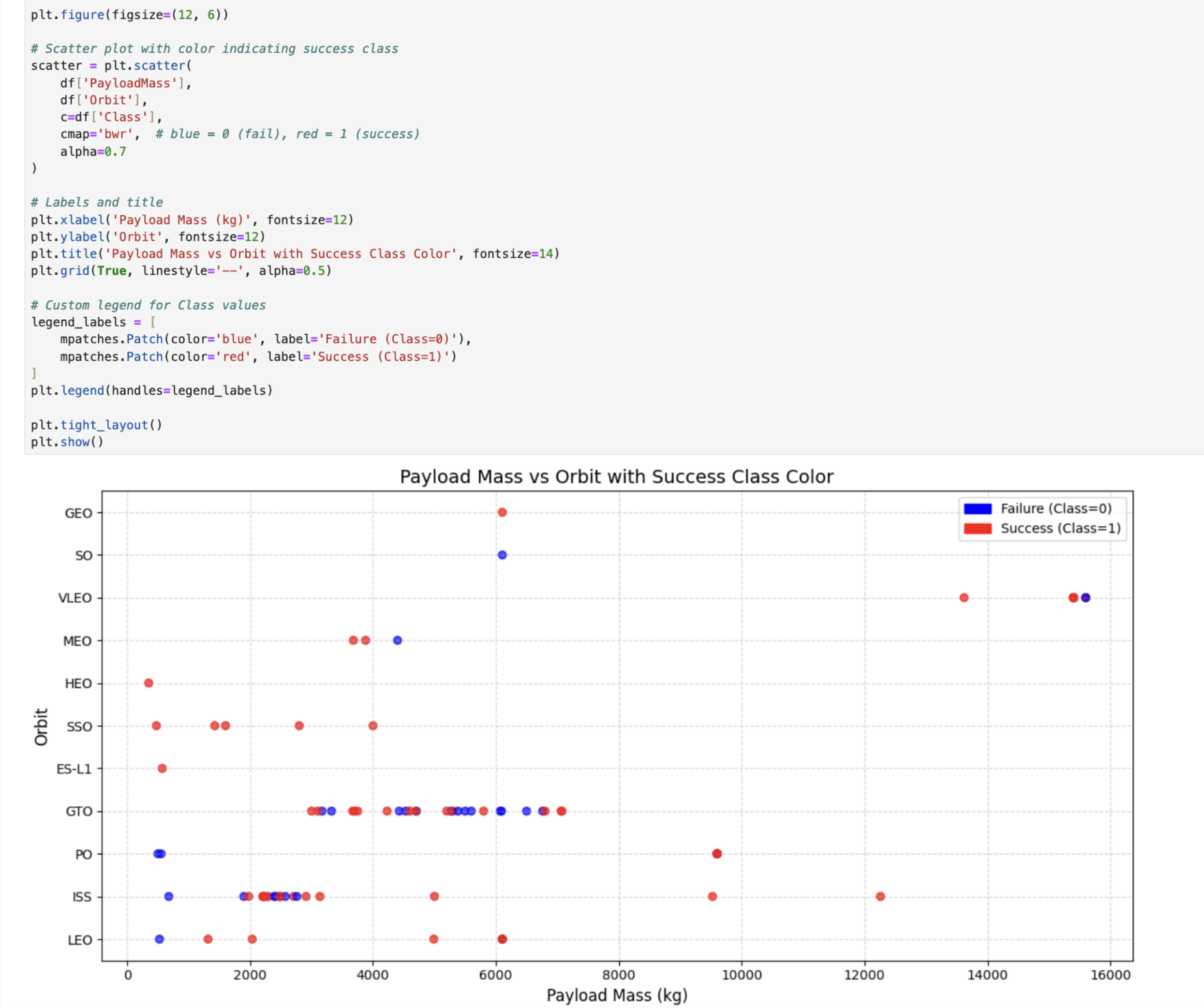
```
# Step 1: Group by Orbit and calculate success rate
orbit_success = df.groupby('Orbit')['Class'].mean().sort_values(ascending=False)

# Step 2: Plot using matplotlib
plt.figure(figsize=(10, 6))
plt.bar(orbit_success.index, orbit_success.values)

# Step 3: Add labels and title
plt.xlabel('Orbit', fontsize=12)
plt.ylabel('Success Rate', fontsize=12)
plt.title('Success Rate by Orbit', fontsize=14)
plt.xticks(rotation=45)
plt.ylim(0, 1)  # Success rate is between 0 and 1
plt.grid(axis='y', linestyle='--', alpha=0.7)

# Step 4: Show the exact values on top of bars
for i, v in enumerate(orbit_success.values):
    plt.text(i, v + 0.02, f"{v:.2f}", ha='center', fontsize=10)

plt.tight_layout()
plt.show()
```

Flight Number vs. Orbit Type



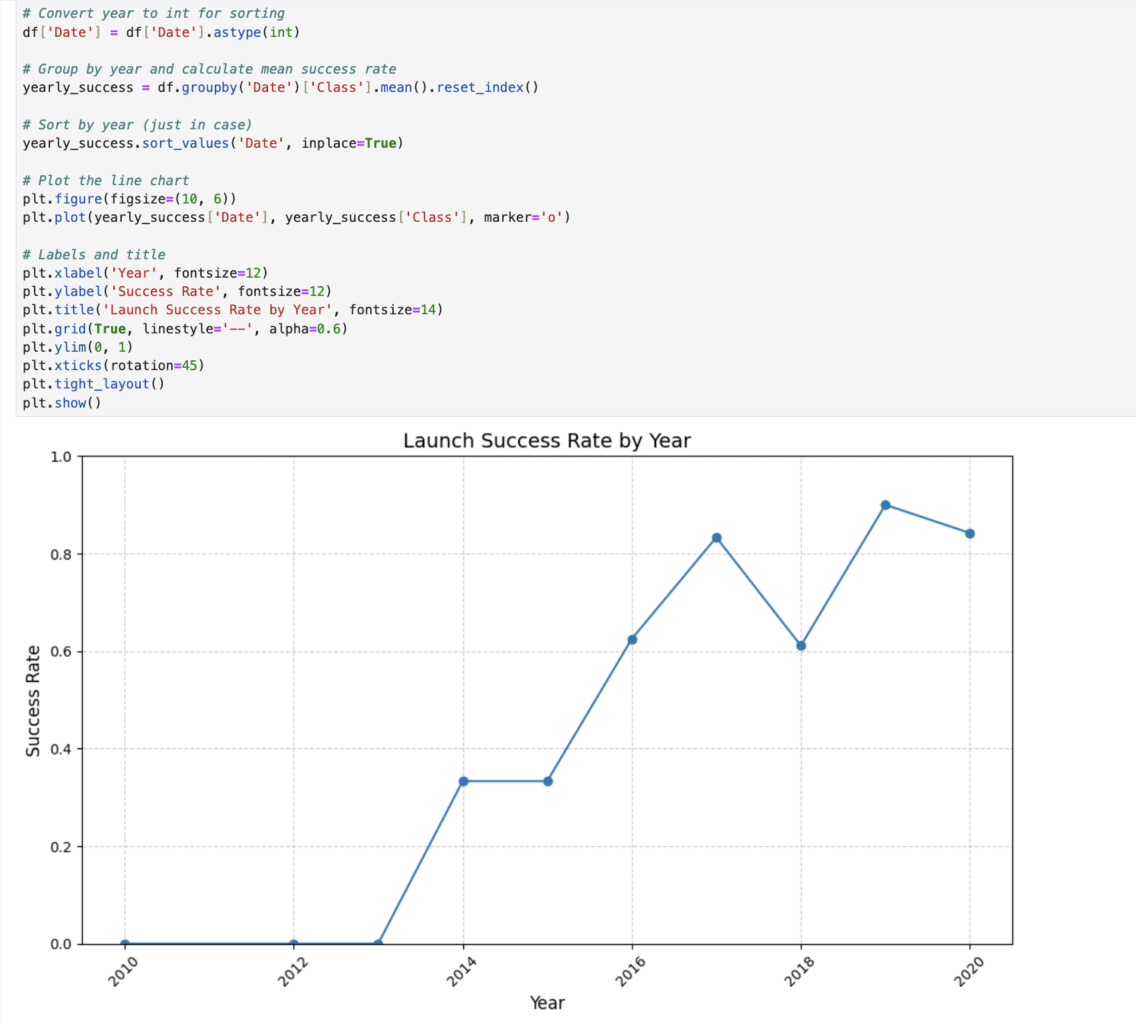
```python
# Create figure and axes
plt.figure(figsize=(12, 6))

# Use scatter plot, c for color based on Class
scatter = plt.scatter(
    df['FlightNumber'],
    df['Orbit'],
    c=df['Class'],
    cmap='bwr',  # blue for 0, red for 1
    alpha=0.7
)

# Add labels and title
plt.xlabel('Flight Number', fontsize=12)
plt.ylabel('Orbit', fontsize=12)
plt.title('Flight Number vs Orbit with Success Class Color', fontsize=14)
plt.grid(True, linestyle='--', alpha=0.5)

# Add a legend manually
import matplotlib.patches as mpatches
legend_labels = [mpatches.Patch(color='blue', label='Failure (Class=0)'),
                mpatches.Patch(color='red', label='Success (Class=1)')]
plt.legend(handles=legend_labels)

plt.tight_layout()
plt.show()
```

Payload vs. Orbit Type



```
plt.figure(figsize=(12, 6))

# Scatter plot with color indicating success class
scatter = plt.scatter(
    df['PayloadMass'],
    df['Orbit'],
    c=df['Class'],
    cmap='bwr',  # blue = 0 (fail), red = 1 (success)
    alpha=0.7
)

# Labels and title
plt.xlabel('Payload Mass (kg)', fontsize=12)
plt.ylabel('Orbit', fontsize=12)
plt.title('Payload Mass vs Orbit with Success Class Color', fontsize=14)
plt.grid(True, linestyle='--', alpha=0.5)

# Custom legend for Class values
legend_labels = [
    mpatches.Patch(color='blue', label='Failure (Class=0)'),
    mpatches.Patch(color='red', label='Success (Class=1)')
]
plt.legend(handles=legend_labels)

plt.tight_layout()
plt.show()
```

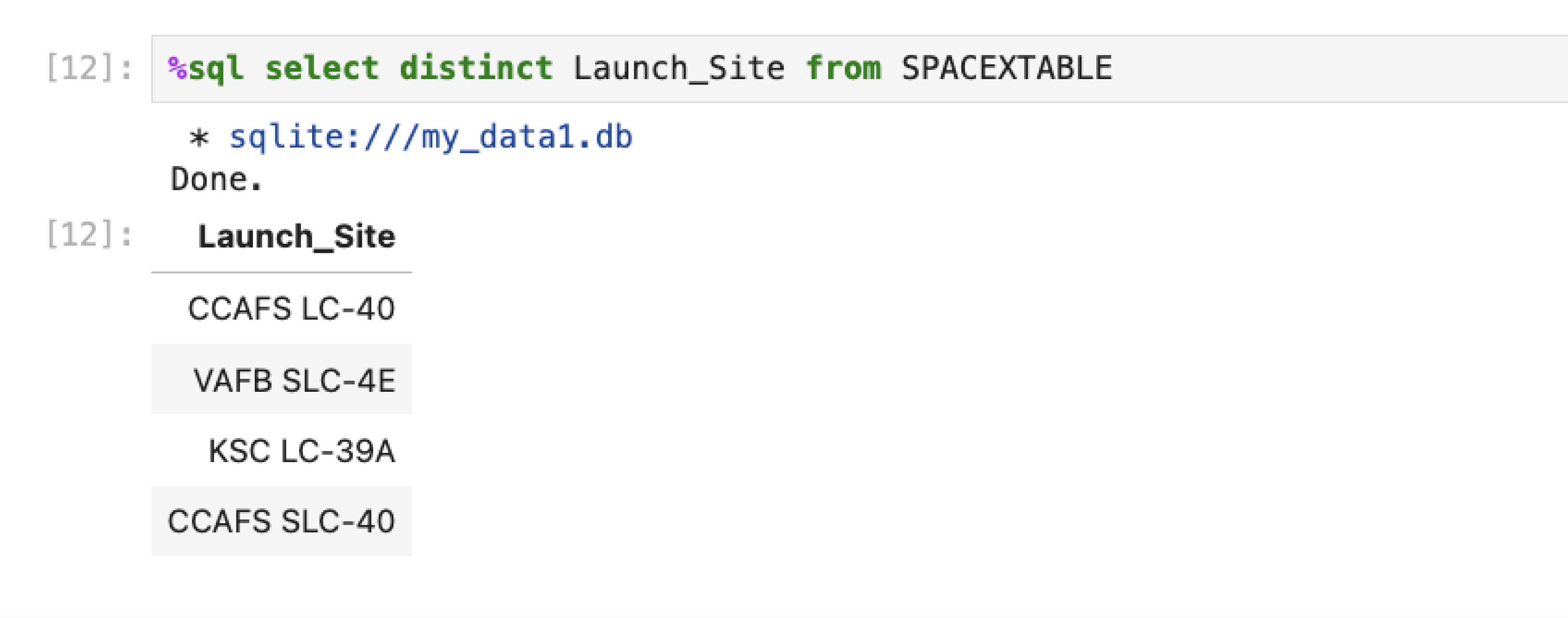
Launch Success Yearly Trend



```
# Convert year to int for sorting
df['Date'] = df['Date'].astype(int)

# Group by year and calculate mean success rate
yearly_success = df.groupby('Date')['Class'].mean().reset_index()

# Sort by year (just in case)
yearly_success.sort_values('Date', inplace=True)

# Plot the line chart
plt.figure(figsize=(10, 6))
plt.plot(yearly_success['Date'], yearly_success['Class'], marker='o')

# Labels and title
plt.xlabel('Year', fontsize=12)
plt.ylabel('Success Rate', fontsize=12)
plt.title('Launch Success Rate by Year', fontsize=14)
plt.grid(True, linestyle='--', alpha=0.6)
plt.ylim(0, 1)
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

All Launch Site Names

- There are four launch sites registered in the dataset :



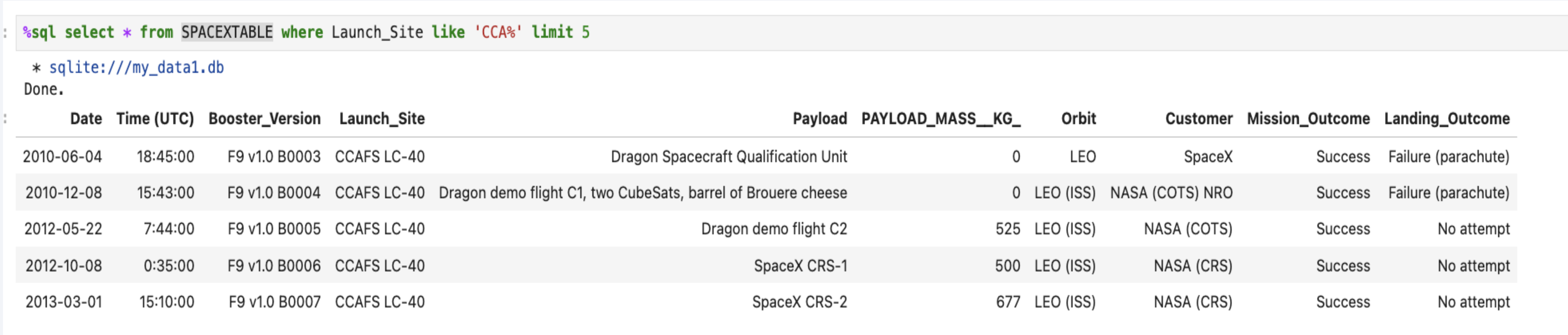
```
[12]: %sql select distinct Launch_Site from SPACEXTABLE

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

[12]:

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

Launch Site Names Begin with 'CCA'

- Here's 5 launch records where launch sites begin with `CCA`



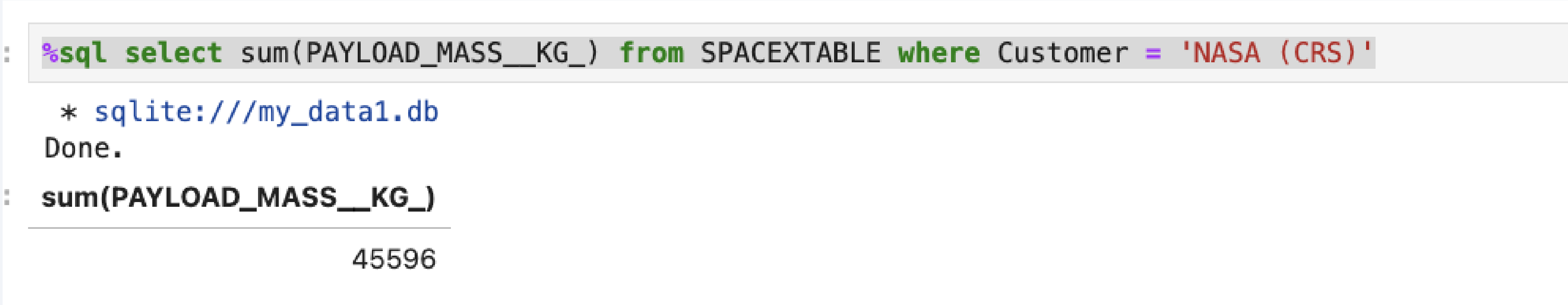
```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_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

- Total payload carried by boosters from NASA (CRS) is 45596 kg:



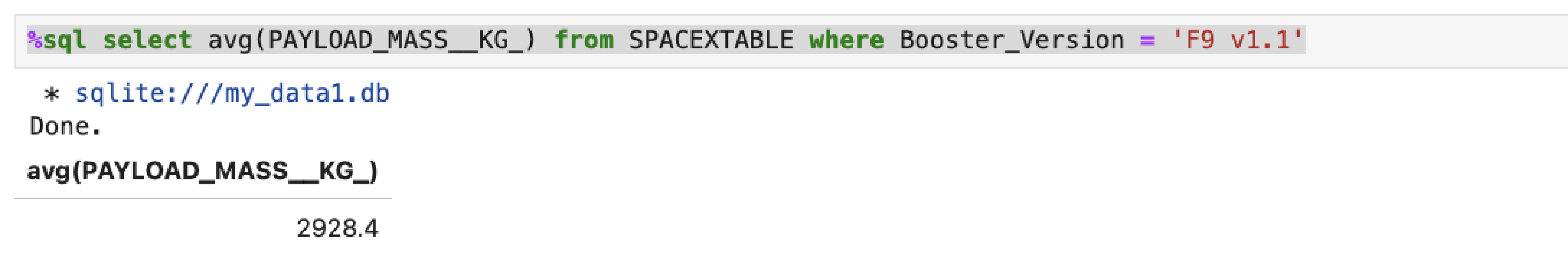
```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 is 2928.4 kg:



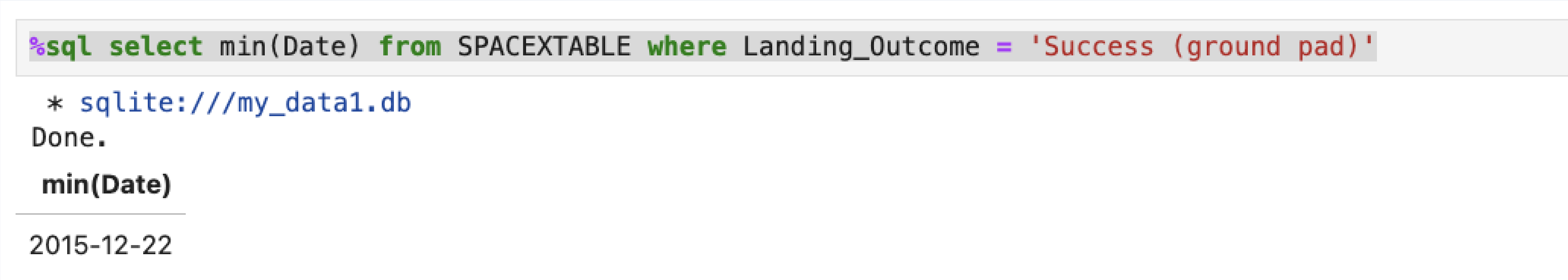
```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version = 'F9 v1.1'
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- The date of the first successful landing outcome on ground pad is December 22, 2015:

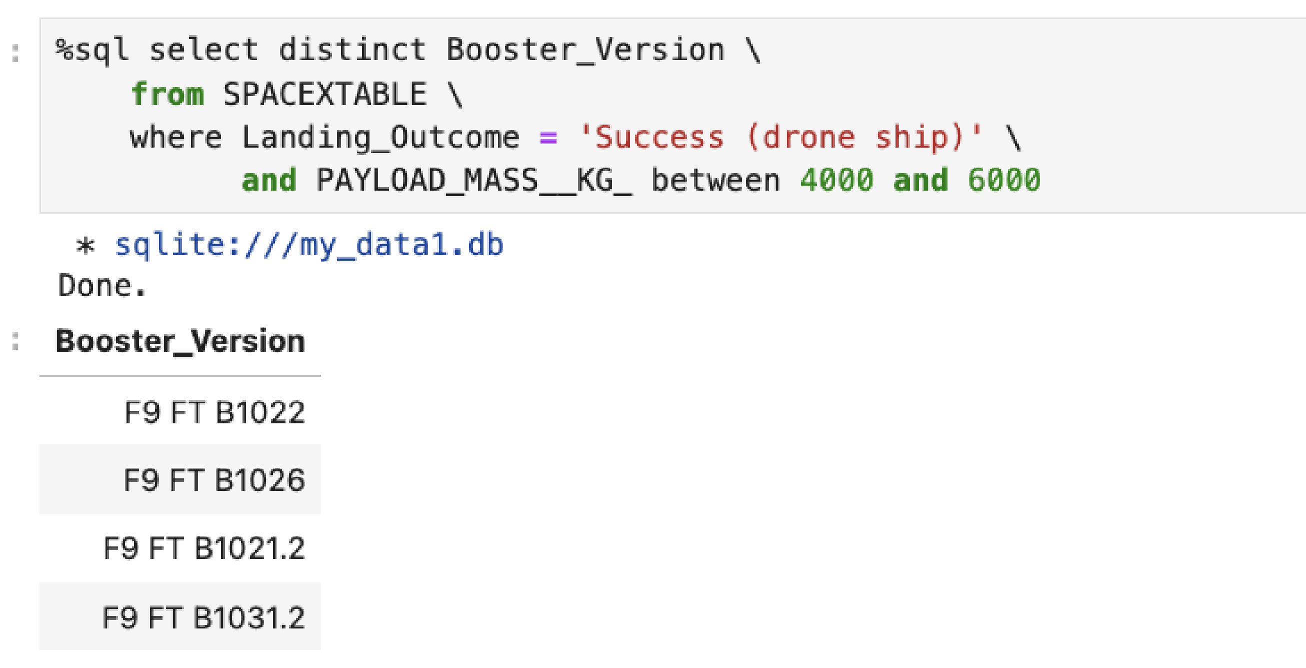


```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'
```

```
 * sqlite:///my_data1.db
Done.
 min(Date)
2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- There are four boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 :



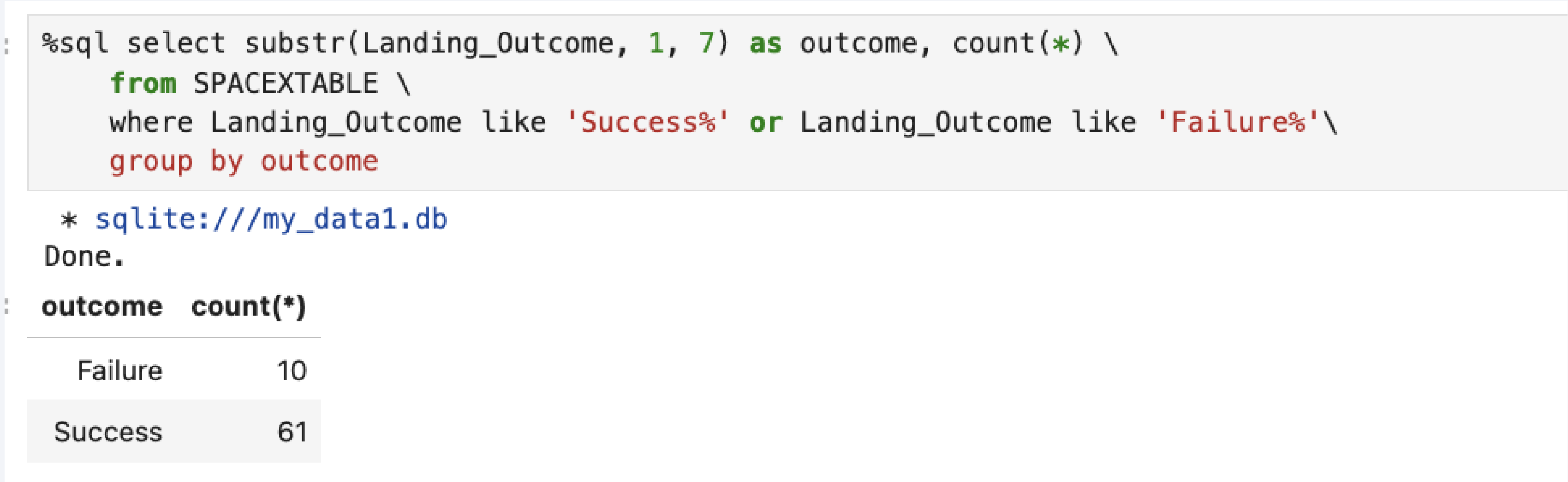
```
%sql select distinct Booster_Version \
    from SPACEXTABLE \
    where Landing_Outcome = 'Success (drone ship)' \
          and PAYLOAD_MASS__KG_ between 4000 and 6000
```

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

- In total, there were 61 successful and 10 failed attempted launches :



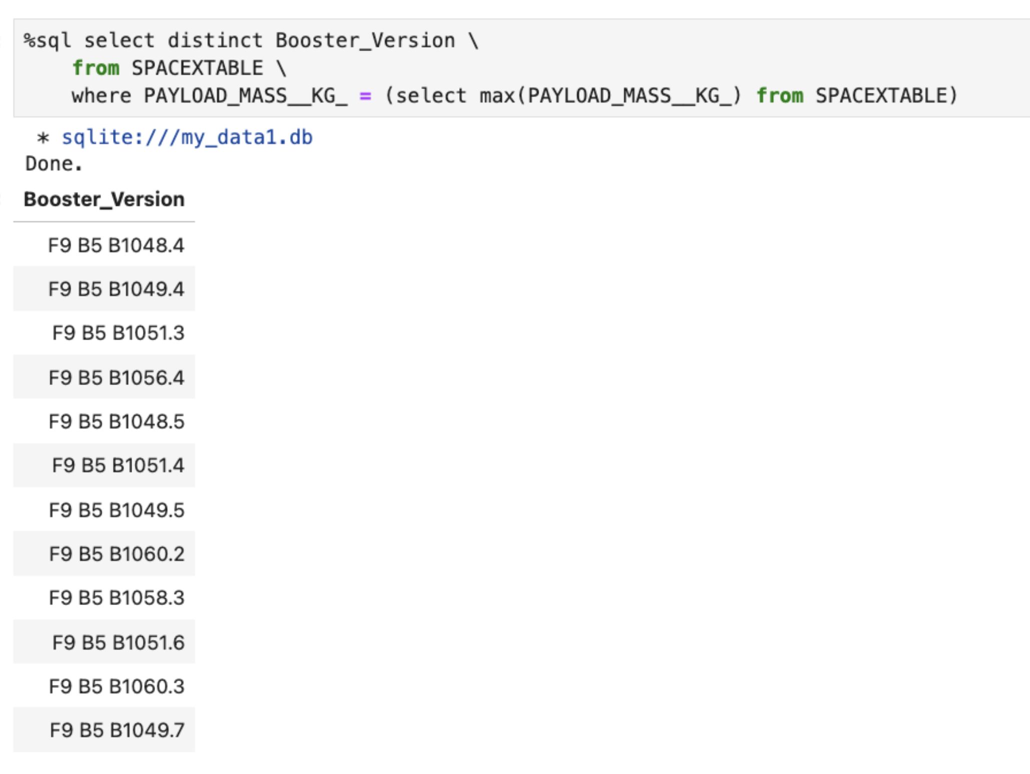
```
%sql select substr(Landing_Outcome, 1, 7) as outcome, count(*) \
    from SPACEXTABLE \
    where Landing_Outcome like 'Success%' or Landing_Outcome like 'Failure%'\
    group by outcome
```

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

outcome	count(*)
Failure	10
Success	61

Boosters Carried Maximum Payload

- Here's the list of boosters which have carried the maximum payload mass :



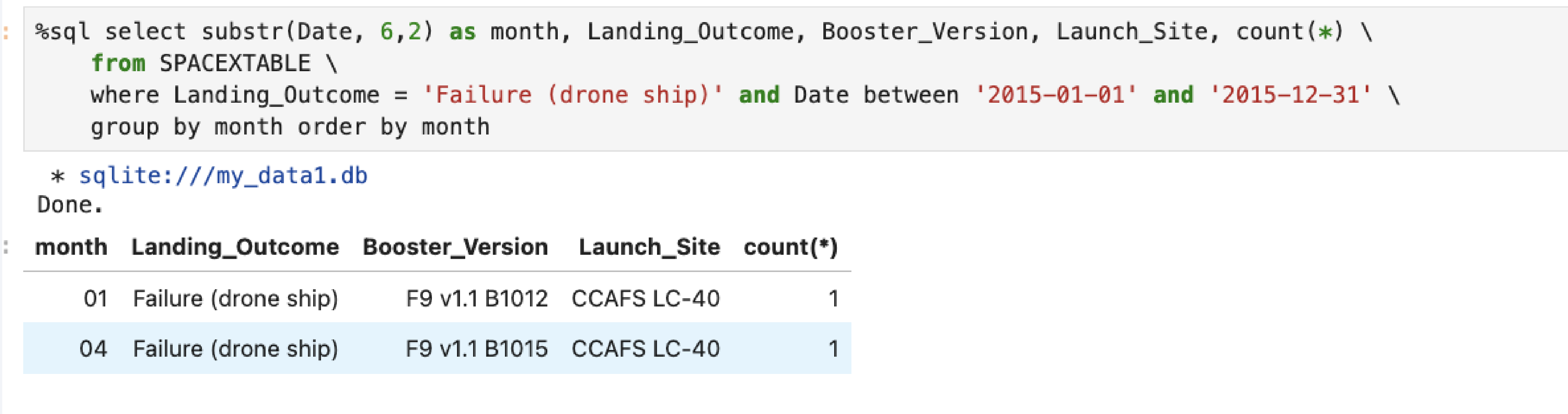
```
%sql select distinct Booster_Version \
    from SPACEXTABLE \
    where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

```
 * 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

- In 2015, there were two failed landing_outcomes in drone ship (in January and in April) :



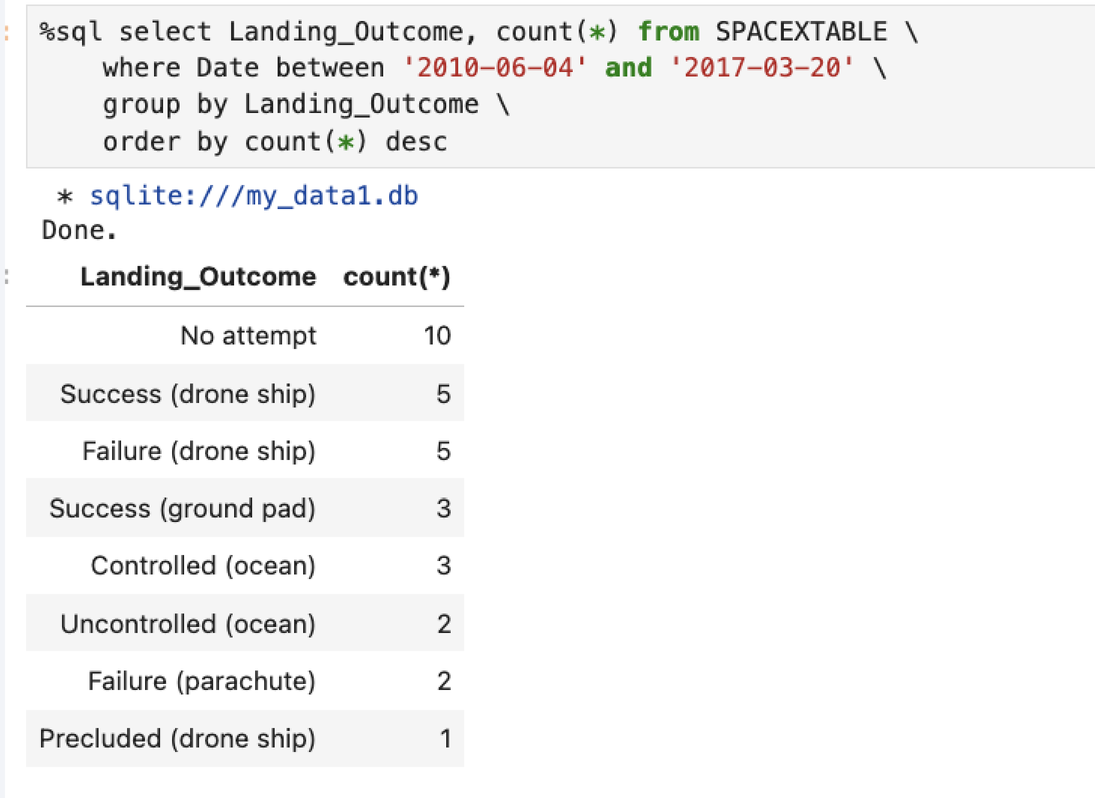
```
%sql select substr(Date, 6,2) as month, Landing_Outcome, Booster_Version, Launch_Site, count(*) \
    from SPACEXTABLE \
    where Landing_Outcome = 'Failure (drone ship)' and Date between '2015-01-01' and '2015-12-31' \
    group by month order by month
```

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

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

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 :



```
%sql select Landing_Outcome, count(*) from SPACEXTABLE \
    where Date between '2010-06-04' and '2017-03-20' \
    group by Landing_Outcome \
    order by count(*) desc
```

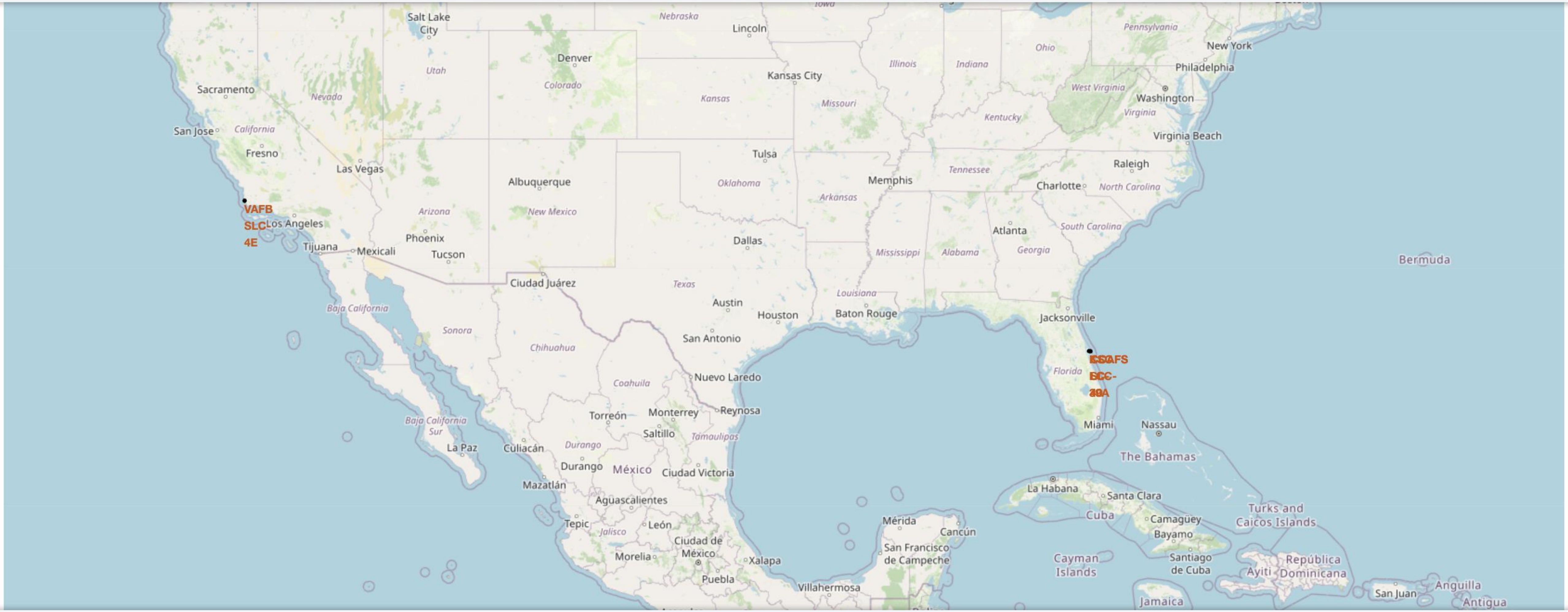
```
 * sqlite:///my_data1.db
Done.
```

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

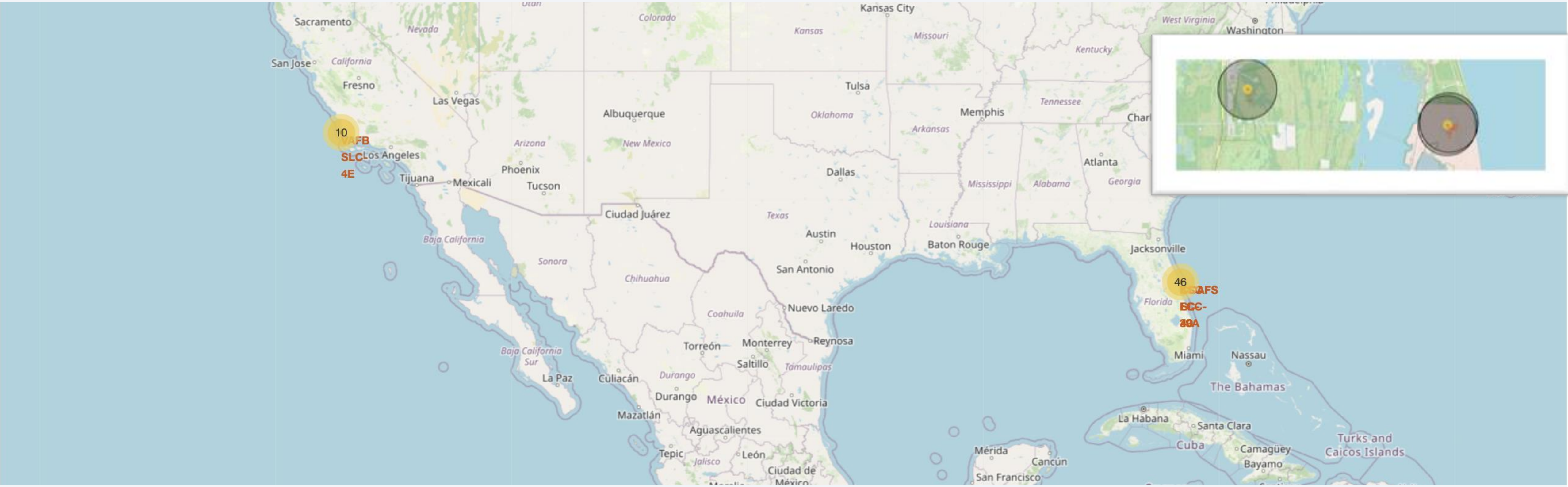
Section 3

Launch Sites Proximities Analysis

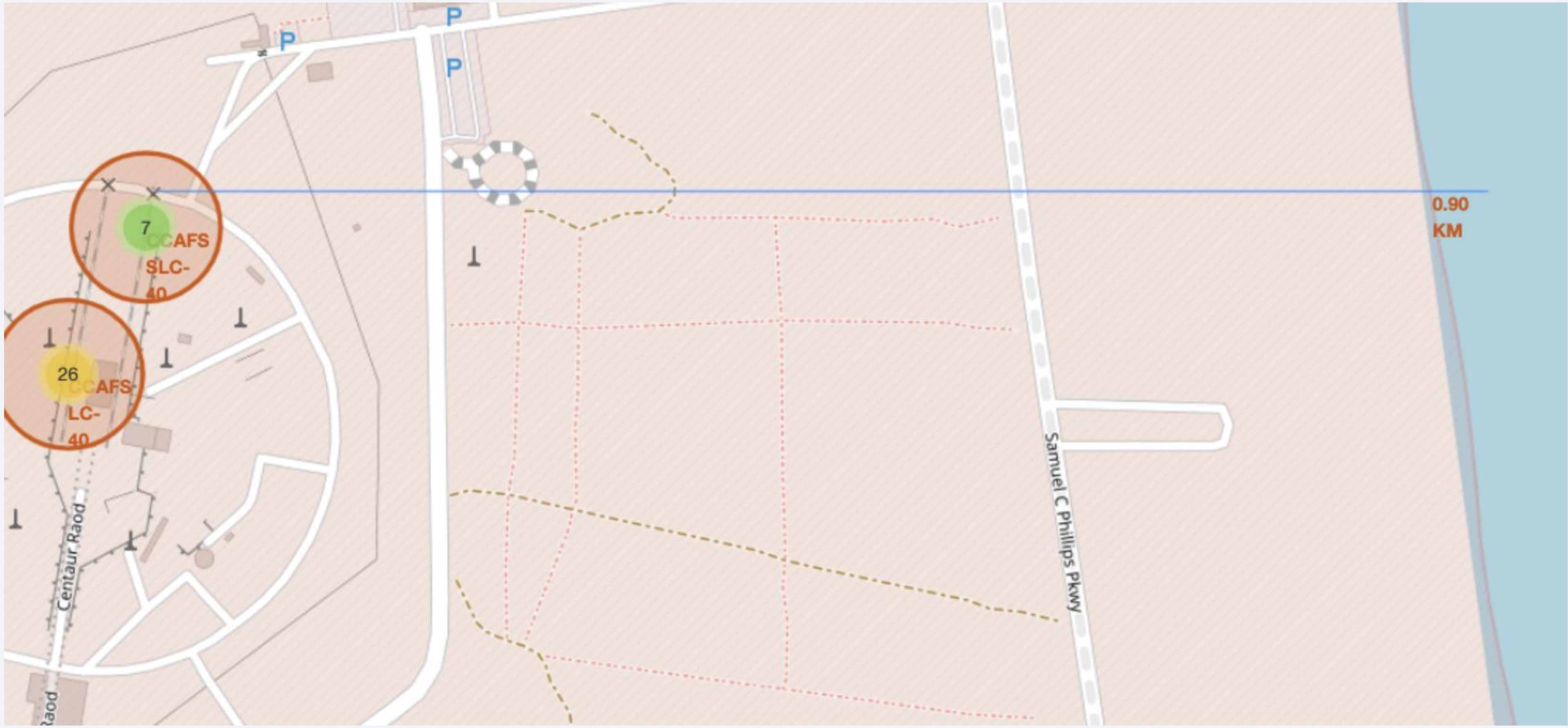
Launch Locations on the Map



Launch Results by Location

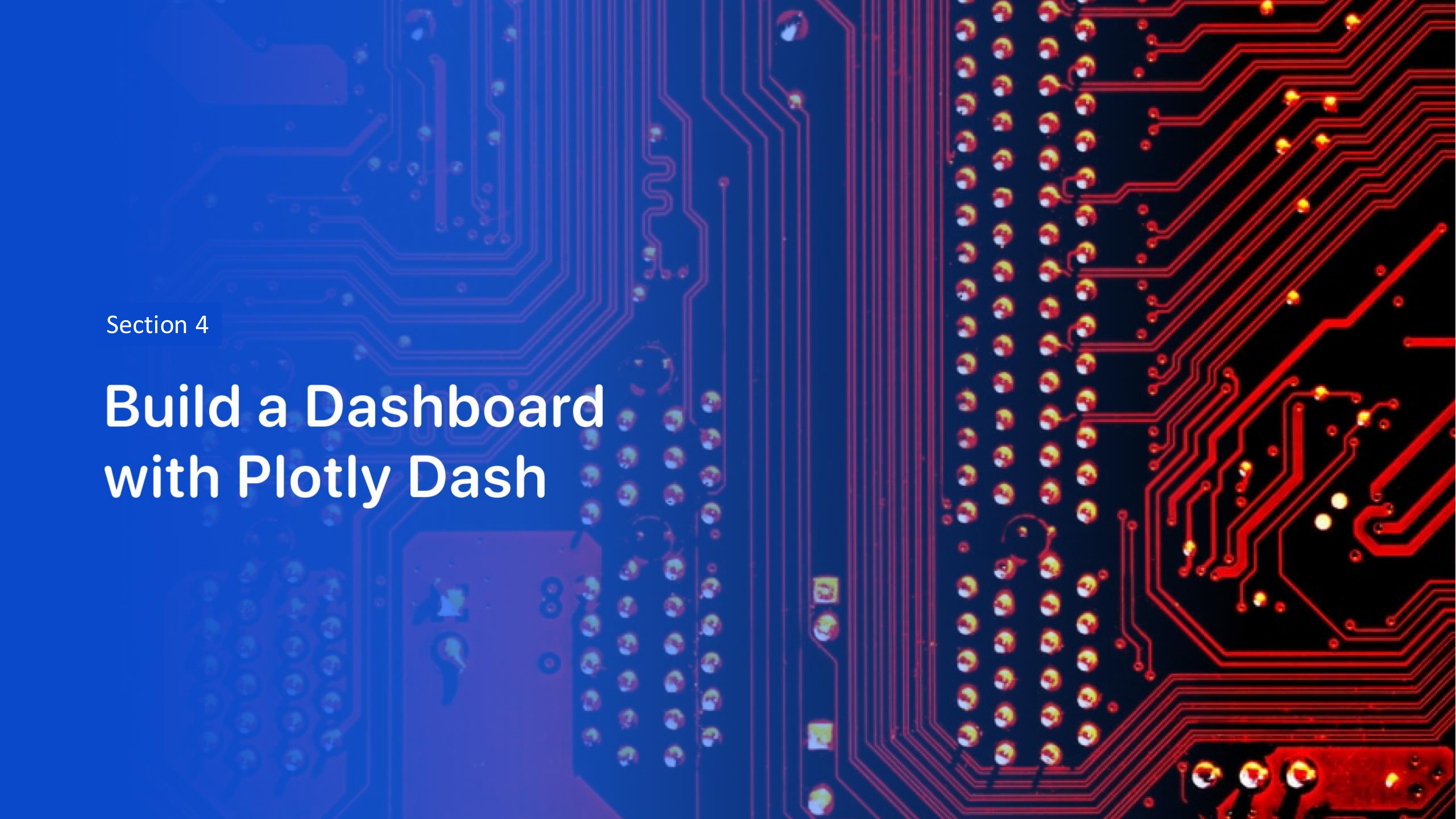


Proximity to the Closest Coastline

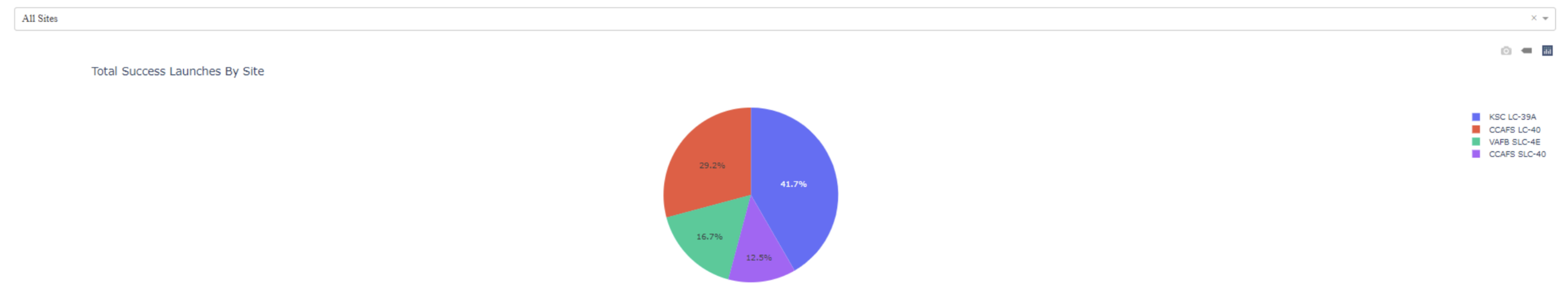


Section 4

Build a Dashboard with Plotly Dash

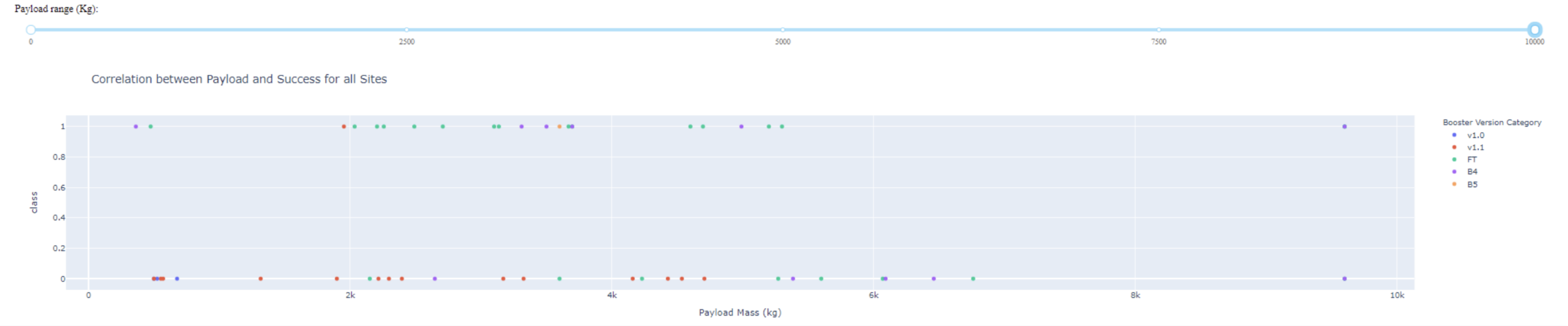


Launch Success Count By Site



- The biggest number of successful launches were made from KSC LC-39A.

Correlation between Payload and Success



Payload range (Kg):

Correlation between Payload and Success for all Sites

The most successful payload sizes are between 2k and 5.5k, with FT as most successful booster version.

Payload sized between 5.5k and 9k never succeeded.

There was also a number of highly unsuccessful attempts around 0.4k.

Section 5

Predictive Analysis (Classification)

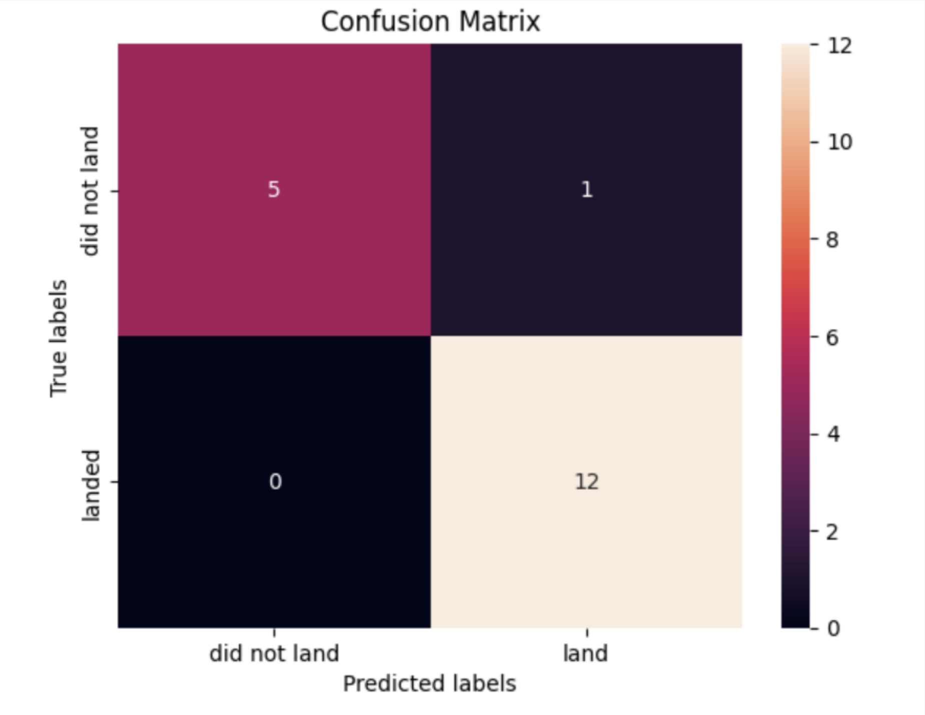
Classification Accuracy

Accuracy by model:

- Decision tree classifier: 94.4% (the winner)
- Logistic regression: 83.3%
- Support vector machine: 83.3%
- K nearest neighbors: 83.3%

Confusion Matrix

- The best performing model (decision tree classifier) showed 94.4% accuracy (12 true positives and 5 true negatives), with no false negatives and one false positive.



Conclusions

- Selected model can be used for predicting launch outcomes.

Thank you!