IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Cristian Distefano
01/08/25

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - ➢ Collect data using SpaceX REST API and web scraping techniques
  - ➢ Wrangle data to construct success/fail outcome variable
  - ➢ Interactive data visualization with Folium
  - ➢ Analyze the data with SQL
  - ➢ Explore launch site success rates due to location using geographical markers
  - ➢ Visualize the launch sites with the most first stage landing success
  - ➢ Build models to predict landing outcomes using logistic regression, support vector machine, decision tree, and K-nearest neighbor
- Summary of all results
  - ➢ KSC LC-39A has the highest success rate among all landing sites
  - ➢ Most launch sites are positioned near the equator and close to the coast
  - ➢ All predictive analysis models performed similarly, but the decision tree model outperformed the rest

Introduction

- Background

  SpaceX, a leader in the space industry, strives to make space travel affordable for everyone. SpaceX advertises the Falcon 9 rocket launch for a crew of private citizens for $62 million per launch. Other providers provide the same service but with a cost upwards of $165 million each. The reason SpaceX can provide such a low price is because of their ability to recover and reuse the first stage of its Falcon 9 rocket. Through our data analysis, we will be able to determine if the first stage will land. To do this, we can use public data and machine learning models to predict whether SpaceX can reuse the first stage.

- We want to explore:
  - How can payload mass, launch site, number of flights, and orbits affect first-stage landing success?
  - What is the rate of successful landing over time?
  - What is the best predictive model for successful landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - Collect data using SpaceX REST API and web scraping techniques
- Perform data wrangling
    - Filtering the data, handling missing values, and applying one hot encoding to prepare the data for analysis and modeling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Build, tune, and evaluate classification models to predict landing outcomes

Data Collection – SpaceX API

- We utilized SpaceX API to collect data, then cleaned the data and wrangled and formatted the data.

- GitHub URL to the SpaceX API call notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/01_SpaceX_Data_Collection_API.ipynb

Steps:

1. Request data from SpaceX API (rocket launch data)
2. Decode response using .json() method and convert into a dataframe using .json_normalize()
3. Request information about the launches from SpaceX API using custom functions
4. Create dictionaries from the data
5. Create dataframe from the dictionary
6. Filter dataframe to contain only Falcon 9 launches
7. Replace missing values of Payload Mass with calculated .mean() method
8. Export data to csv file

Data Collection - Scraping

- We applied web scraping techniques to scrape Falcon 9 launch records with Beautiful Soup

- GitHub URL to the Web Scraping notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/02_SpaceX_Web_Scraping.ipynb

Steps:

1. Request the Falcon 9 launch data from Wikipedia
2. Create BeautifulSoup object from HTML response
3. Extract column names from HTML table header
4. Collect data from parsing HTML tables
5. Create dictionary from the data
6. Create dataframe from the dictionary
7. Export data to csv file

Data Wrangling

- Steps
  - Perform EDA and determine data labels
  - Calculate:
    - # of launches per site
    - # and occurrence of orbit
    - # and occurrence of mission outcome per orbit
  - Create binary landing outcome column
  - Export data to csv file

GitHub URL to the Data Wrangling notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/03_SpaceX_Data_Wrangling.ipynb

- Landing Outcome
  - Landing was not always successful
  - True Ocean: represents a successful landing on the ocean
  - False Ocean: represents an unsuccessful landing on the ocean
  - True RLTS: represents a successful landing on a ground pad
  - False RLTS: represents an unsuccessful landing on a ground pad
  - True ASDS: represents a successful landing on a drone ship
  - False ASDS: represents an unsuccessful landing on a drone ship
  - Outcomes have been converted into a 1 for a success and a 0 for an unsuccessful landing

EDA with Data Visualization

- Charts utilized:
  - Flight Number vs. Payload
  - Flight Number vs. Launch Site
  - Payload Mass (kg) vs. Launch Site
  - Payload Mass (kg) vs. Orbit type

GitHub URL to the EDA w/ Data Visualization notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/05_SpaceX_EDA_Data_Visualization.ipynb

- Analysis
  - Viewed relationship by using scatter plots. If a relationship existed, we used those variables for machine learning prediction models.
  - Showed comparisons among discrete categories with bar charts. Bar charts showed the relationships among the categories and a measured value.

EDA with SQL

- Display:
  - Names of unique launch sites
  - 5 records where launch site begins with 'CCA'
  - Total payload mass carried by boosters launched by NASA
  - Average payload mass carried by booster version F9 v1.1.
- List:
  - Date of first successful landing on ground pad
  - Names of boosters which had success landing on drone ship and have payload mass greater than 4,000kg but less than 6,000kg
  - Total number of successful and failed landings
  - Names of booster versions which have carried the max payload
  - Failed landing outcomes on drone ship, and their booster version and launch site for the months in the year 2015
  - Count of landing outcomes between 2010-06-04 and 2017-03-20

GitHub URL to the EDA w/ SQL notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/04_SpaceX_EDA_SQL.ipynb

Build an Interactive Map with Folium

- Markers Indicating Launch Sites
  - Added blue circle at NASA Johnson Space Center's coordinates with a popup label that showed its name by using its longitude and latitude coordinates
  - Added red circles at all launch site coordinates with a popup label that showed its name using its longitude and latitude coordinates
- Colored Markers of Launch Outcomes
  - Added colored markers: green for successful and red for unsuccessful at each launch site to show which launch sites have high success rates
- Distance Between a Launch Site to Proximities
  - Added colored lines to show distance between launch site CCAFS SLC-40 and its proximity to the nearest coastline, railway, highway, and city

GitHub URL to the Interactive Map with Folium notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/06_SpaceX_Interactive_Visual_Analytics_Folium.ipynb

Build a Dashboard with Plotly Dash

- Dropdown List with Launch Sites
  - Allow user to select all launch sites or a certain launch site of their choosing
- Pie Chart Showing Successful Launches
  - Allow user to see successful and unsuccessful launches as a percent of the total launches
- Slider of Payload Mass Range
  - Allow user to select payload mass range
- Scatter Chart Showing Payload Mass vs. Success Rate by Booster Vision
  - Allow user to see the correlation between Payload and Launch Success

GitHub URL to the Interactive Map with Plotly Dash notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/07_SpaceX_Interactive_Visual_Analytics_Plotly.py

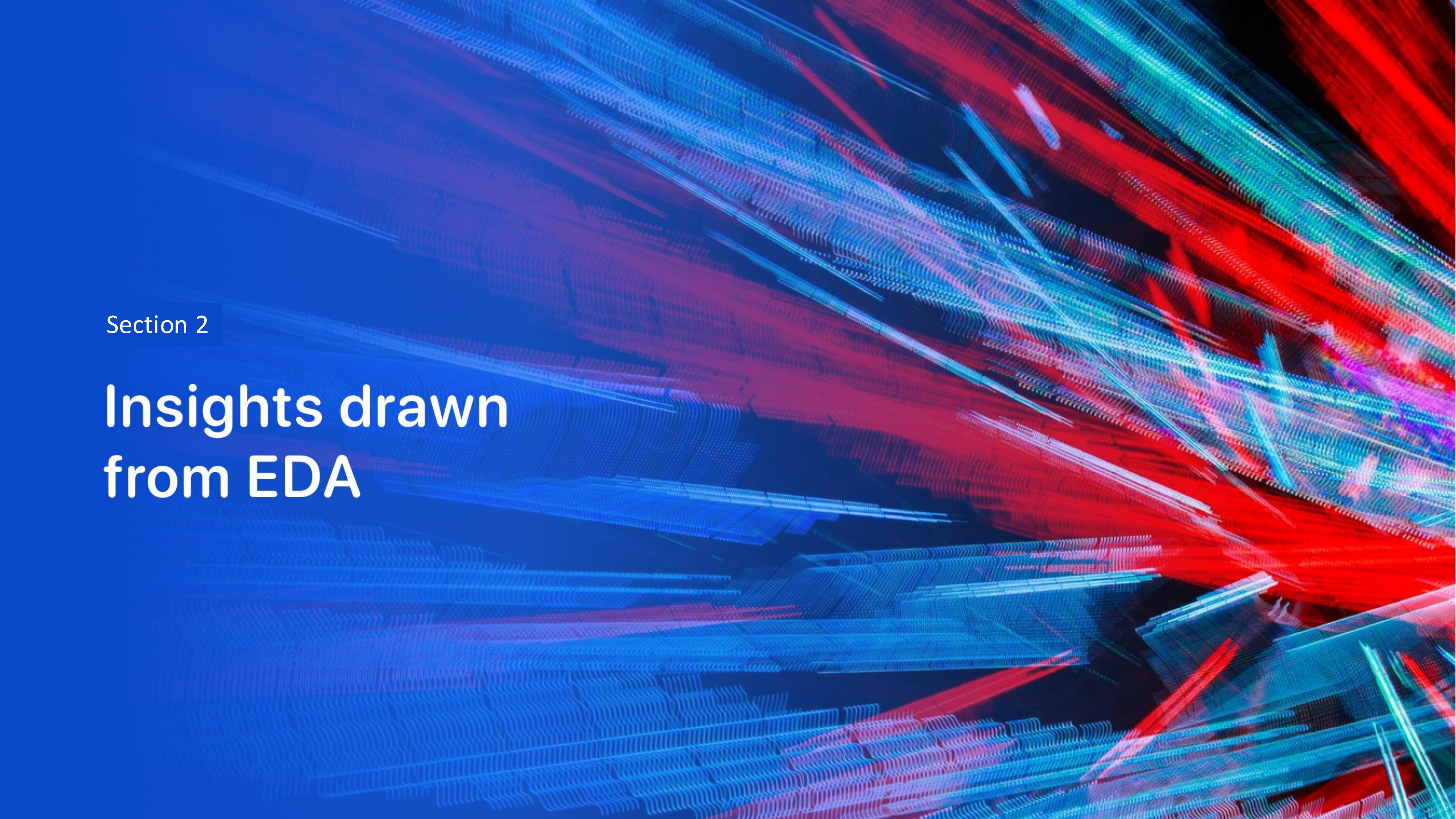
Predictive Analysis (Classification)

- Process:
  - Created NumPy array from the Class column
  - Standardized the data with StandardScaler; Fit and transformed the data
  - Split the data using train_test_split
  - Created a GridSearchCV object with cv = 10 for parameter optimization
  - Applied GridSearchVC on varying algorithms: Logistic Regression, Support Vector Machine, Decision Tree, K-Nearest Neighbor
  - Calculated accuracy on the test data using .score() for all models
  - Assessed the confusion matrix for all models
  - Identified the best model using Jaccard_Score, F1_Score, and Accuracy

GitHub URL to the Machine Learning Prediction notebook: https://github.com/cld22007/Applied-Data-Science-Capstone-Project/blob/main/08_SpaceX_Machine%20Learning%20Prediction.ipynb

Summary of Results

- Exploratory Data Analysis Results
  - Launch success has improved over time
  - KSC LC-39A has the highest success rate among landing sights
  - Orbits ES-L1, GEO, HEO and SSO have a 100% success rate
- Interactive Analytics
  - Most launch sites are near the equator and close to the coast
  - Launch sites are far from cities, highways, and railways in case of failed launches but they are just close enough to transport people and materials to and from the launch site
- Predictive analysis results
  - Decision Tree model is the best predictive model for this dataset

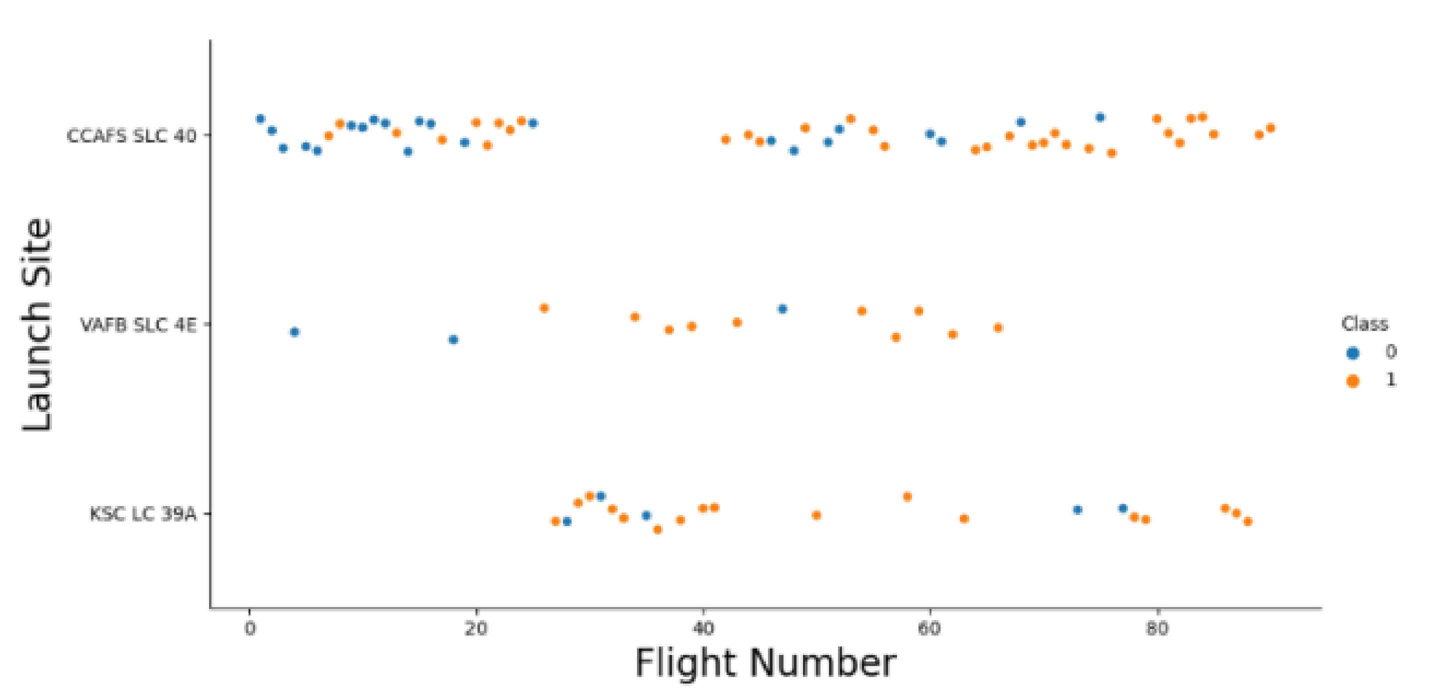


Section 2

Insights drawn from EDA

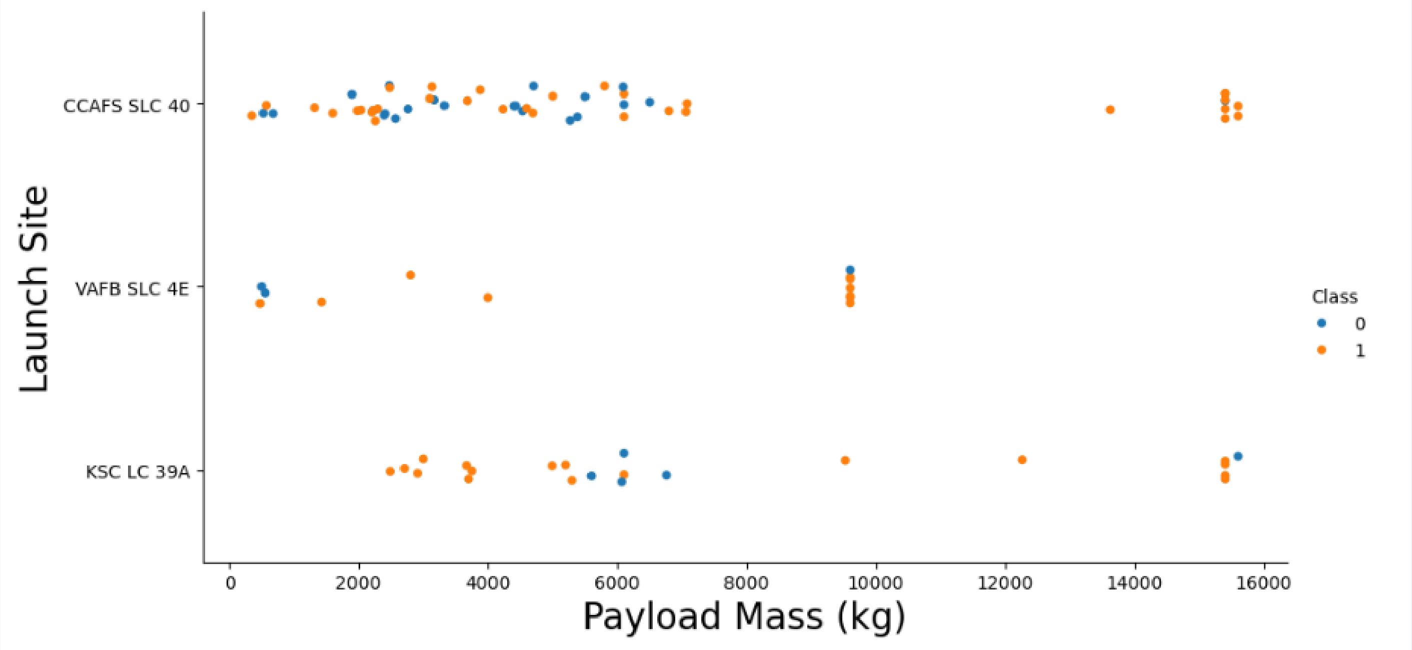
Flight Number vs. Launch Site

- Findings:
  - Earlier flights had a lower success rate (blue)
  - Later flights had a higher success rate (orange)
  - About half of the launches were from the CCAFS SLC 40 launch site
  - We can conclude that newer launches have higher success rates



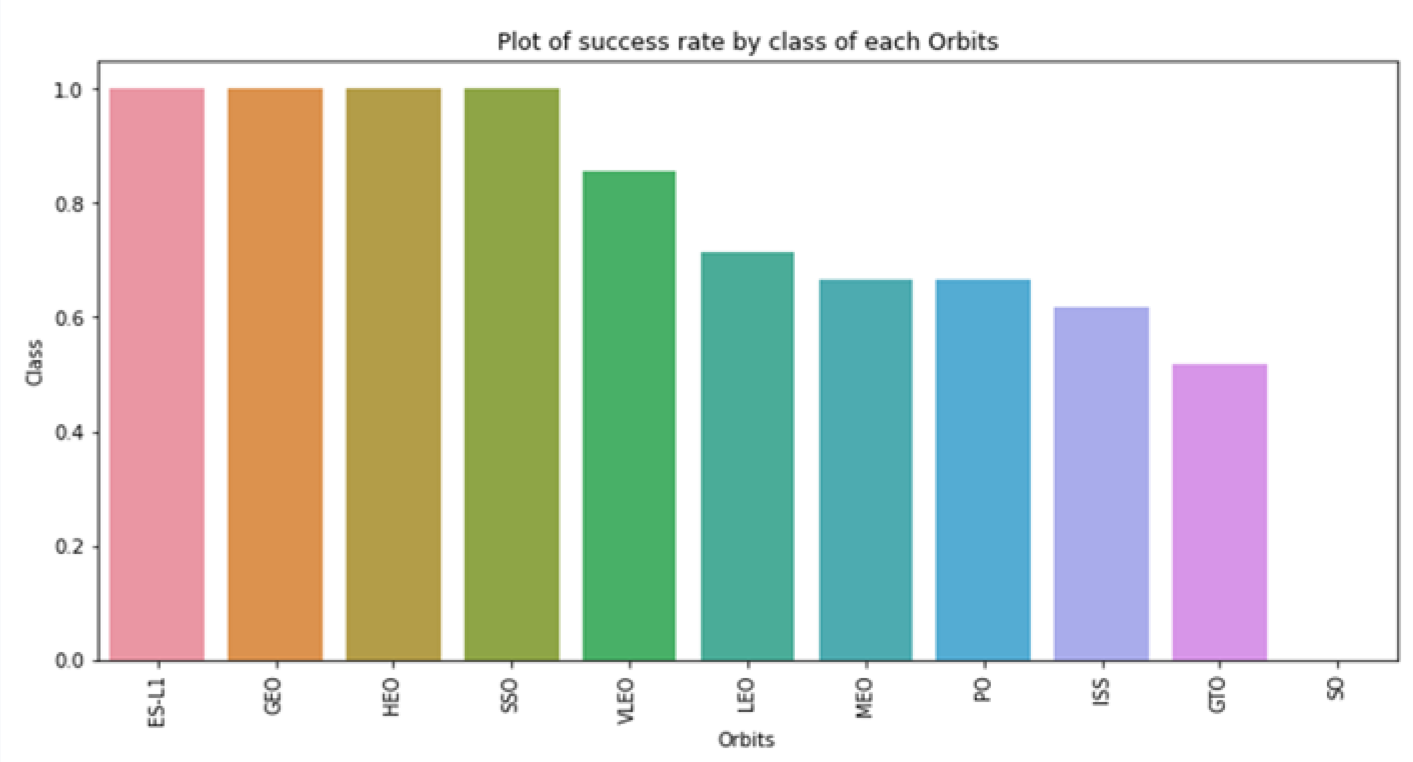
Payload vs. Launch Site

- Findings:
  - The higher the payload mass (kg), the higher the success rate
  - Most launches with a payload mass greater than 8,000kg were successful
  - VAFB SKC 4E has not launched any ships with a payload mass greater than 10,000 kg
  - 100% of KSC LC 39A launches less than 5,500 kg were successful



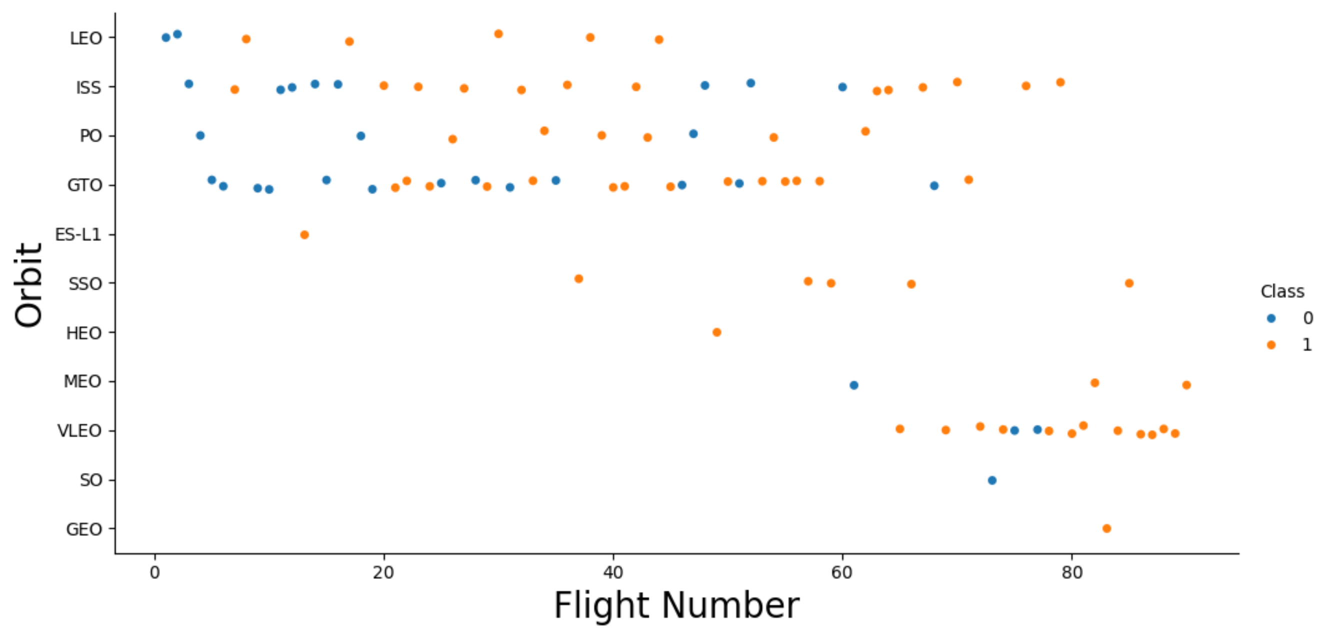
Success Rate vs. Orbit Type

- Findings:
  - 100% Success Rate: ES-L1, GRO, HEO, SSO
  - 85% Success Rate: VLEO
  - 80%-50% Success Rate: LEO, MEO, PO, ISS, GTO
  - 0% Success Rate: SO



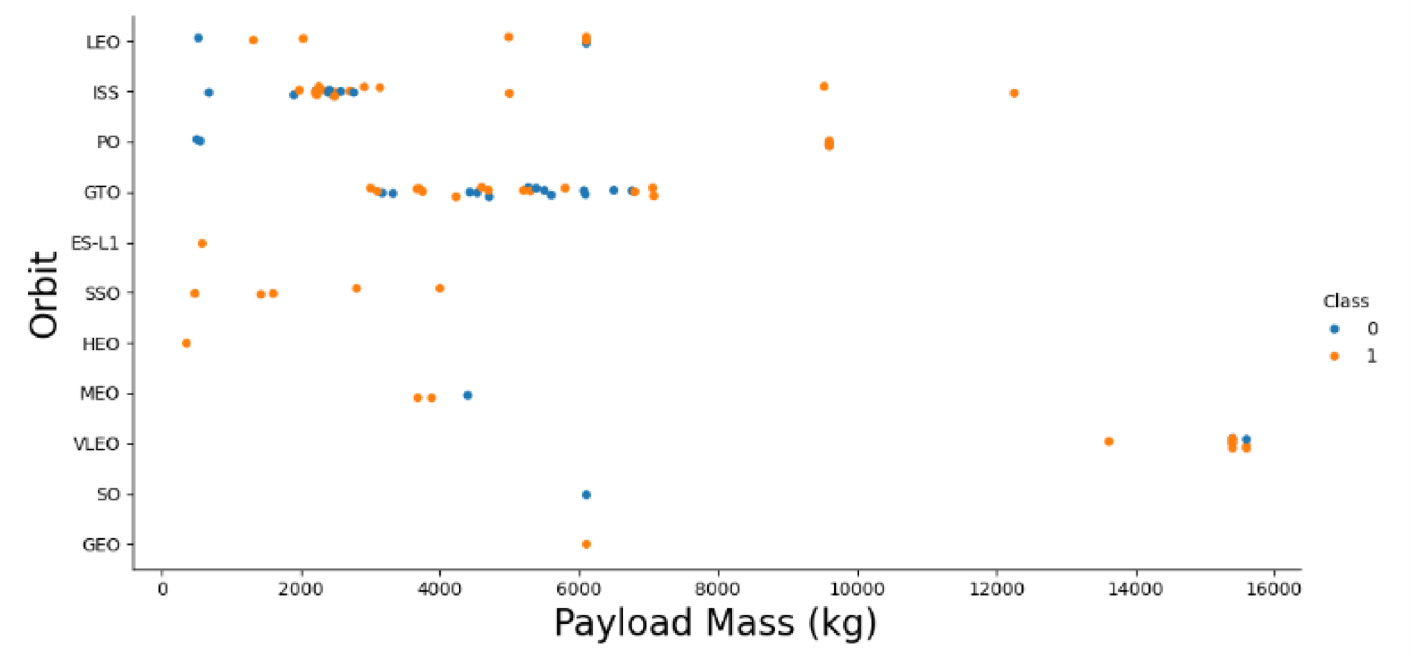
Flight Number vs. Orbit Type

- Findings:
  - The success rate increases with the number of flights for each orbit
  - This is particularly visible in the LEO orbit
  - The GTO orbit seems to follow this trend albeit to a limited extent



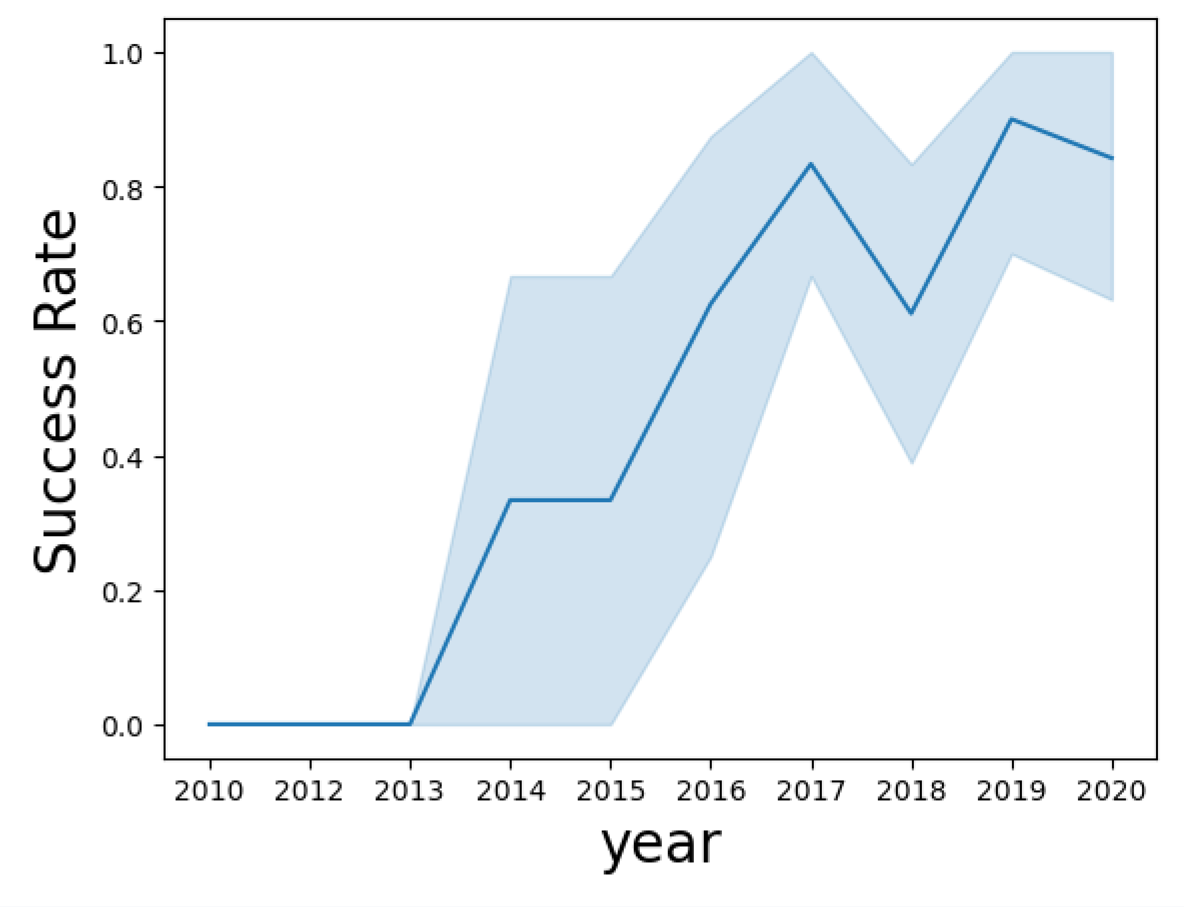
Payload vs. Orbit Type

- Findings:
  - Heavier payloads have more success landing in the LEO, ISS, and PO orbits
  - SSO has a 100% success rate with all its launches having lighter payload masses compared to the other orbits



Launch Success Yearly Trend

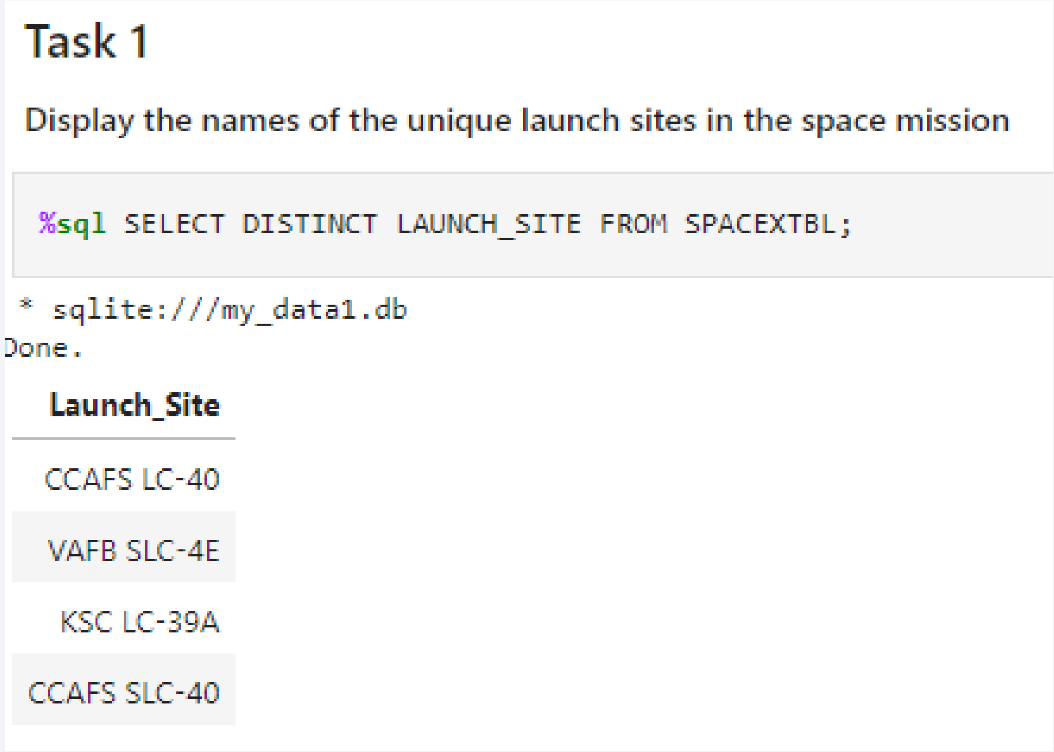
- Findings:
  - Overall, we can see that success rate rises over time
  - A significant improvement in success rate can be observed from 2013-2017 and from 2018-2019
  - Slight decreases in success rate can be observed from 2017-2018 and 2019-2020



All Launch Site Names

- Launch Site Names
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40

- We used the key word DISTINCT to show only the unique launch sites from the SpaceX data



Task 1

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

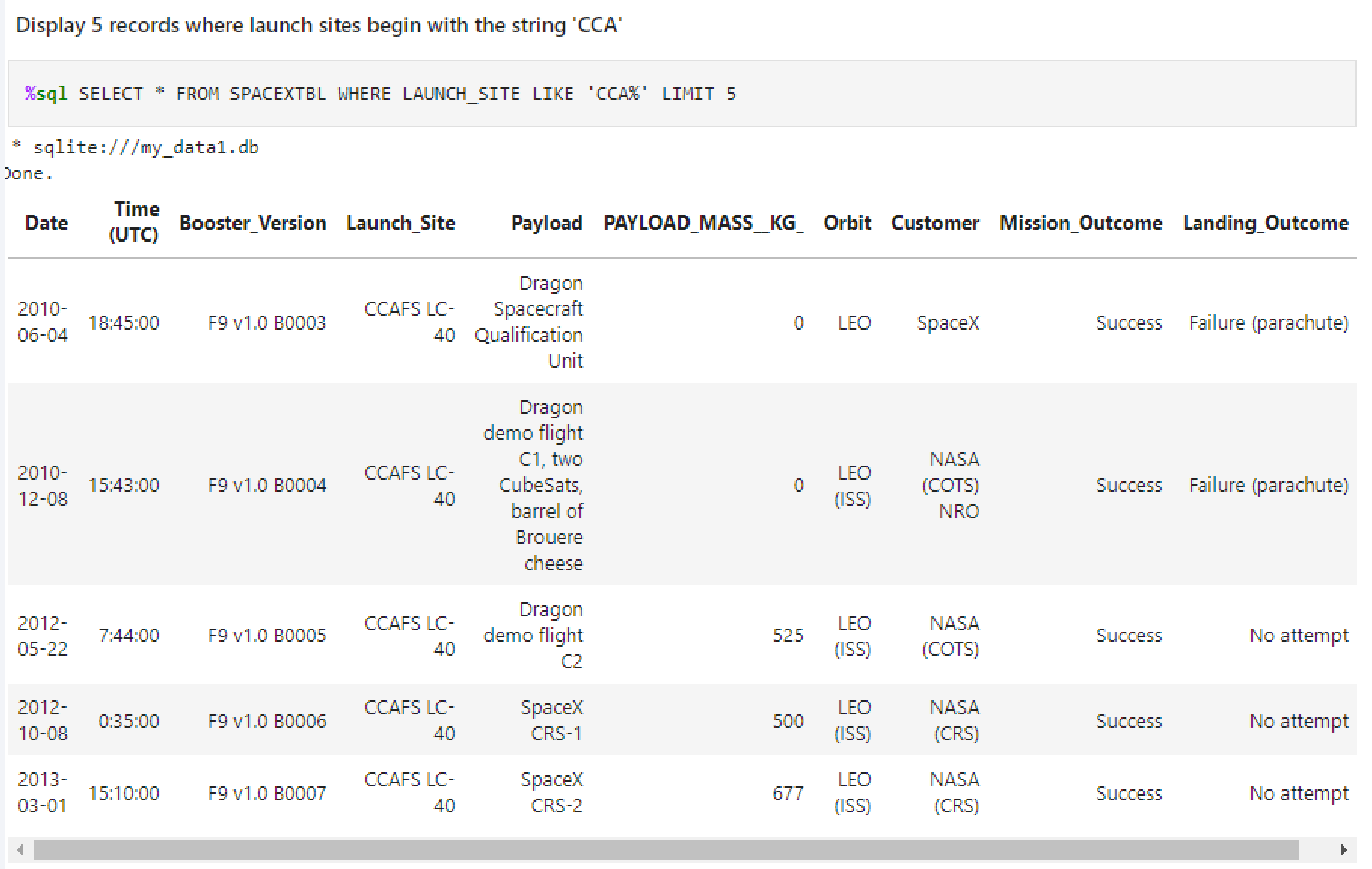
```
%sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL;
```

```
 * 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'

- Using the following query, we were able to display 5 records where the launch site begins with 'CCA'



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

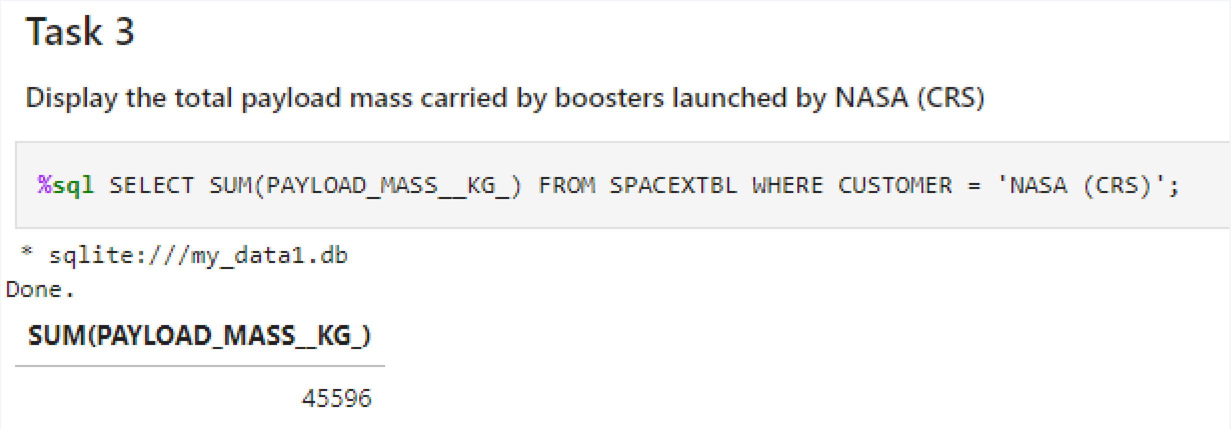
```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

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

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

- Using the following query, we calculated the total payload carried by boosters launched by NASA, giving us 45596 kg



Task 3

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

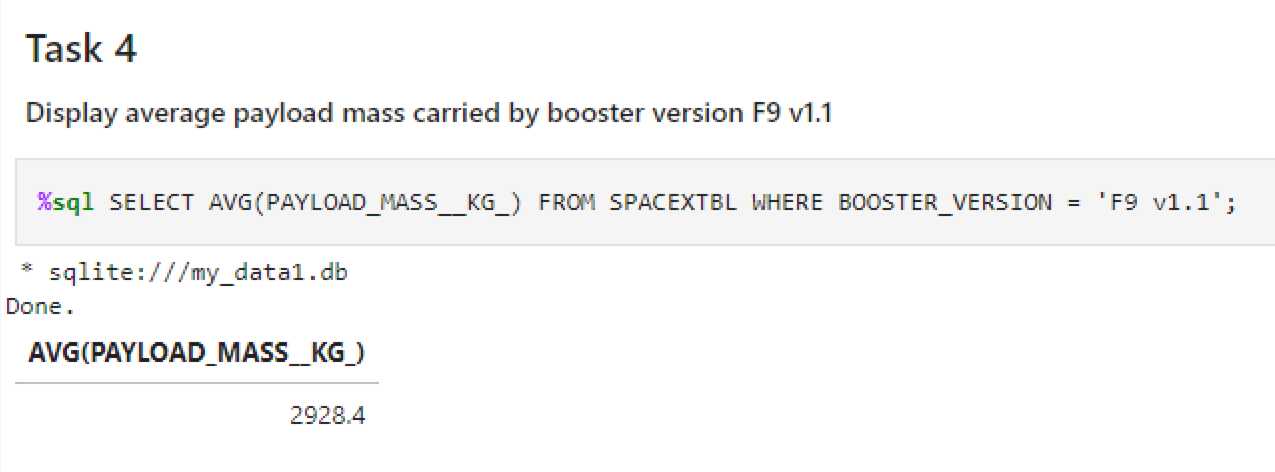
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)';
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Using the following query, we calculated the average payload mass carried by booster version F9 v1.1, giving us 2928.4 kg



Task 4

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

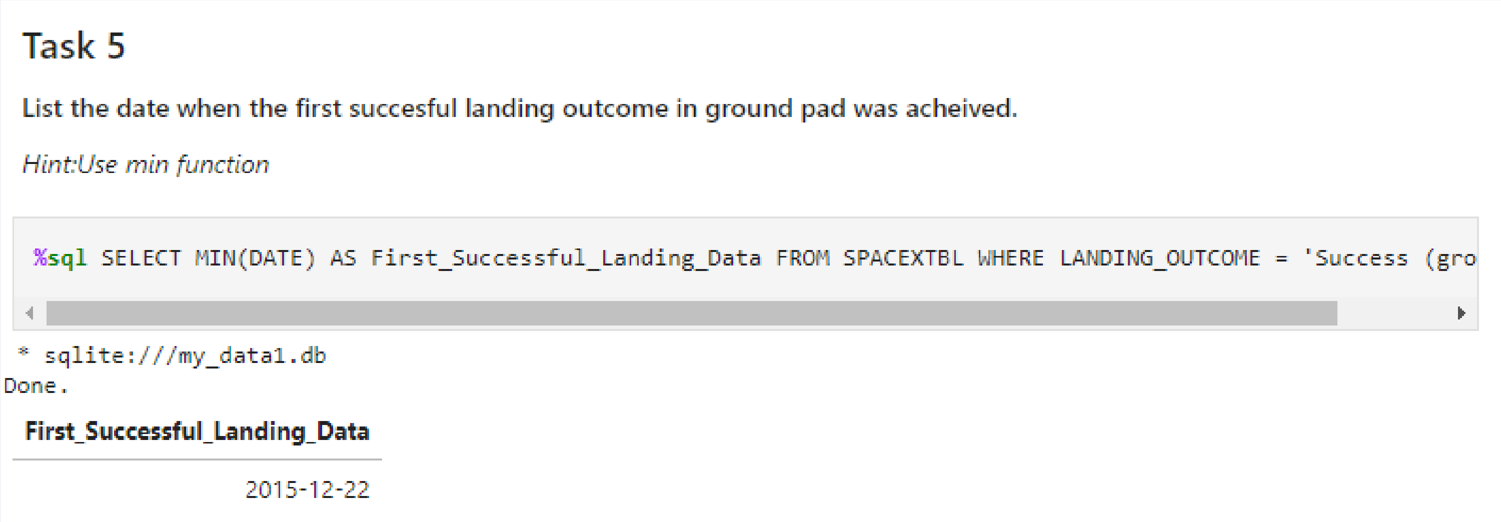
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Using the following query, we observed that the date of the first successful landing on a ground pad was December 22nd, 2015



Task 5

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

Hint:Use min function

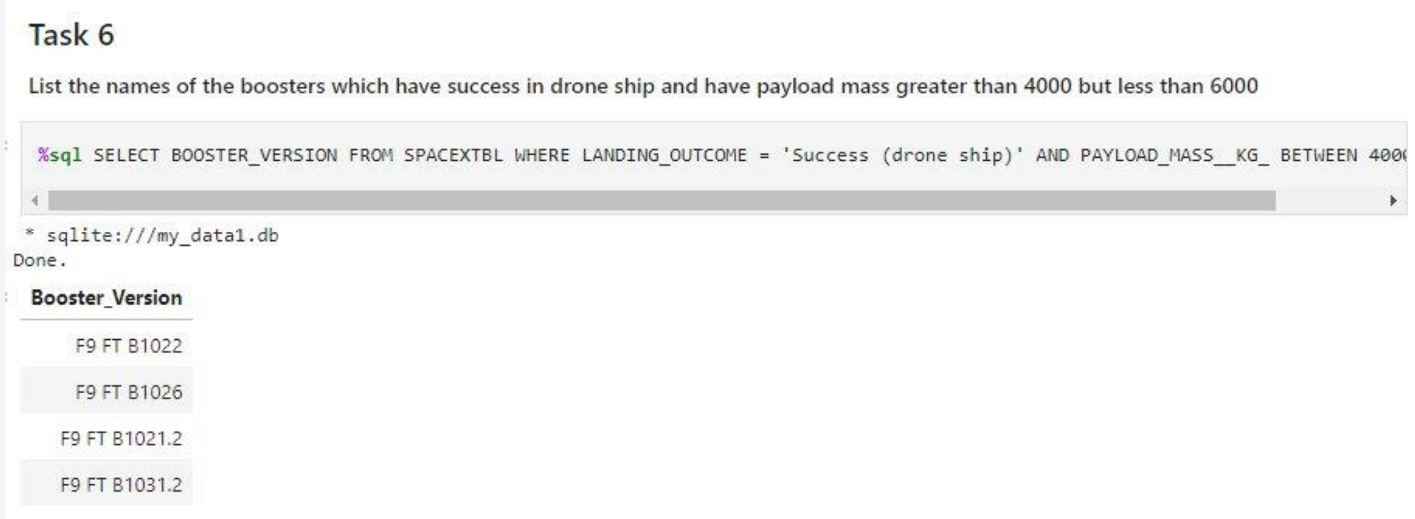
```
%sql SELECT MIN(DATE) AS First_Successful_Landing_Data FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (gro
```

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

First_Successful_Landing_Data
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using the following query, we utilized the WHERE clause to filter for boosters that have successfully landed on a drone ship and applied the AND condition to determine successful landing with a payload mass between 4000kg and 6000kg



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

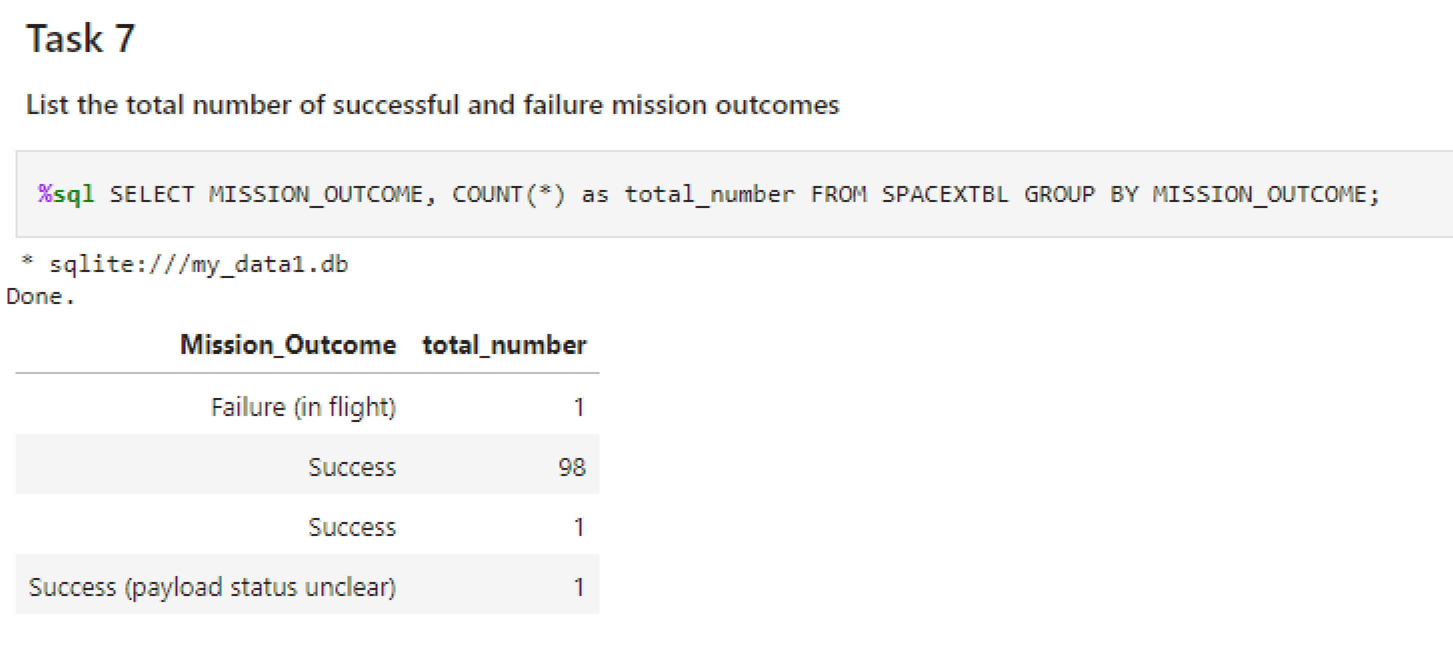
```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 400
```

```
 * 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

- Using the following query, we filtered through the data to find the successful and failed mission outcomes, giving us 100 total successes and 1 failure



Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(*) as total_number FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

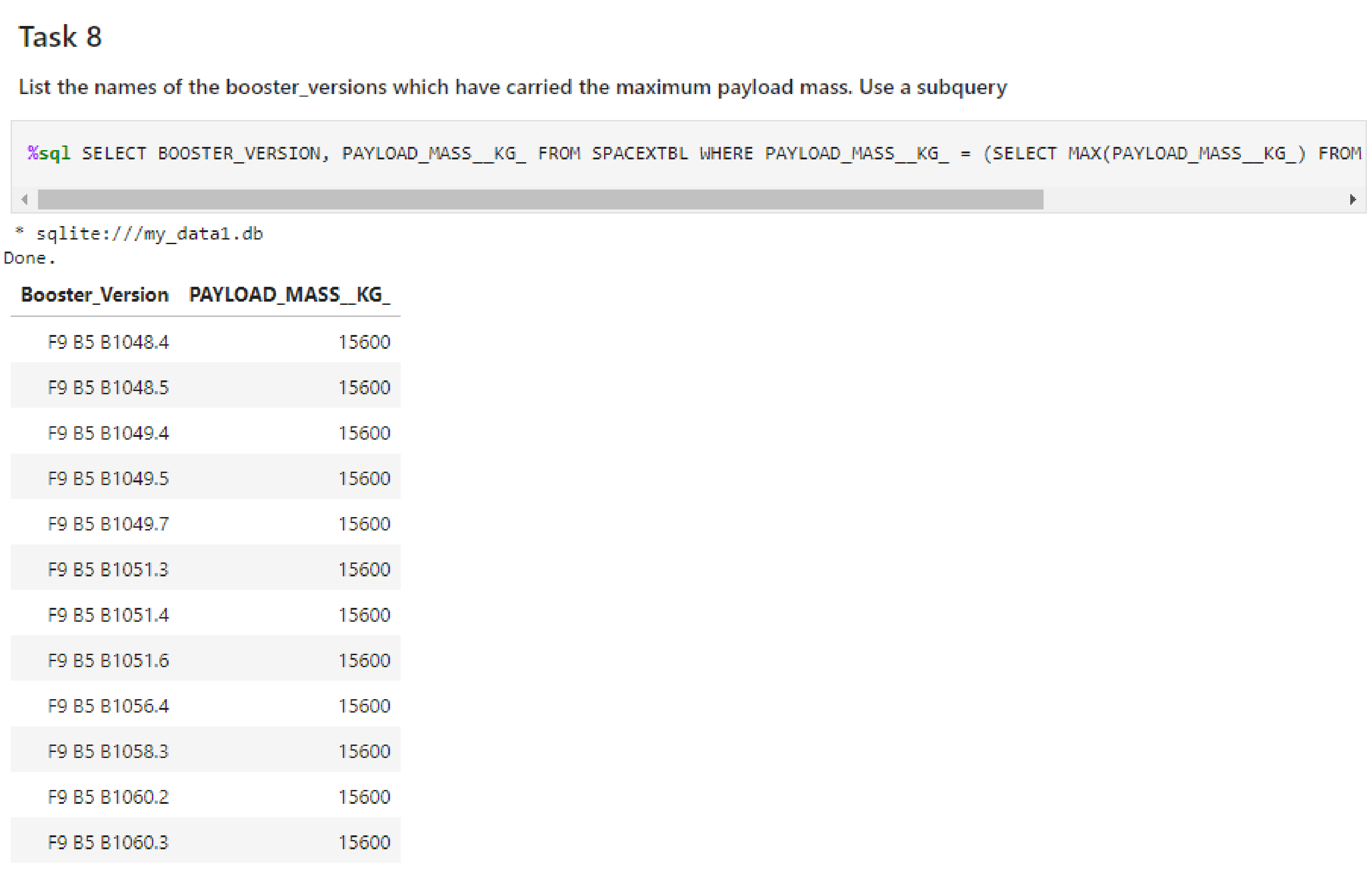
```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	total_number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- Here, we found the boosters that carried the maximum payload using a subquery in the WHERE clause with the MAX() function

- Listed in the image are the previously mentioned boosters



Task 8

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

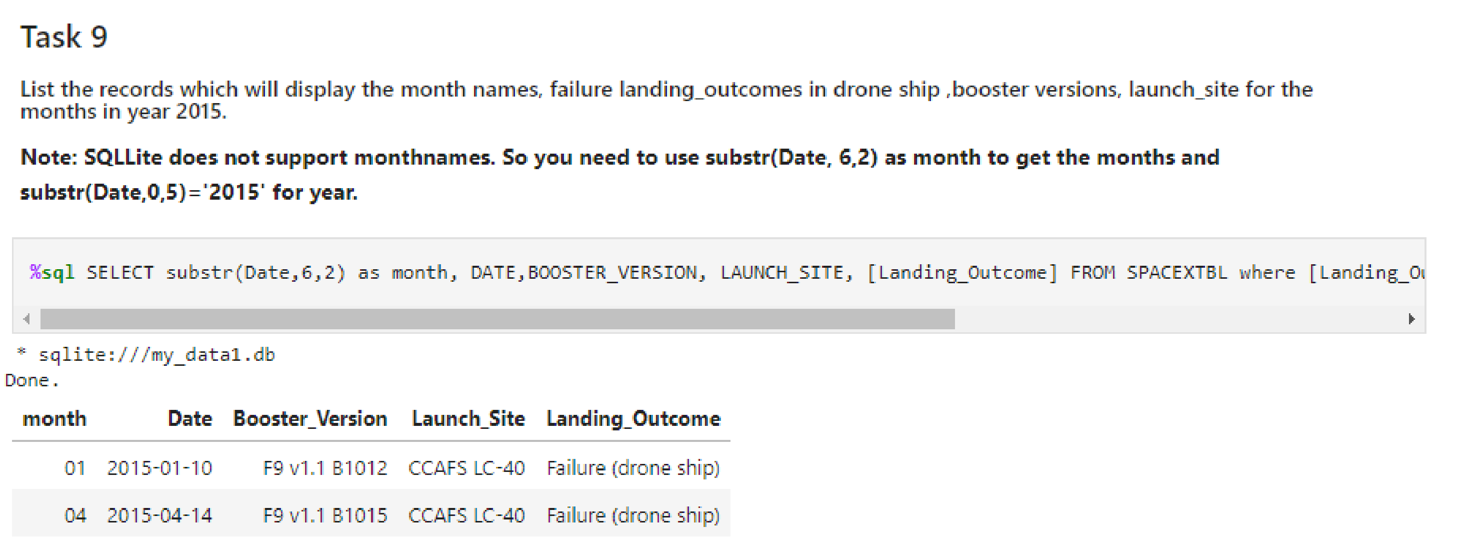
```
%sql SELECT BOOSTER_VERSION, PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM
```

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

2015 Launch Records

- Using the following query, we were able to obtain the Month, Date, Booster Version, Launch Site, and Landing Outcome of the 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.

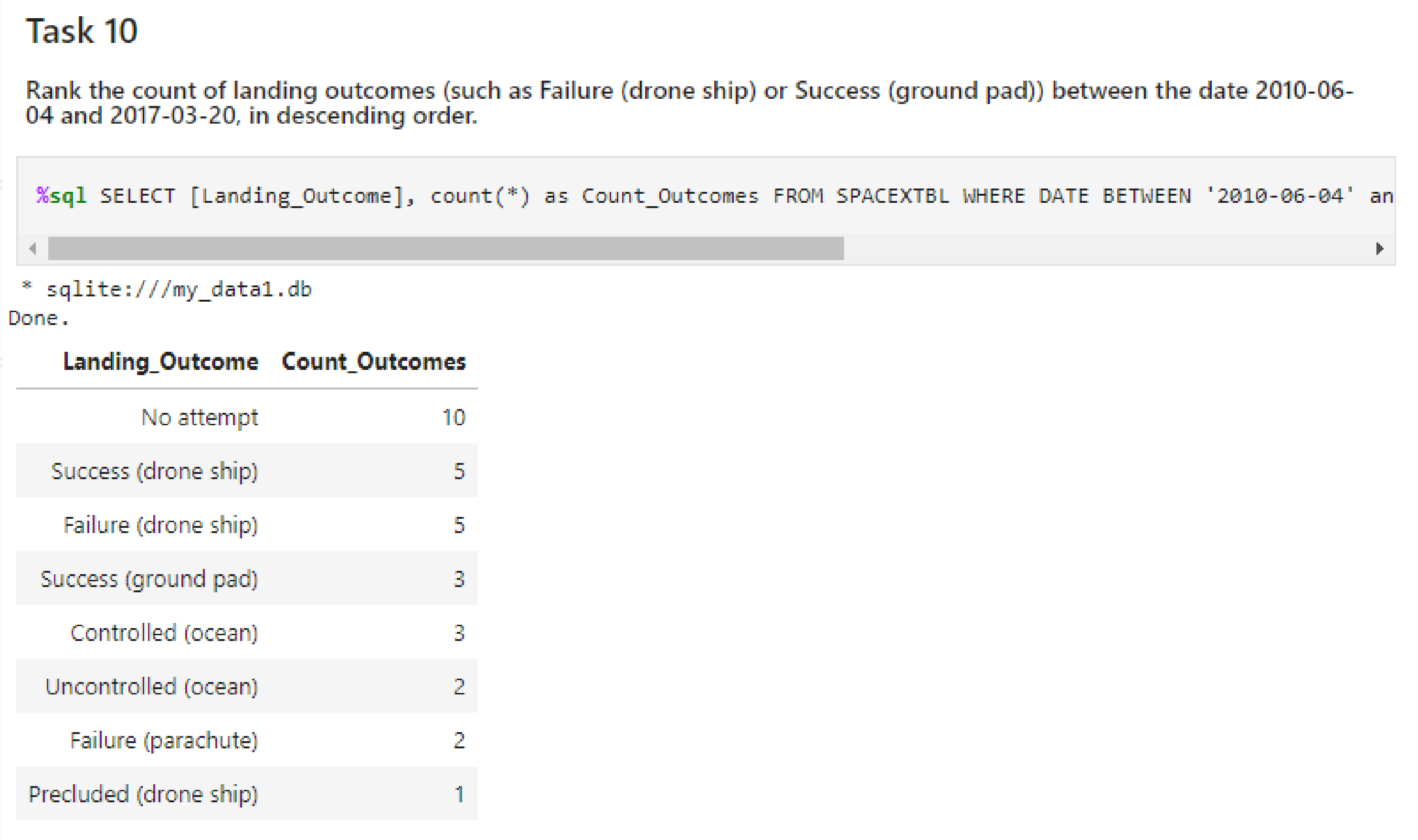
```
%sql SELECT substr(Date,6,2) as month, DATE,BOOSTER_VERSION, LAUNCH_SITE, [Landing_Outcome] FROM SPACEXTBL where [Landing_Ou
```

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

month	Date	Booster_Version	Launch_Site	Landing_Outcome
01	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Using the following query, we selected the Landing Outcomes with the COUNT of landing outcomes from the data, using the WHERE clause to filter for landing outcomes BETWEEN 2010-06-04 and 2017-03-20
- We used the GROUP BY clause to group the landing outcomes and the ORDER BY clause to specify the descending order



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.

```
%sql SELECT [Landing_Outcome], count(*) as Count_Outcomes FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' an
```

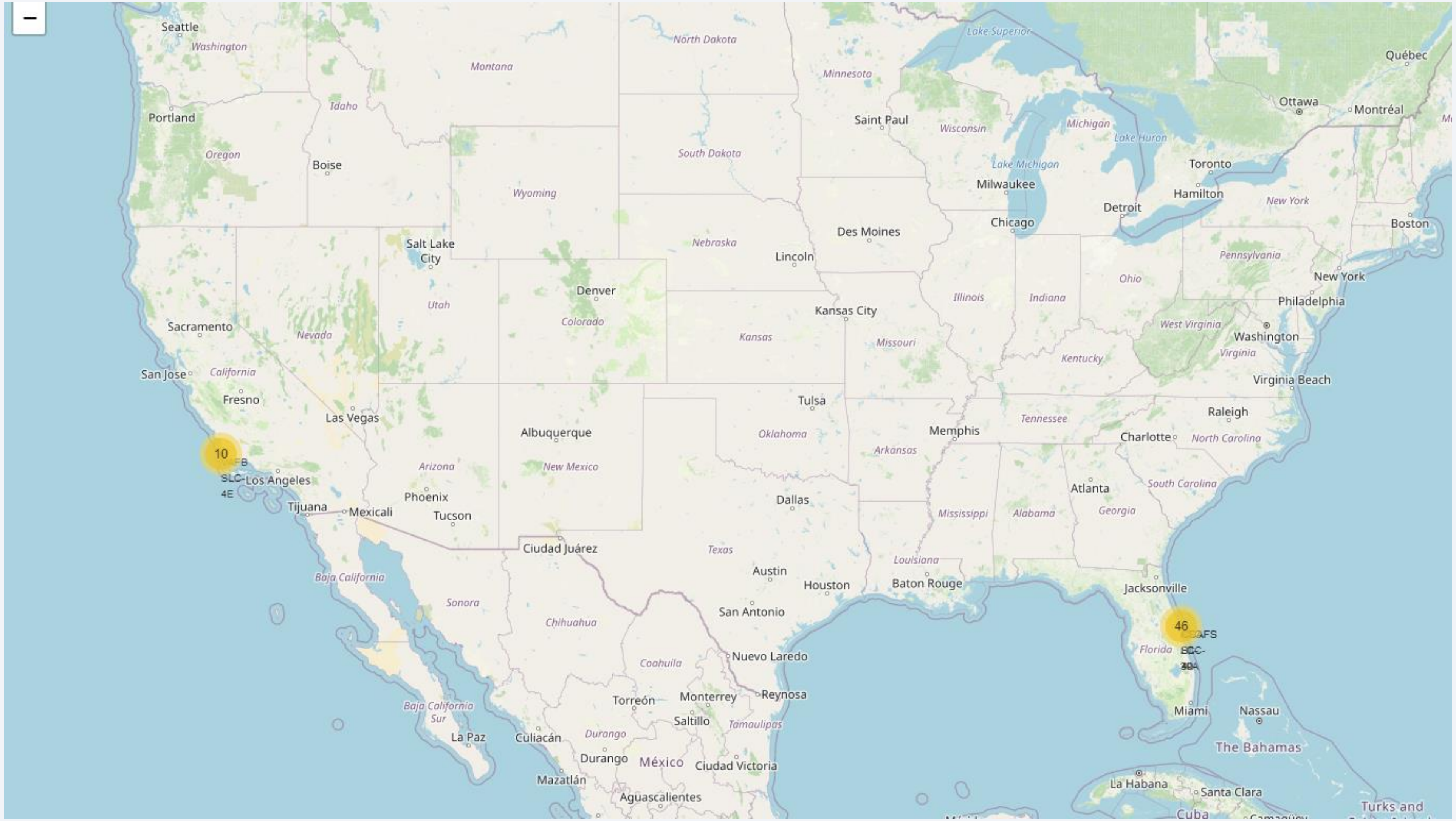
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	Count_Outcomes
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

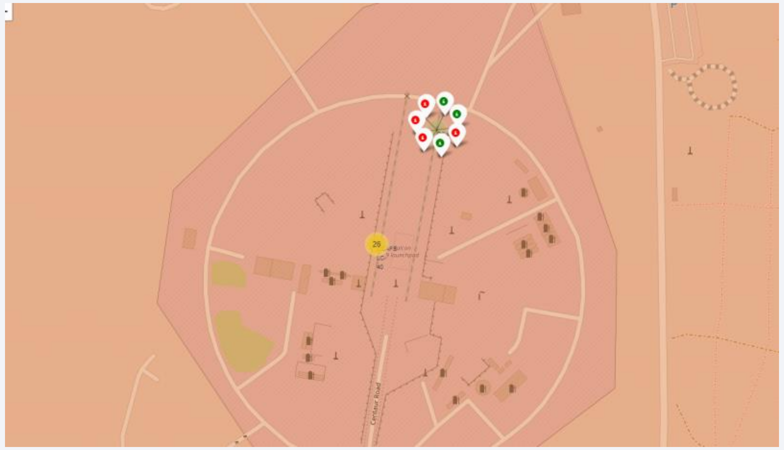
Launch Sites Proximities Analysis

Launch Sites

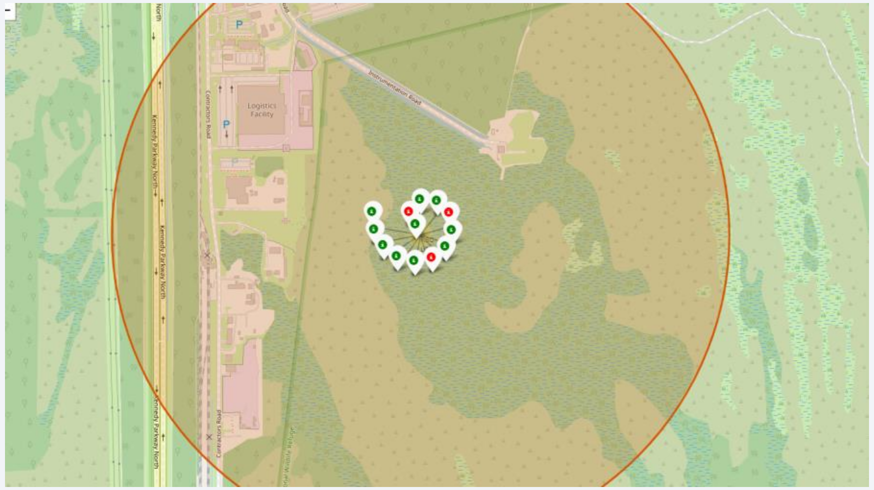


- As highlighted, we can see all the launch sites are:
  - Close to the coast
  - Near the equator

Launch Outcomes



Launch Site CCAFS SLC-40

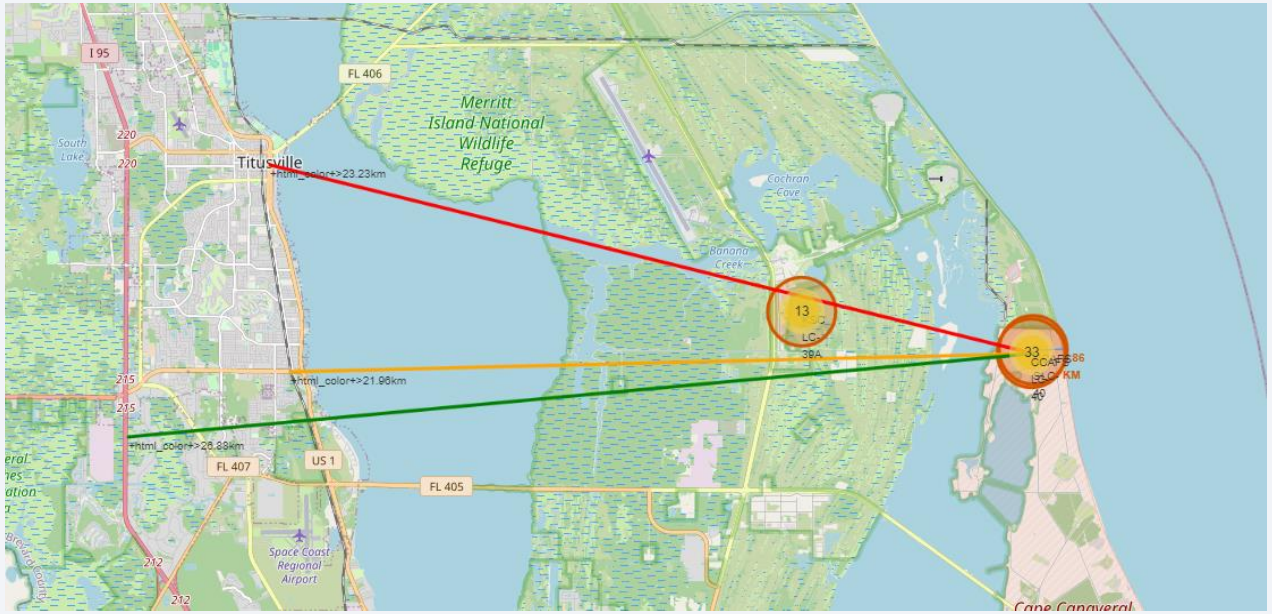


Launch Site KSC LC-39A

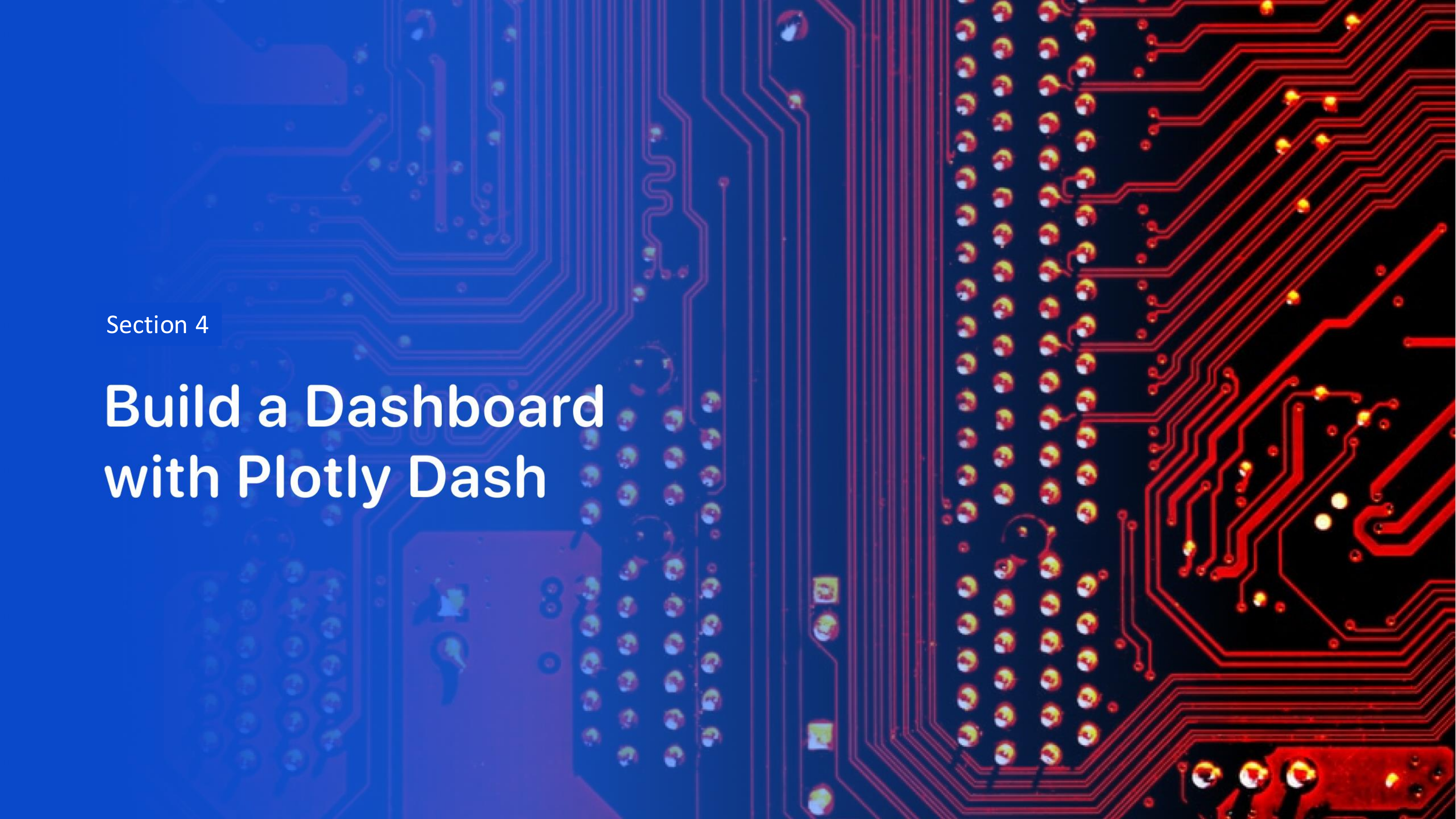
- Shown in the images to the left are launch sites CCAFS SLC-40 and KSC LC-39A, both launch sites from Florida
- The Green markers indicate successful launches
- The Red markers indicate unsuccessful launches
- As observed, launch site CCAFS SLC-40 has a success rate of 3/7 or 42.9%

Launch Site Proximities

- CCAFS SLC-40
  - .86 km from nearest coastline
  - 21.96km from nearest railway
  - 23.23km from nearest city
  - 26.88km from nearest highway



- Proximity Significance:
  - Closeness to coasts ensures that failed launches or spent rocket stages don't fall on people or property
  - Exclusion zone around launch sites help keep away unauthorized civilians and keeps them safe
  - Distance from railways and highways need to be far enough to not damage them, but close enough to transport personnel and material

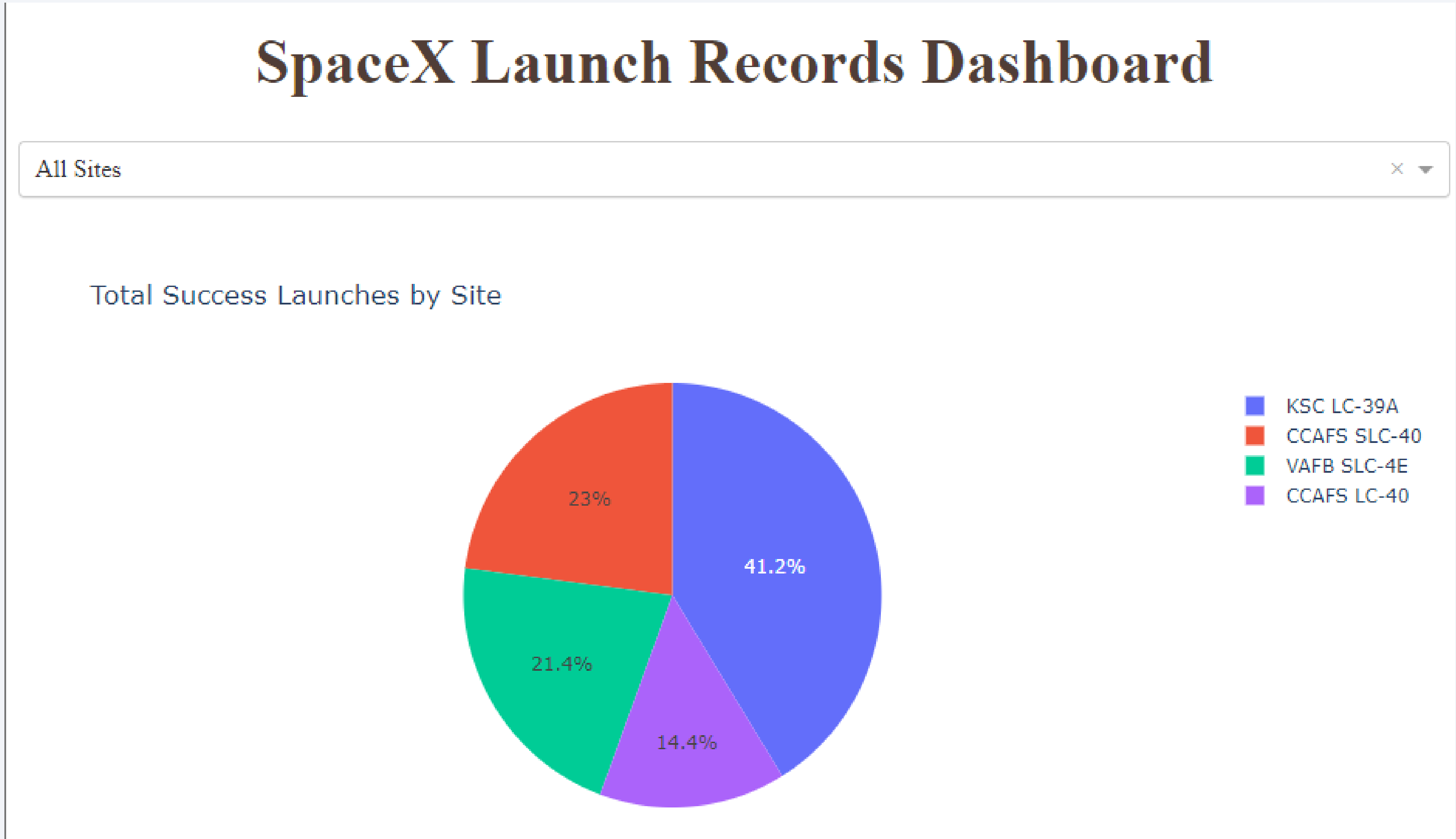


Section 4

Build a Dashboard with Plotly Dash

Launch Success by Launch Site

- KSC LC-39A has the most successful launches among all launch sites

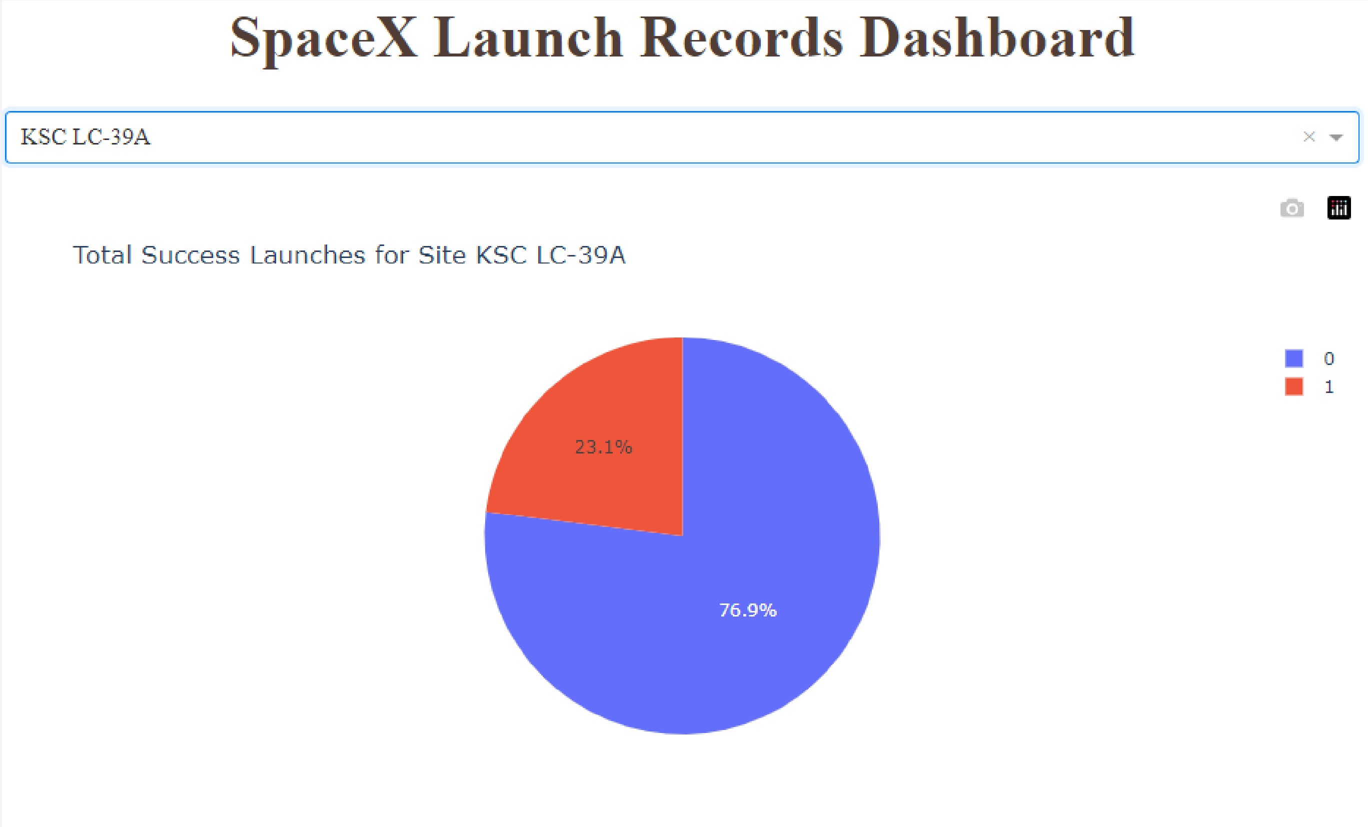


SpaceX Launch Records Dashboard

All Sites

Launch Success of KSC LC-29A

- KSC LC-39A has the highest success rate among all launch sites (76.9%)
- 10 successes | 3 failed launches



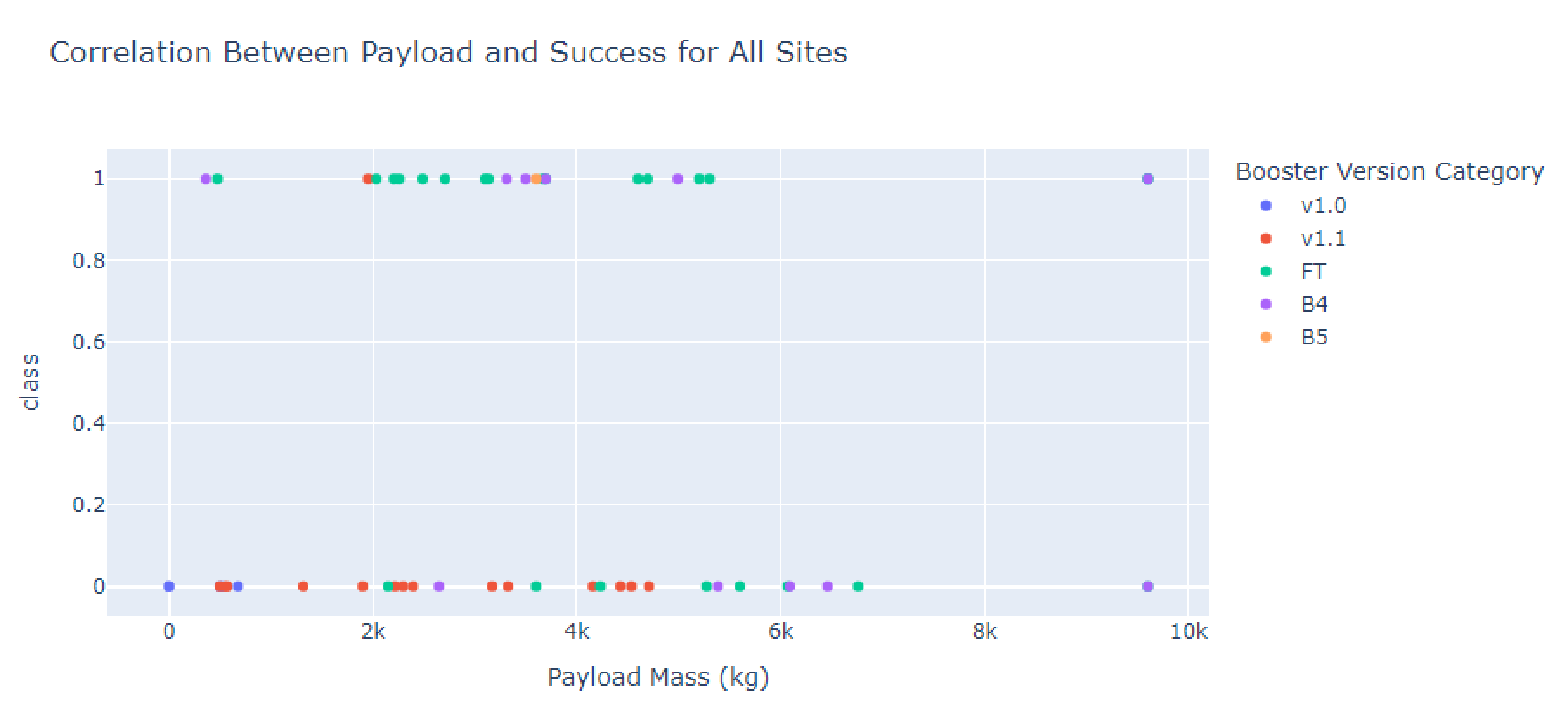
SpaceX Launch Records Dashboard

KSC LC-39A

Total Success Launches for Site KSC LC-39A

Payload mass and Launch Outcome

- Payloads between 2,000kg and 5,000kg have the highest success rate
- Class = 1 represents a success | Class = 0 represents a failure

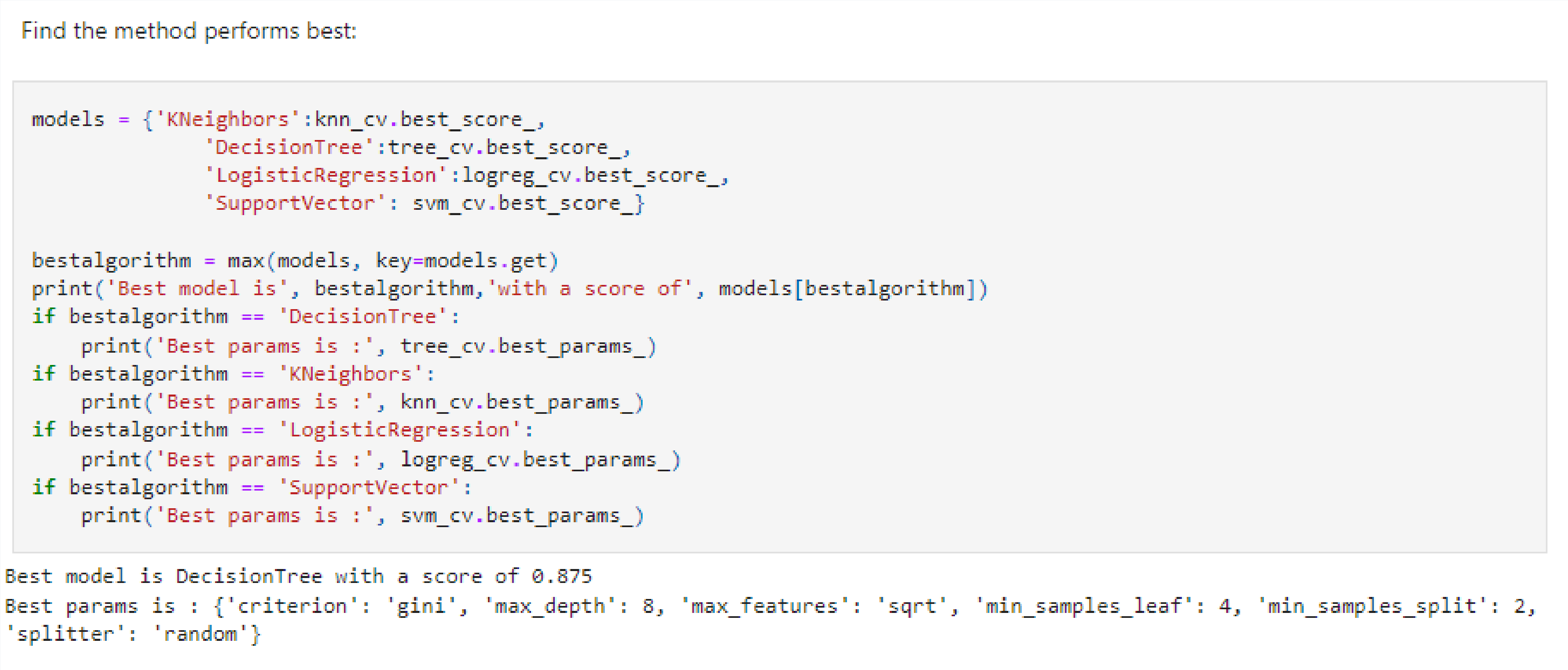


Section 5

Predictive Analysis (Classification)

Classification Accuracy

- All models had about the same scores and accuracy, likely due to the small dataset, but the decision tree model slightly outperformed the rest.



Find the method performs best:

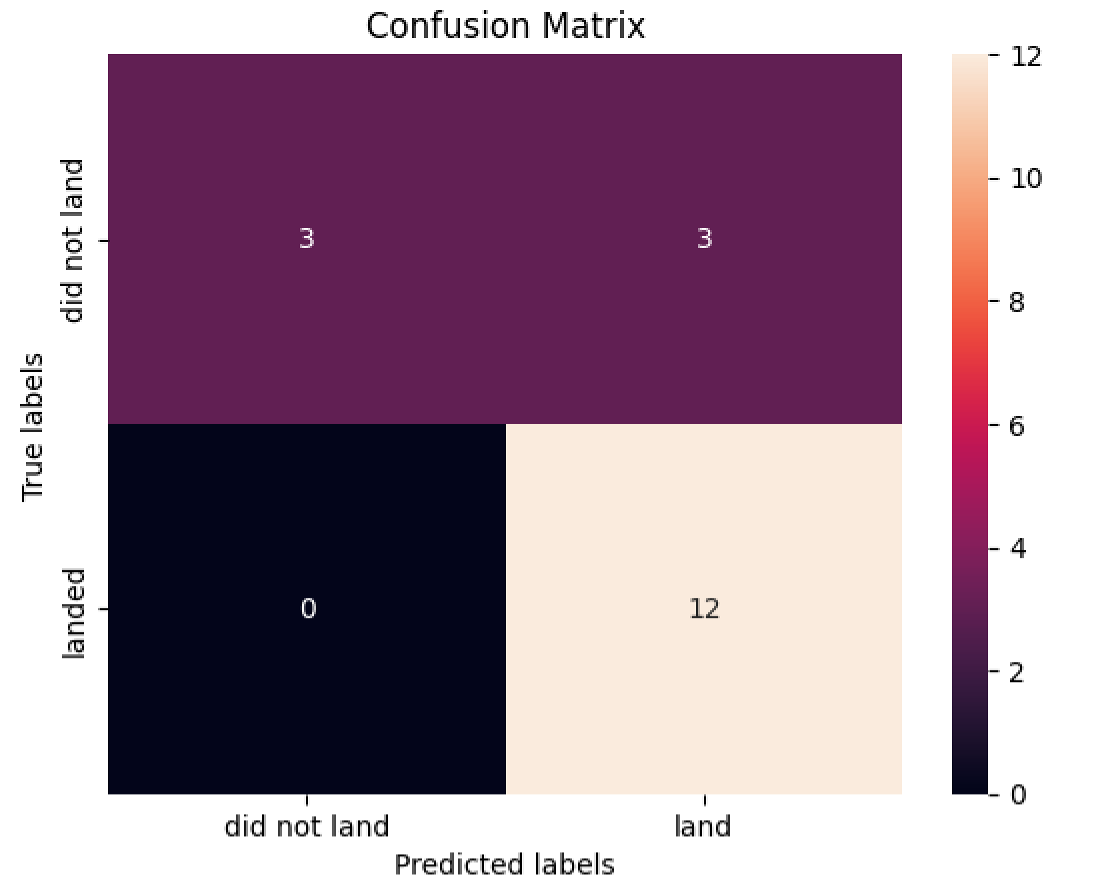
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.875
Best params is : {'criterion': 'gini', 'max_depth': 8, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 2,
'splitter': 'random'}
```

Confusion Matrix

- A confusion matrix summarizes the performance of a classification algorithm
- All confusion matrices for the varying models were identical
- The confusion matrix tells us that there are 12 true positives, 3 true negatives, but 3 false positives and 0 false negatives



- Precision is given by TP / (TP+FP) = **0.8**
- Recall is given by TP / (TP+FN) = **1**
- F1 Score is given by 2 * (Precision * Recall) / (Precision + Recall) = **0.89**
- Accuracy is given by (TP + TN) / (TP + TN + FP + FN) = **0.833**

Conclusions

- Based on our data analysis, we can conclude:
  - Model Performance: The models performed similarly on the test set, with the decision tree model slightly outperforming the rest
  - Site Proximity: Most of the launch sites are positioned near the equator for an additional, natural boost (due to the rotational speed of the earth) which helps save costs. The launch sites are also close to the coasts for the safety of the people
  - Launch success: Increases over time
  - KSC LC-39A: Has the highest success rate among all launch sites, with a 100% success rate for launches less than 5,500kg
  - Orbits: ES-L1, GEO, HEO, and SSO have a 100% success rate
  - Payload mass: Across all launch sites, we observed that the higher the payload mass (kg), the higher the success rate

Thank you!