IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Data Science Project

RAMISETTY SAISRINIVAS
August 24th 2024

GitHub Link:
https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project.git

Outline

- Executive Summary (3)
- Introduction (4)
- Methodology (6)
- Results (16)
- Conclusion (45)
- Appendix (46)

Executive Summary

- **SUMMARY OF METHODOLOGIES**
  - ○ Data collection
  - ○ Data wrangling
  - ○ EDA with data visualization
  - ○ EDA with SQL
  - ○ Building an interactive map with Folium
  - ○ Building a Dashboard with Plotly Dash
  - ○ Predictive analysis (Classification)

- **SUMMARY OF ALL RESULTS**
  - ○ Exploratory data analysis results
  - ○ Interactive analytics demo in screenshots
  - ○ Predictive analysis results

Introduction

❖ PROJECT BACKGROUND AND CONTEXT

- The era of commercial space has arrived, and there are several companies that are making space travel affordable for everyone. Perhaps the most successful of them is SpaceX, and one of the reasons is that their rocket launch is relatively inexpensive.
- 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, we will predict if the Falcon 9 first stage will land successfully. If we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

❖ PROBLEMS YOU WANT TO FIND ANSWERS

- Correlations between each rocket variables and successful landing rate
- Conditions to get the best results and ensure the best successful landing rate

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API & Web Scraping Falcon 9 and Falcon Heavy Launches Records from Wikipedia
- Perform data wrangling
  - Convert outcomes into Training Labels with the booster successfully/unsuccessful landed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Find best Hyperparameter for SVM, Classification Trees and Logistic Regression

Data Collection

- Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.
- The next slide will show the flowchart of data collection from API and the one after will show the flowchart of data collection from web scraping.

❖ SPACE X API DATA COLUMNS:

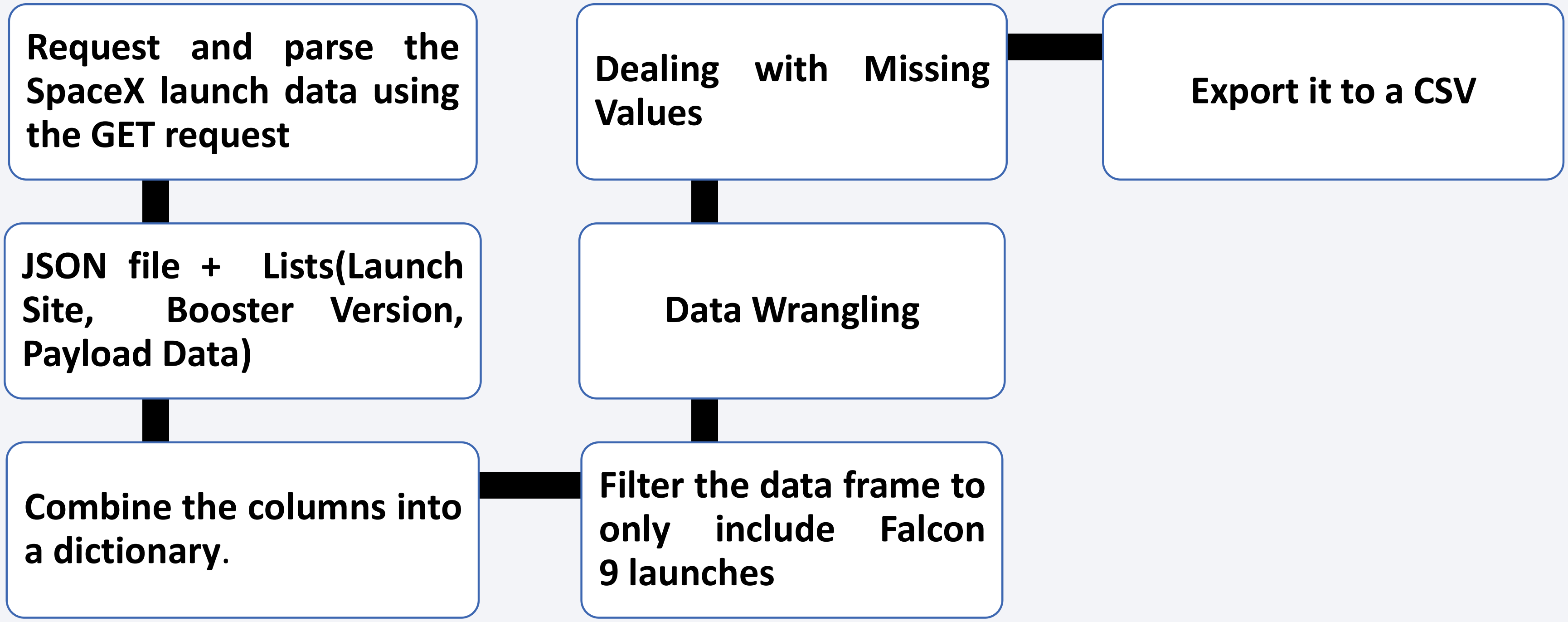
- Flight Number, Date, Booster Version, Payload Mass, Orbit, Launch Site, Outcome, Flights, Grid Fins,
- Reused, Legs, Landing Pad, Block, Reused Count, Serial, Longitude, Latitude

❖ WIKIPEDIA WEBSCRAPE DATA COLUMNS:

- Flight Number, Launch site, Payload, Payload Mass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts



Request and parse the SpaceX launch data using the GET request

JSON file + Lists(Launch Site, Booster Version, Payload Data)

Combine the columns into a dictionary.

Filter the data frame to only include Falcon 9 launches

Data Wrangling

Dealing with Missing Values

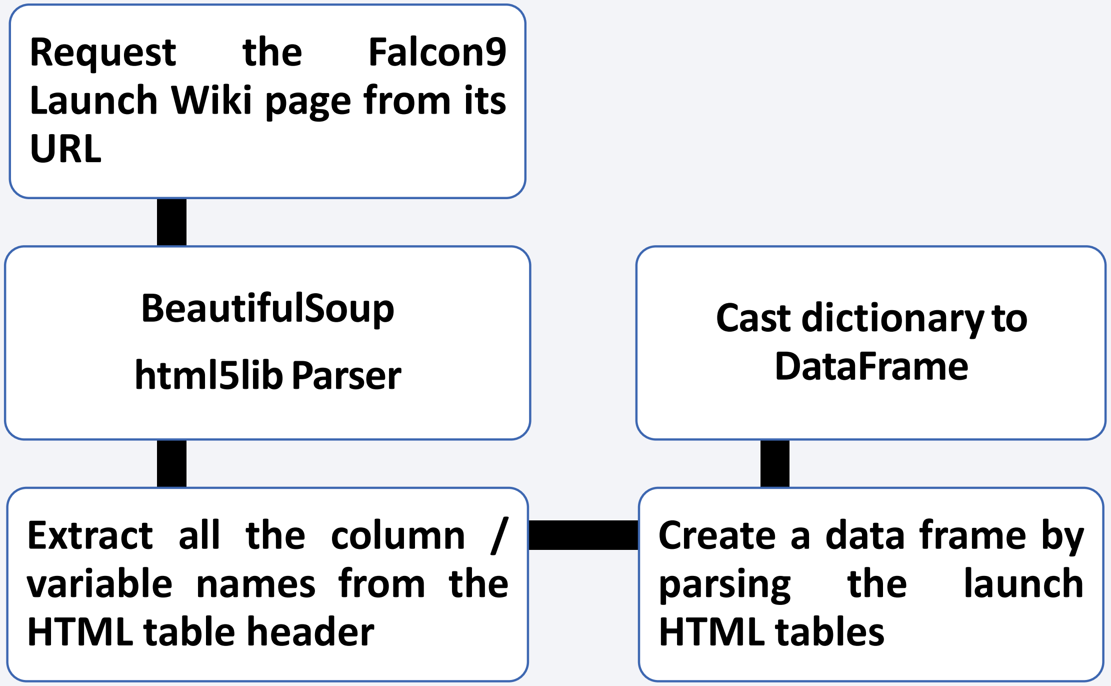
Export it to a CSV

- GitHub URL:

https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c808470480038d0af7709/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts



Request the Falcon9 Launch Wiki page from its URL

BeautifulSoup

html5lib Parser

Extract all the column / variable names from the HTML table header

Create a data frame by parsing the launch HTML tables

Cast dictionary to DataFrame

- GitHub URL:

  https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/jupyter-labs-webscraping.ipynb

Data Wrangling

- There are several cases in which the booster failed to successfully land on the dataset, and sometimes it attempted to land but failed because of accident.
  - True Ocean: the mission result has successfully landed in a specific area of the ocean
  - False Ocean: the mission result has not successfully landed in a specific area of the ocean
  - True RTLS: the mission result successfully landed on the ground pad
  - False RTLS: the mission result has not successfully landed on the ground pad
  - True ASDS: the mission result has successfully landed on the drone ship
  - False ASDS: the mission result has not landed on the drone ship
- Converting these results into training labels:
  - 1 = successful / 0 = failure

- GitHub URL:

  https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

❖ SCATTER CHART:

- ○ Flight Number vs. Launch Site
- ○ Payload vs. Launch Site
- ○ Flight Number vs. Orbit Type
- ○ Payload vs. Orbit Type
- ○ A scatter plot shows how much one variable is affected by another. The relationship between two variables is called a correlation. This plot is generally composed of large data bodies.

❖ BAR CHART:

- ○ Orbit Type vs. Success Rate
- ○ A Bar chart makes it easy to compare datasets between multiple groups at a glance. X axis represents a value and the Y represents a discrete value. The purpose of this chart is to indicate the relationship between the axes.

❖ LINE CHART:

- ○ Year vs. Success Rate
- ○ A Line chart shows data variables and trends very clearly and helps predict the results of data that has not yet been recorded.

- GitHub URL:

https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- ❖ Loading the dataset into the corresponding table in a Db2 database, and executing SQL queries to answer following questions:
  - ○ Displaying the names of the unique launch sites in the space mission
  - ○ Displaying 5 records where launch sites begin with the string ‘KSC'
  - ○ Displaying the 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 drone ship was achieved
  - ○ Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
  - ○ Listing the total number of successful and failure mission outcomes
  - ○ Listing the names of the booster_versions which have carried the maximum payload mass
  - ○ Listing the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2017
  - ○ Ranking 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

- GitHub URL:

https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/jupyter-labs-eda-sql-edx_sqllite.ipynb

Build an Interactive Map with Folium

❖ Objects created and added to a folium map:

- ○ Markers that show all launch sites on a map
- ○ Markers that show the success/failed launches for each site on the map
- ○ Lines that show the distances between a launch site to its proximities

❖ By adding these objects, following geographical patterns about launch sites are found:

- ○ Are launch sites in close proximity to railways? Yes
- ○ Are launch sites in close proximity to highways? Yes
- ○ Are launch sites in close proximity to coastline? Yes
- ○ Do launch sites keep certain distance away from cities? Yes

- GitHub URL:

https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

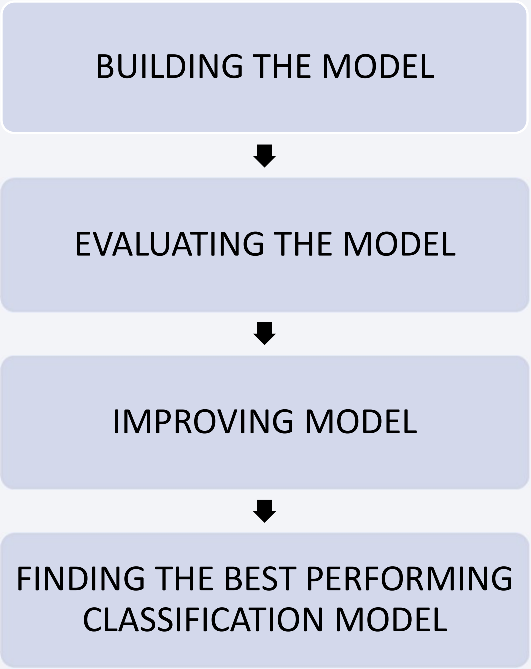
- The dashboard application contains a pie chart and a scatter point chart
  - Pie chart
    - For showing total success launches by sites
    - This chart can be selected to indicate a successful landing distribution across all launch sites or to indicate the success rate of individual launch sites.
  - Scatter chart
    - For showing the relationship between Outcomes and Payload mass(Kg) by different boosters
    - Has 2 inputs: All sites/individual site & Payload mass on a slider between 0 and 10000 kg
    - This chart helps determine how success depends on the launch point, payload mass, and booster version categories.

- GitHub URL: https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/23b1b5c488d324594b8d853374d1c502aca80e1f/spacex_dash_app.py

Predictive Analysis (Classification)

- ❖ Perform exploratory Data Analysis & determine Training Labels
  - ○ Create a column for the class
  - ○ Standardize the data
  - ○ Split into training data and test data

- ❖ Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
  - ○ Find the method performs best using test data


BUILDING THE MODEL

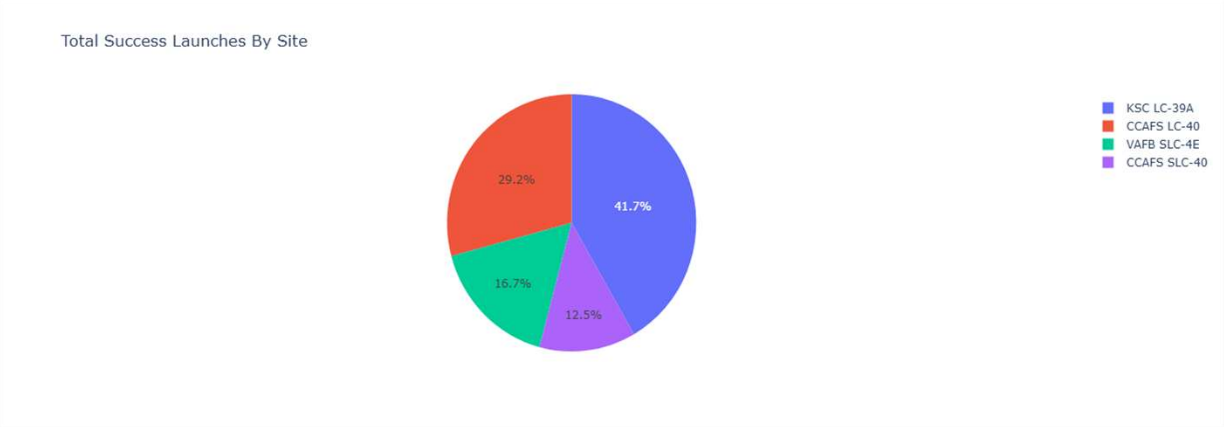
EVALUATING THE MODEL

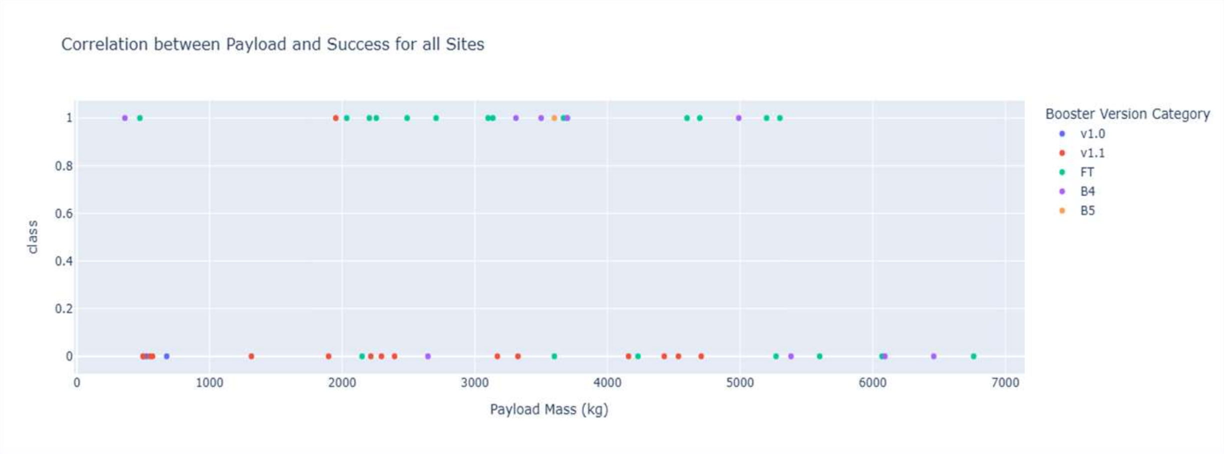
IMPROVING MODEL

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

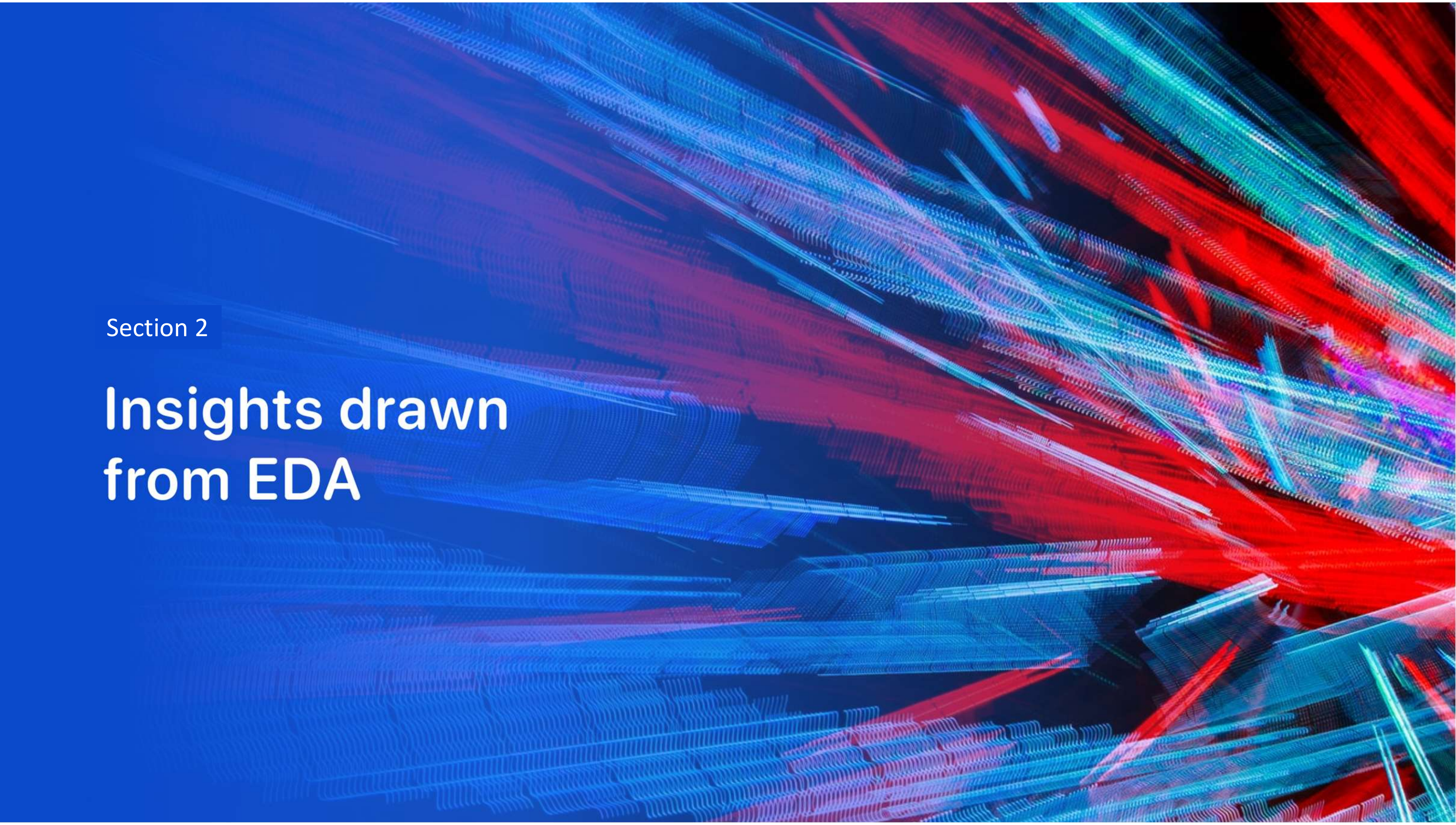
GitHub URL:
https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project/blob/a8eb412107e908a737f5c80847048038d0af7709/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results





- The left screenshot is a preview of the Dashboard with Plotly Dash.
- The results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and Interactive Dashboard will be shown in the next slides.
- In the test set, the accuracy of two models was virtually the same at 77.78% and accuracy of other two models was same at 72.22%

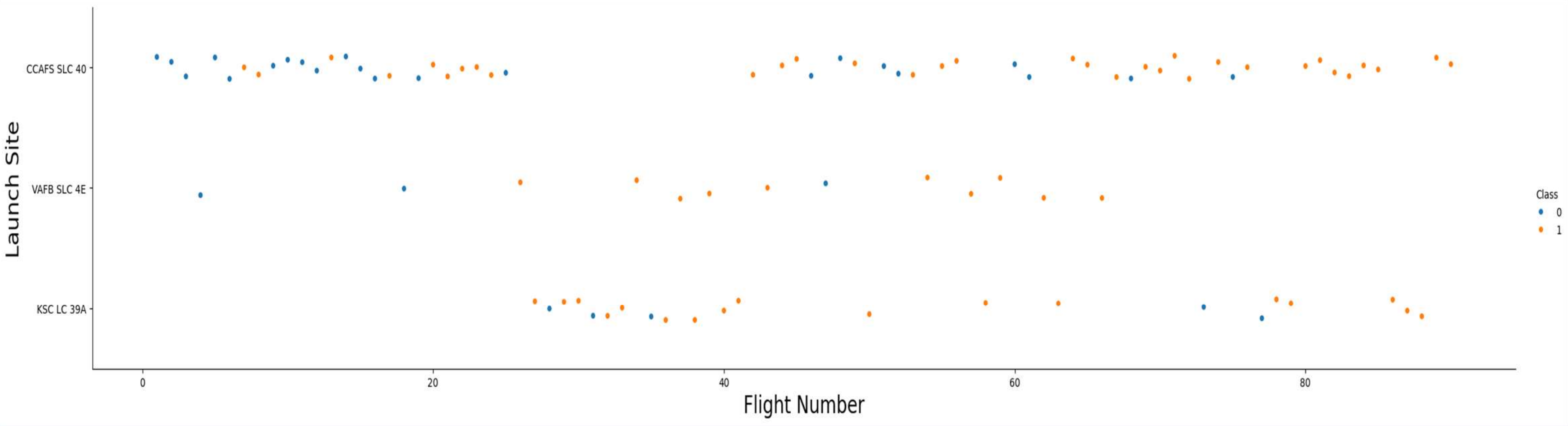


Section 2

Insights drawn from EDA

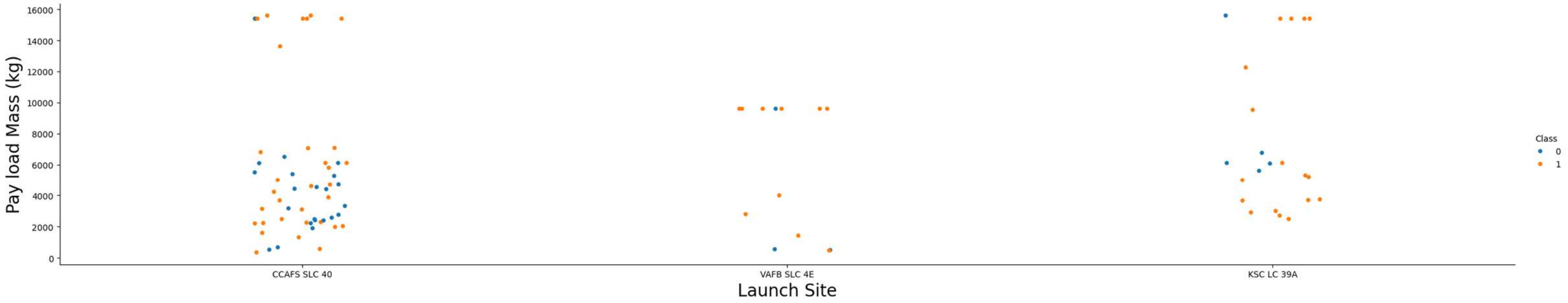
Flight Number vs. Launch Site

- Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.
- This figure shows that the success rate increased as the number of flights increased.



Payload vs. Launch Site

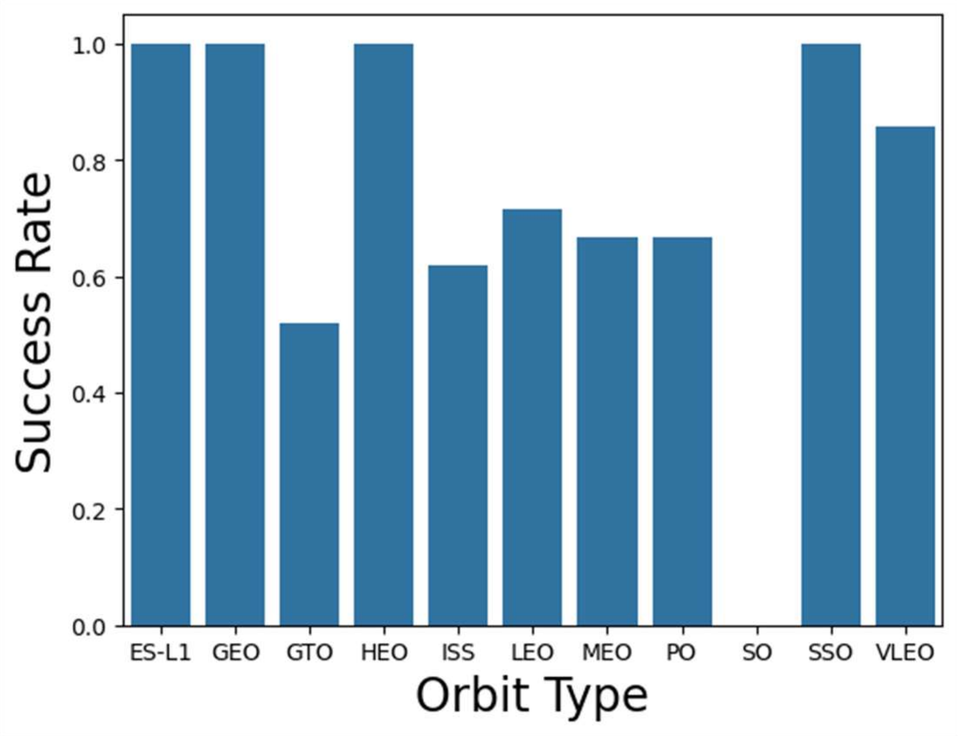
- Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.
- At first glance, the larger pay load mass, the higher the rocket's success rate, but it seems difficult to make decisions based on this figure because no clear pattern can be found between successful launch and Pay Load Mass.



Success Rate vs. Orbit Type

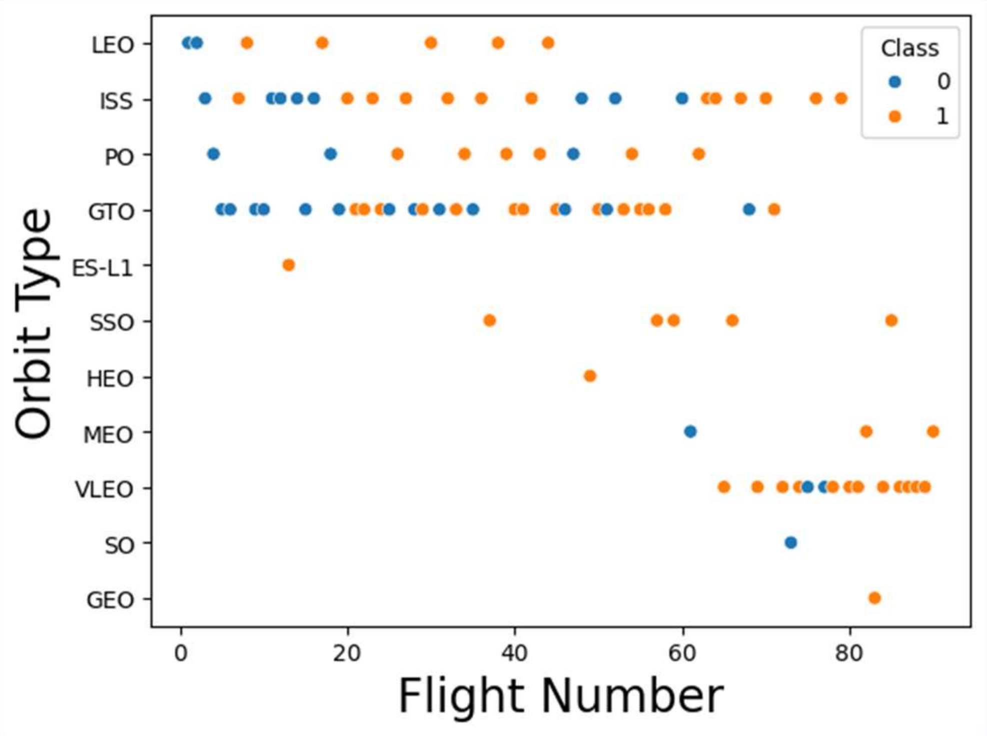
- The success rate of orbit type GTO is only 50%, and it is the lowest except for type SO, which recorded failure in a single attempt.

- On the other hand, Orbit types ES-L1, GEO, HEO, SSO, and have the highest success rates (100%).



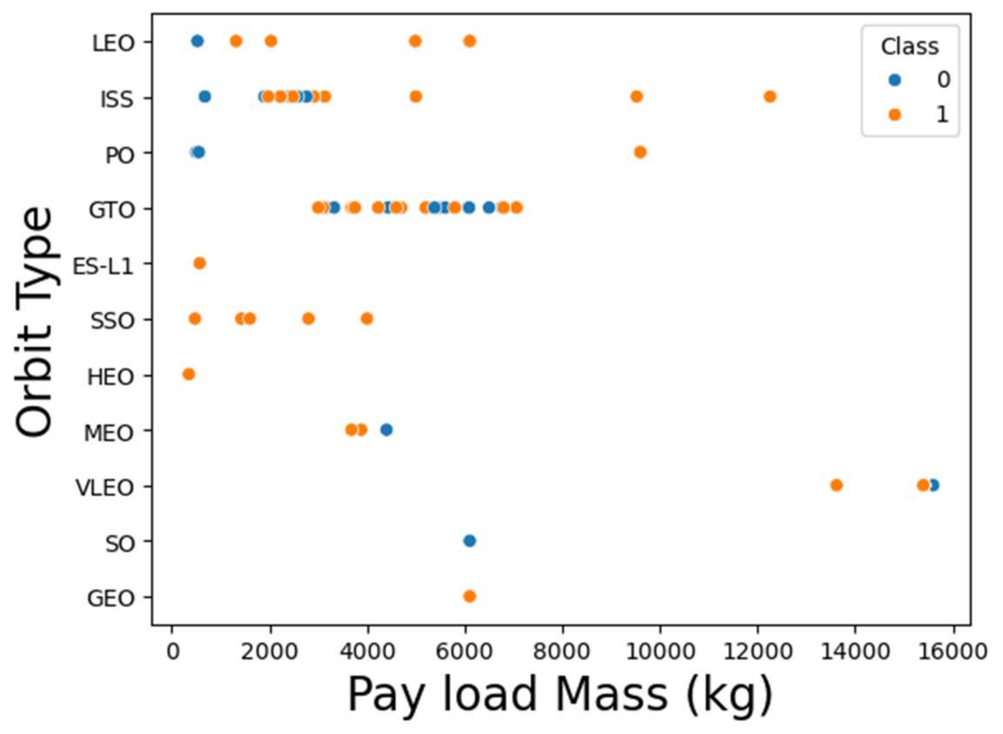
Flight Number vs. Orbit Type

- Class 0 (blue) represents unsuccessful launch, and Class 1(orange) represents successful launch.
- SpaceX starts with LEO with a moderate success rate, and it seems that VLEO, which has a high success rate, is used the most in recent launches.
- In most cases, the launch outcome seems to be correlated with the flight number.



Payload vs. Orbit Type

- Class 0 (blue) represents unsuccessful launch, and Class 1(orange) represents successful launch.
- With heavy payloads the successful landing or positive landing rate are more for LEO and ISS.
- But, in the case of GTO, it is hard to distinguish between the positive landing rate and the negative landing rate.

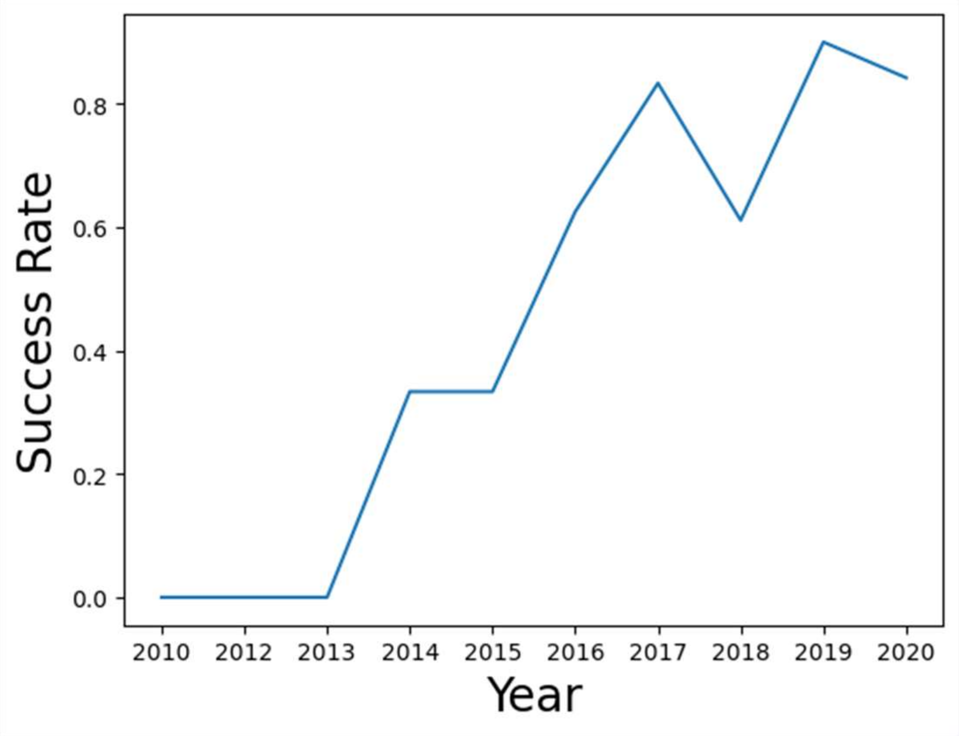


Launch Success Yearly Trend

- Since 2013, the success rate has continued to increase until 2017.

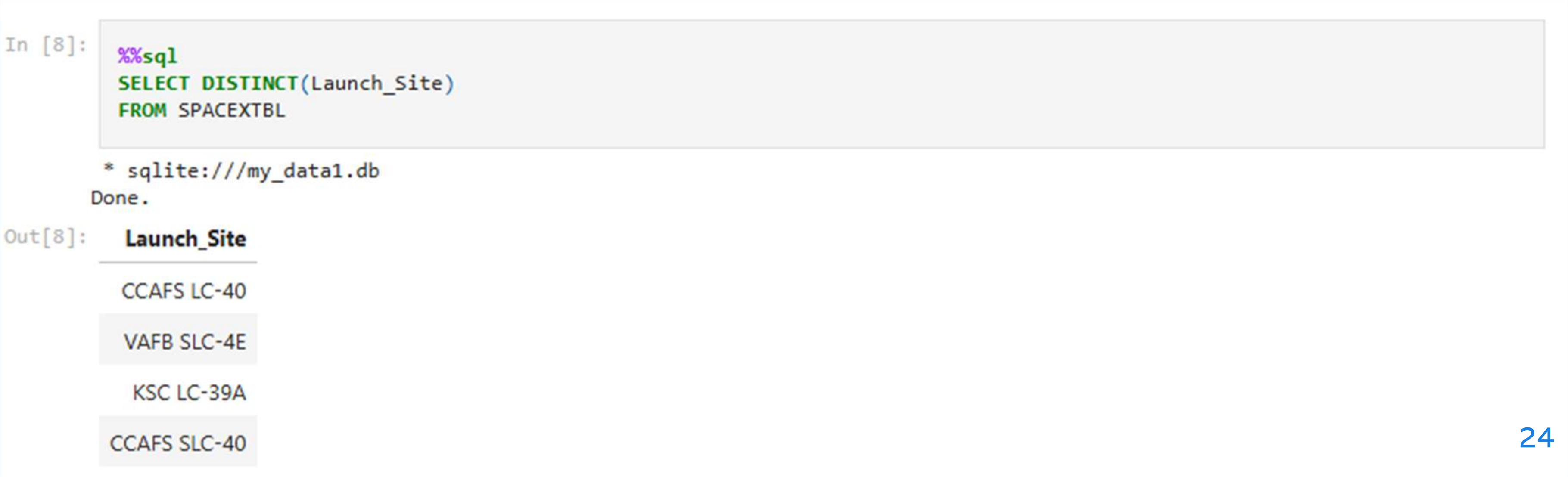
- The rate decreased slightly in 2018

- Recently, it has shown a success rate of about 80%.



All Launch Site Names

- When the SQL DISTINCT clause is used in the query, only unique values are displayed in the Launch_Site column from the SpaceX table.
- There are four unique launch sites: CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40



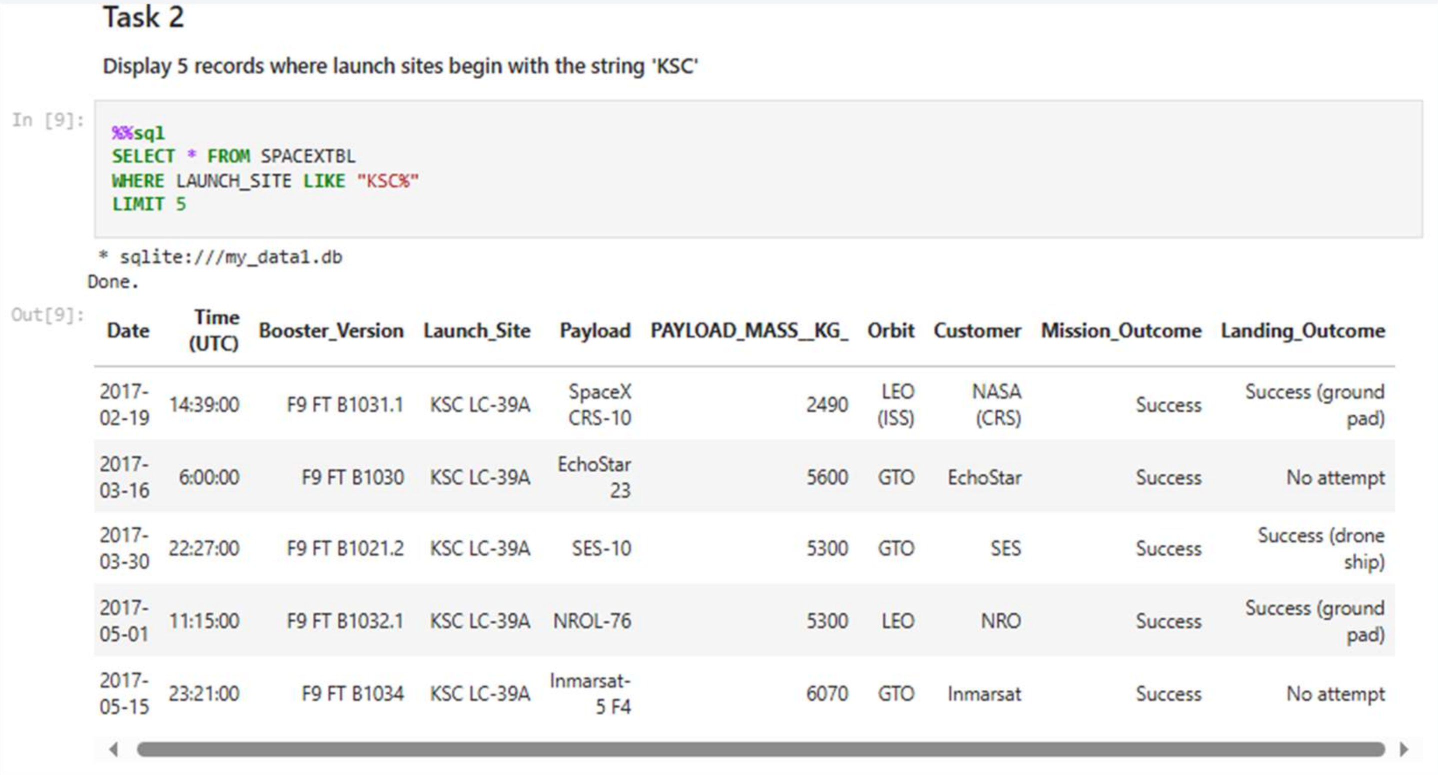
```
In [8]: %%sql
        SELECT DISTINCT(Launch_Site)
        FROM SPACEXTBL

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

Out[8]:

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

Launch Site Names Begin with 'KSC'

- Only five records of the SpaceX table were displayed using LIMIT 5 clause in the query.
- Using the LIKE operator and the percent sign (%) together, the Launch_Site name starting with KSC could be called.



Task 2

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

In [9]:

```
%%sql
SELECT * FROM SPACEXTBL
WHERE LAUNCH_SITE LIKE "KSC%"
LIMIT 5
```

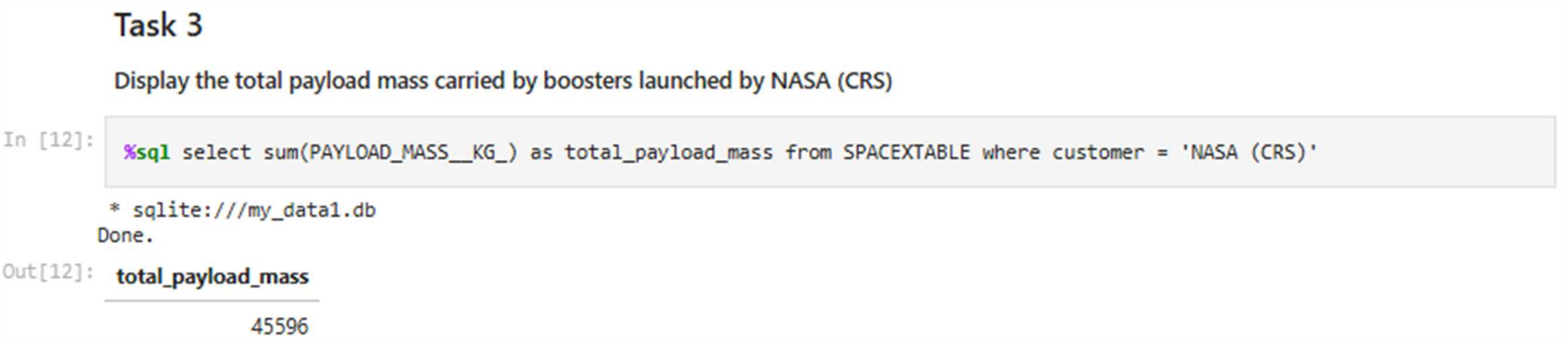
```
 * sqlite:///my_data1.db
Done.
```

Out[9]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

Total Payload Mass

- Using the SUM() function to calculate the sum of column PAYLOAD_MASS__KG_.
- In the WHERE clause, filter the dataset to perform calculations only if Customer is NASA (CRS).



Task 3

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

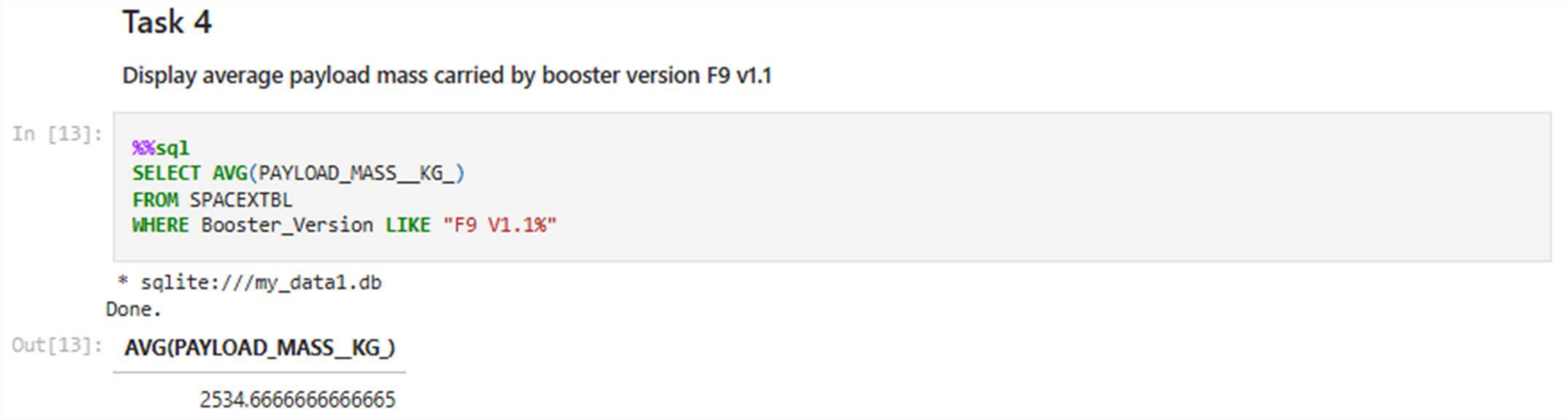
```
In [12]: %sql select sum(PAYLOAD_MASS__KG_) as total_payload_mass from SPACEXTABLE where customer = 'NASA (CRS)'
```

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

```
Out[12]: total_payload_mass
                      45596
```

Average Payload Mass by F9 v1.1

- Using the AVG() function to calculate the average value of column PAYLOAD_MASS__KG_.
- In the WHERE clause, filter the dataset to perform calculations only if Booster_version is F9 v1.1.



Task 4

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

In [13]:
```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE Booster_Version LIKE "F9 V1.1%"
```

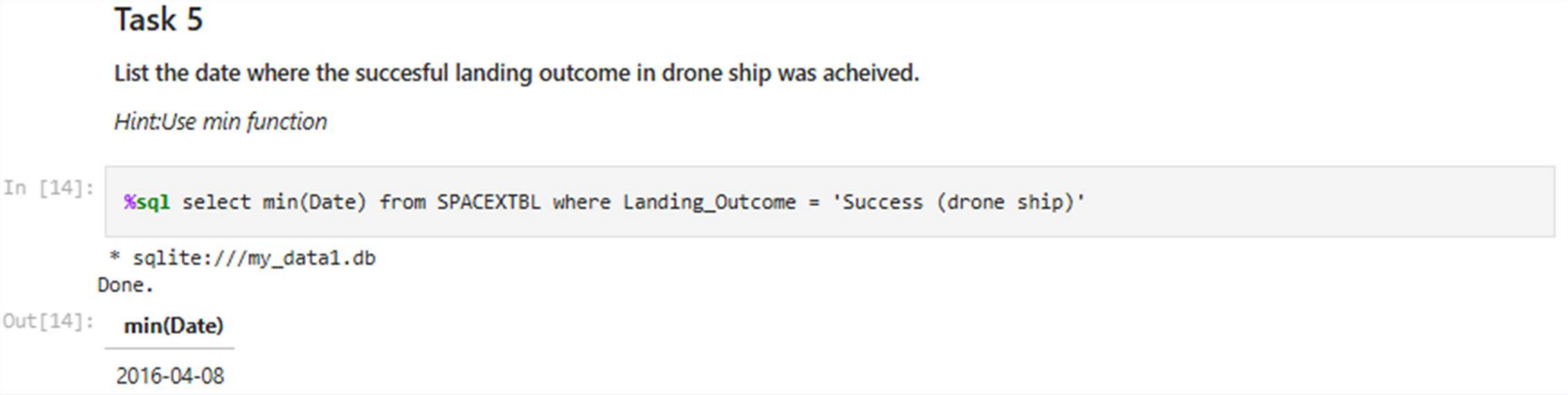
```
 * sqlite:///my_data1.db
Done.
```

Out[13]:

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Drone Ship Landing Date

- Using the MIN() function to find out the earliest date in the column DATE.
- In the WHERE clause, filter the dataset to perform a search only if Landing__outcome is Success (drone ship).



Task 5

List the date where the succesful landing outcome in drone ship was acheived.

Hint:Use min function

```
In [14]:  %sql select min(Date) from SPACEXTBL where Landing_Outcome = 'Success (drone ship)'
```

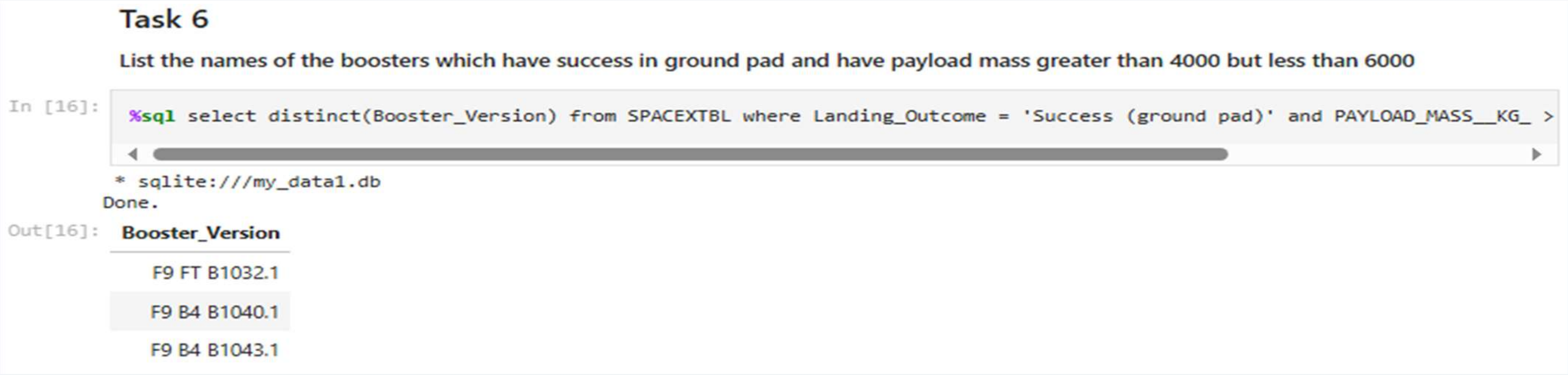
```
 * sqlite:///my_data1.db
Done.
```

Out[14]:

min(Date)
2016-04-08

Successful Ground pad Landing with Payload between 4000 and 6000

- In the WHERE clause, filter the dataset to perform a search if Landing__outcome is Success (ground pad)
- Using the AND operator to display a record if additional condition PAYLOAD_MASS__KG_ is between 4000 and 6000.



Task 6

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

In [16]:
```
%sql select distinct(Booster_Version) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)' and PAYLOAD_MASS__KG_ >
```

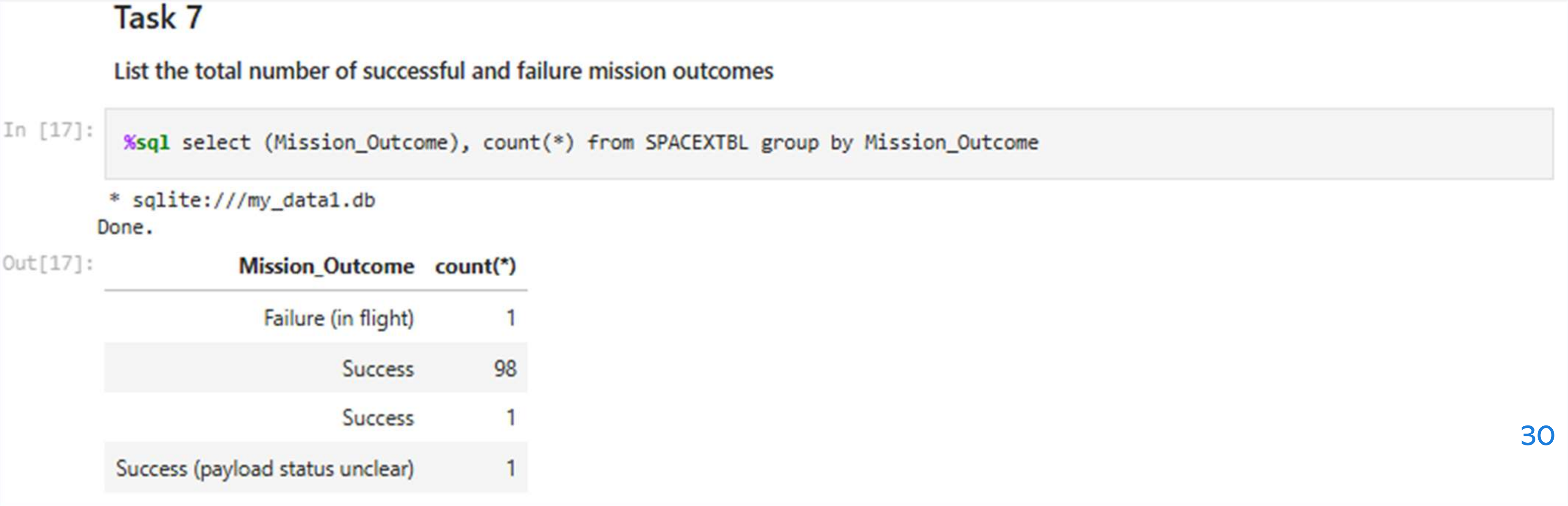
```
 * sqlite:///my_data1.db
Done.
```

Out[16]:

Booster_Version
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

Total Number of Successful and Failure Mission Outcomes

- Using the COUNT() function to calculate the total number of columns.
- Using the GROUP BY statement, groups rows that have the same values into summary rows to find the total number in each Mission_outcome.
- SpaceX have successfully completed nearly 99% of its missions



Task 7

List the total number of successful and failure mission outcomes

```
In [17]: %sql select (Mission_Outcome), count(*) from SPACEXTBL group by Mission_Outcome
```

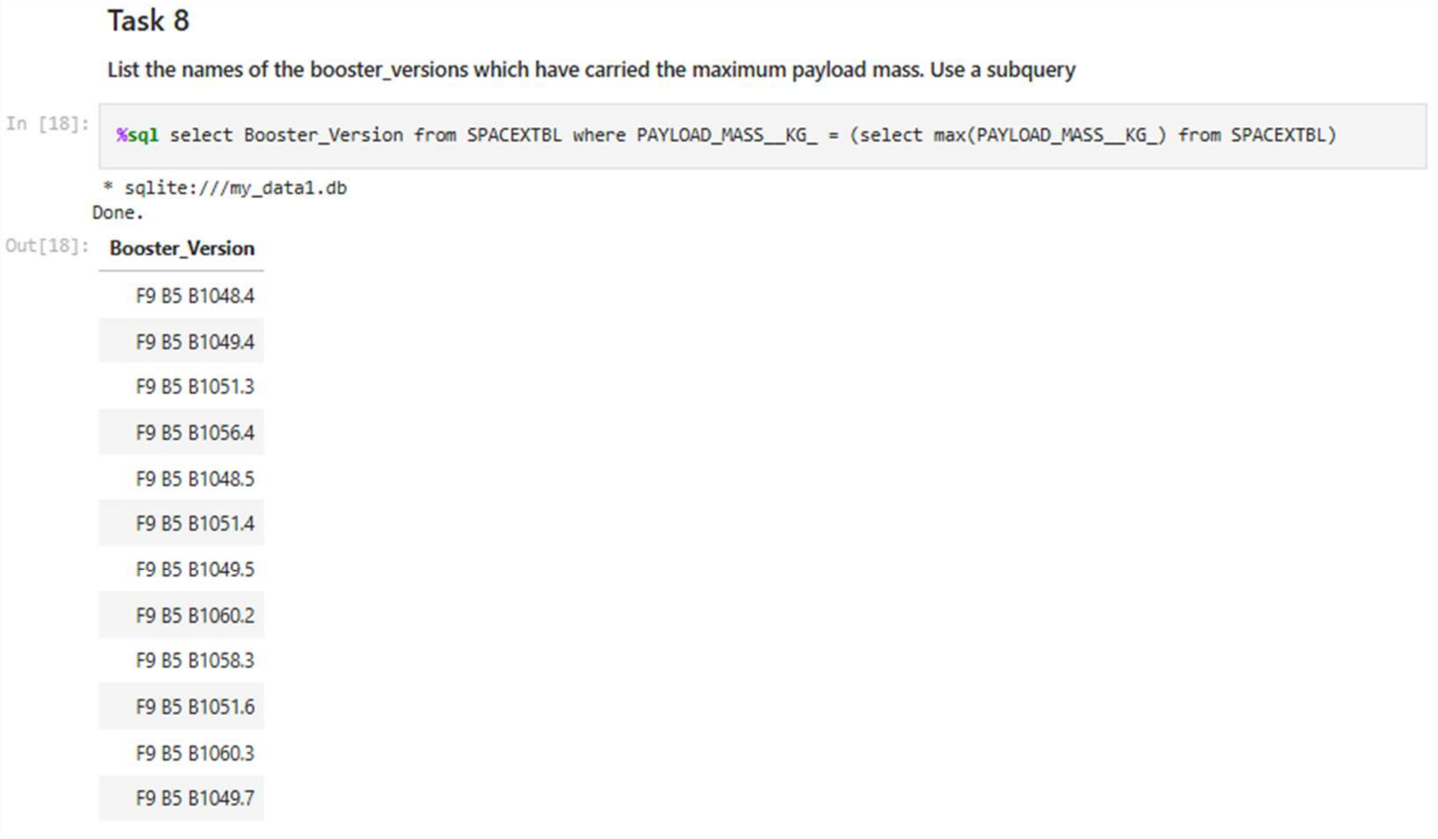
```
 * sqlite:///my_data1.db
Done.
```

Out[17]:

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

Boosters Carried Maximum Payload

- According to the result, version F9 B5 B10xx.x boosters could carried the maximum payload.



Task 8

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

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

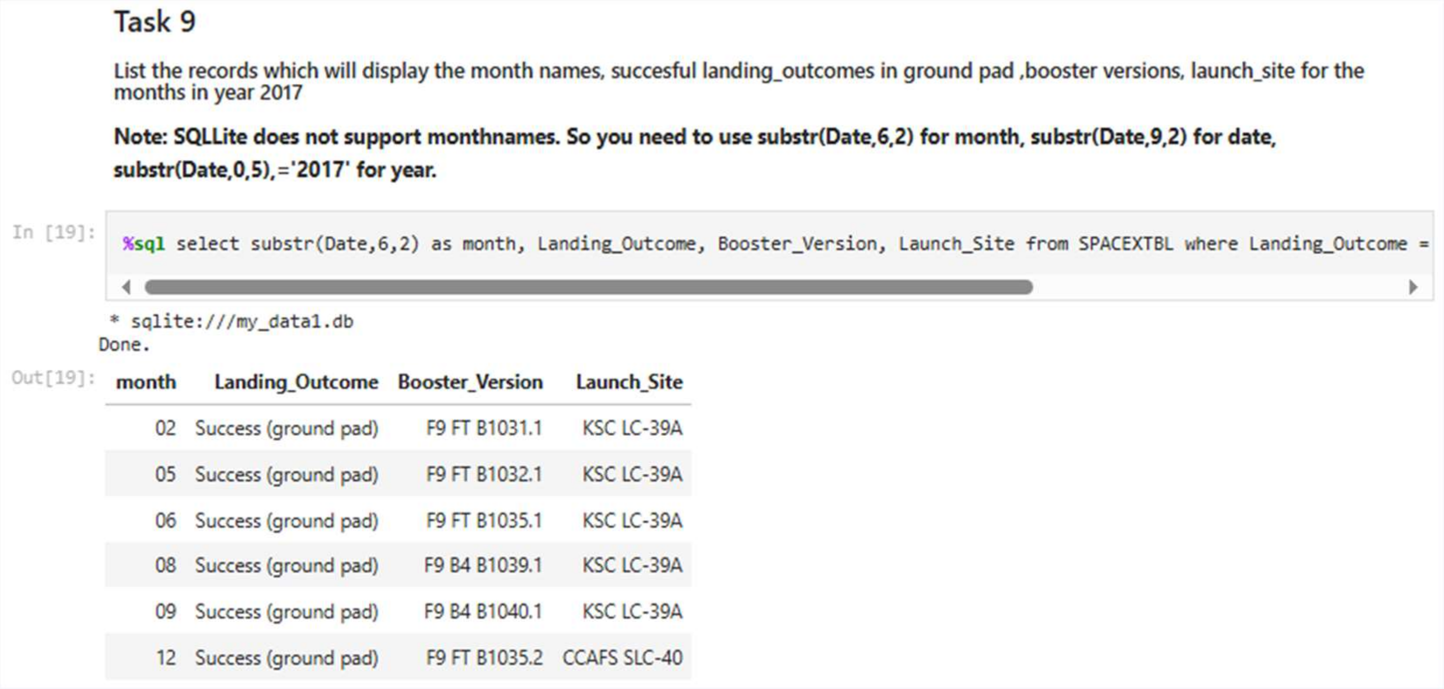
```
 * sqlite:///my_data1.db
Done.
```

Out[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
F9 B5 B1049.7

2017 Launch Records

- In the WHERE clause, filter the dataset to perform a search if Landing__outcome is success (ground pad).
- Using the AND operator to display a record if additional condition YEAR is 2017.



Task 9

List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017

Note: SQLLite does not support monthnames. So you need to use substr(Date,6,2) for month, substr(Date,9,2) for date, substr(Date,0,5),='2017' for year.

In [19]:

```
%sql select substr(Date,6,2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTBL where Landing_Outcome =
```

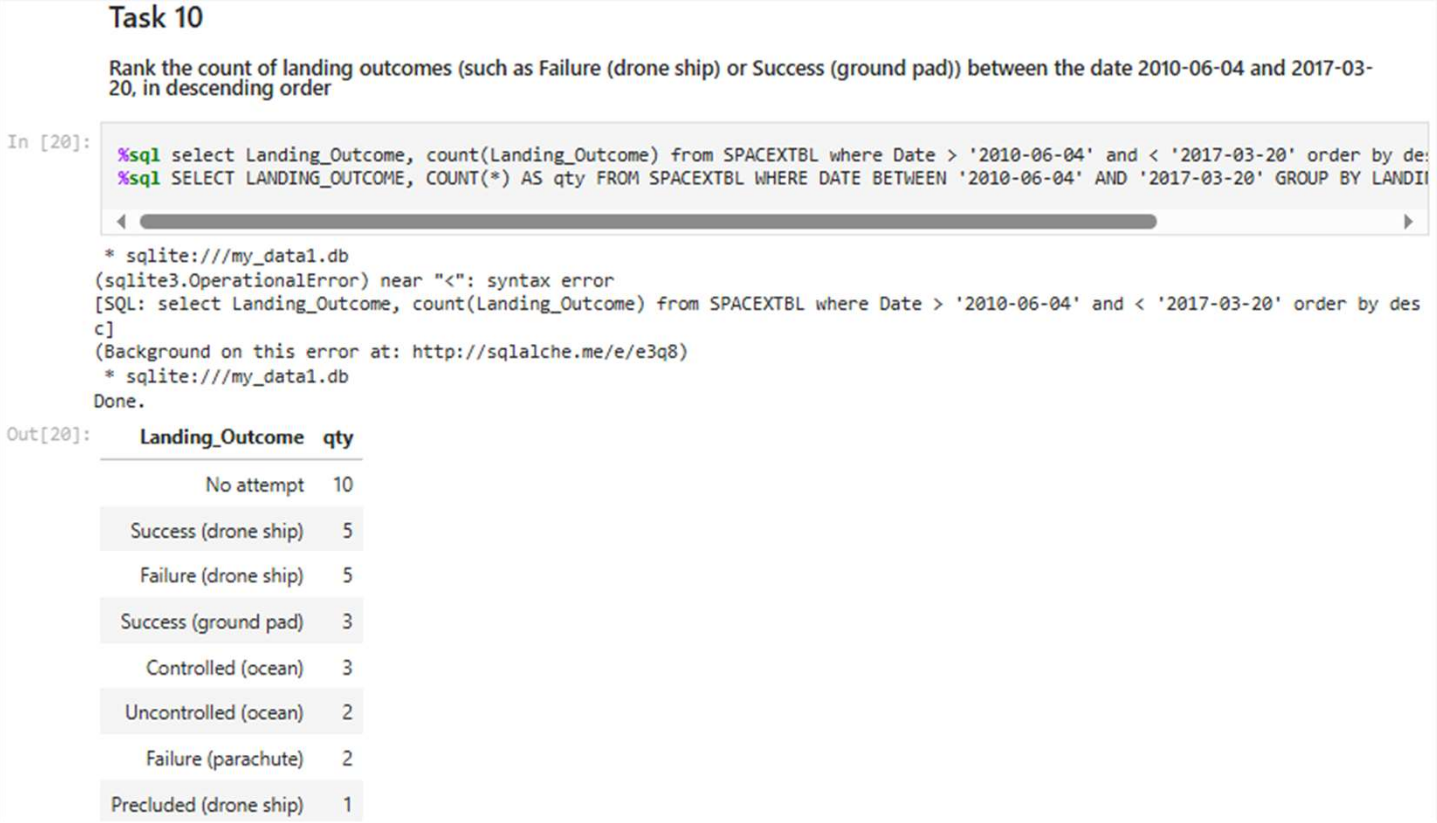
```
 * sqlite:///my_data1.db
Done.
```

Out[19]:

month	Landing_Outcome	Booster_Version	Launch_Site
02	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
05	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
06	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
08	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
09	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
12	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

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

- Below is 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.



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

In [20]:

```
%sql select Landing_Outcome, count(Landing_Outcome) from SPACEXTBL where Date > '2010-06-04' and < '2017-03-20' order by de
%sql SELECT LANDING_OUTCOME, COUNT(*) AS qty FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDI
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) near "<": syntax error
[SQL: select Landing_Outcome, count(Landing_Outcome) from SPACEXTBL where Date > '2010-06-04' and < '2017-03-20' order by des
c]
(Background on this error at: http://sqlalche.me/e/e3q8)
 * sqlite:///my_data1.db
Done.
```

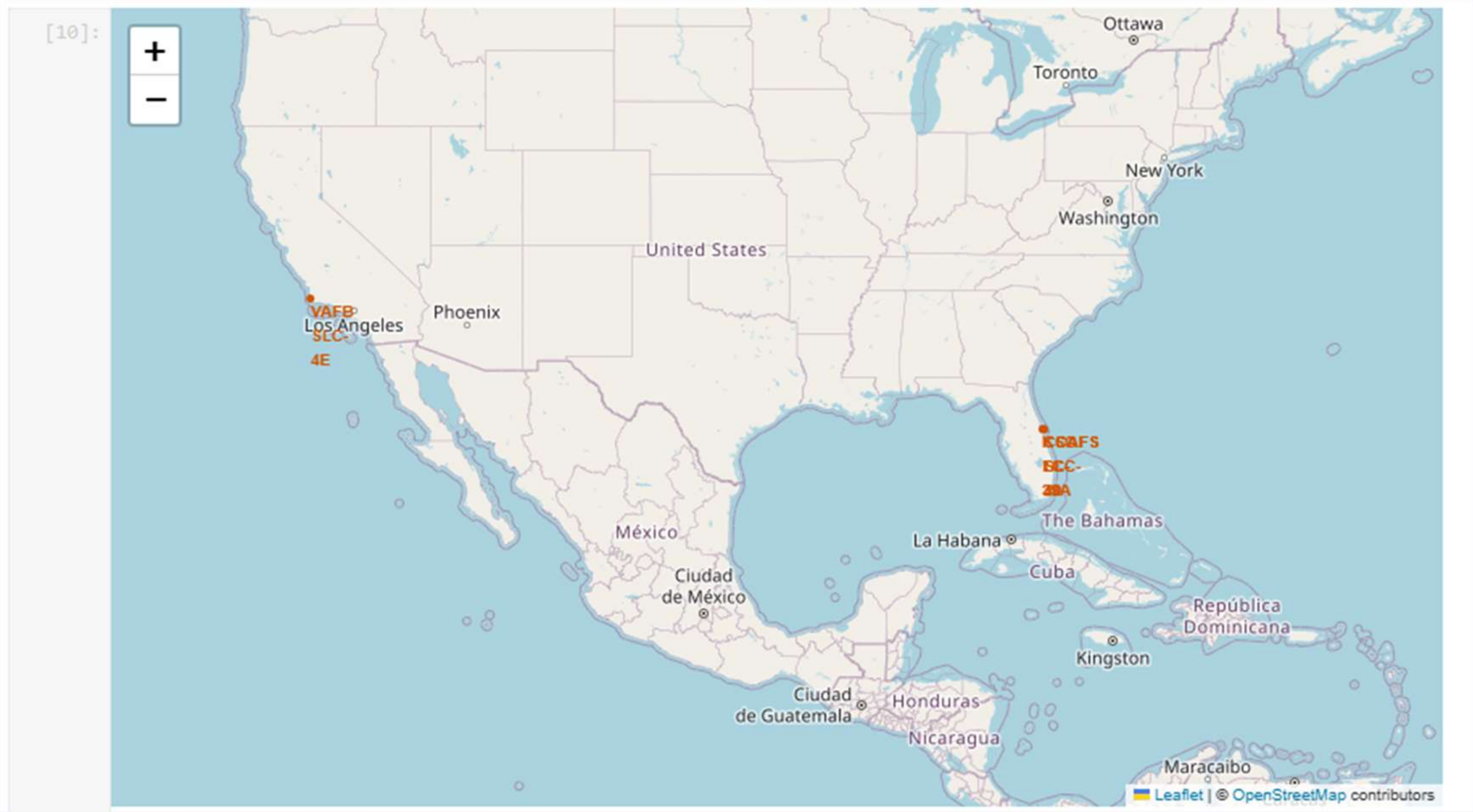
Out[20]:

Landing_Outcome	qty
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

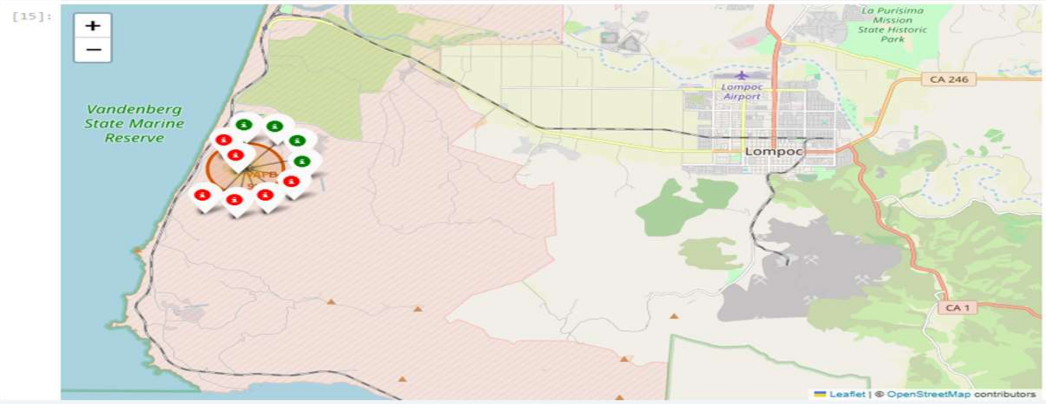
All Launch Sites Locations

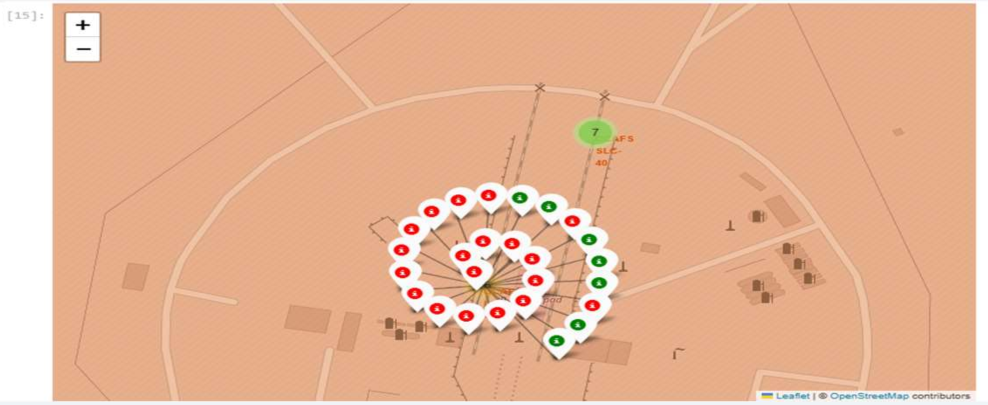


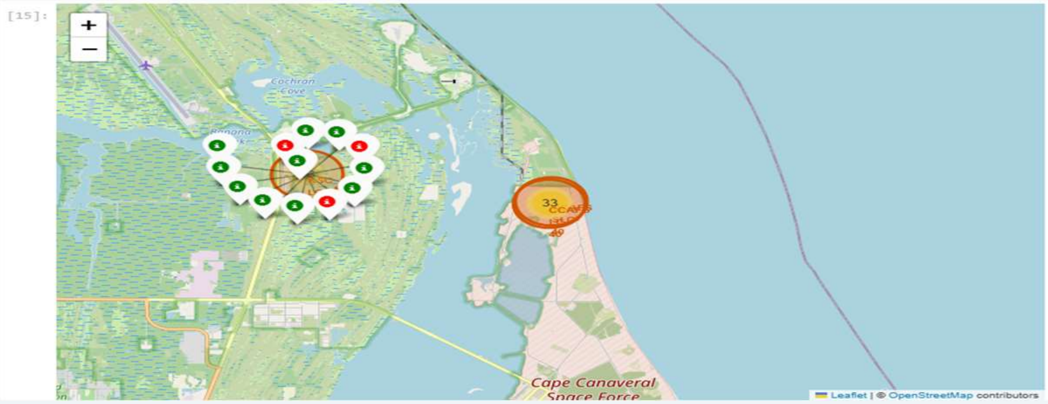
- The map shows all SpaceX launch sites, and the map also shows that all launch sites are in the United States.

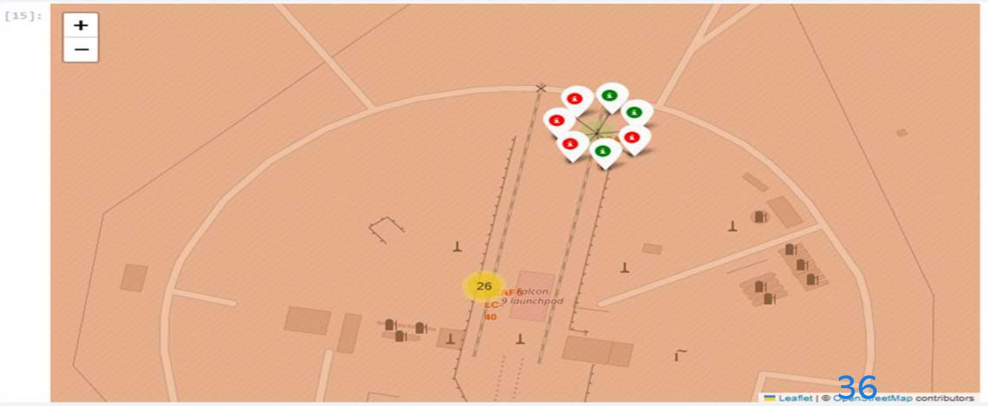
- As can be seen from the map, all launch sites are near the coast.

Color-labeled Launch Outcomes

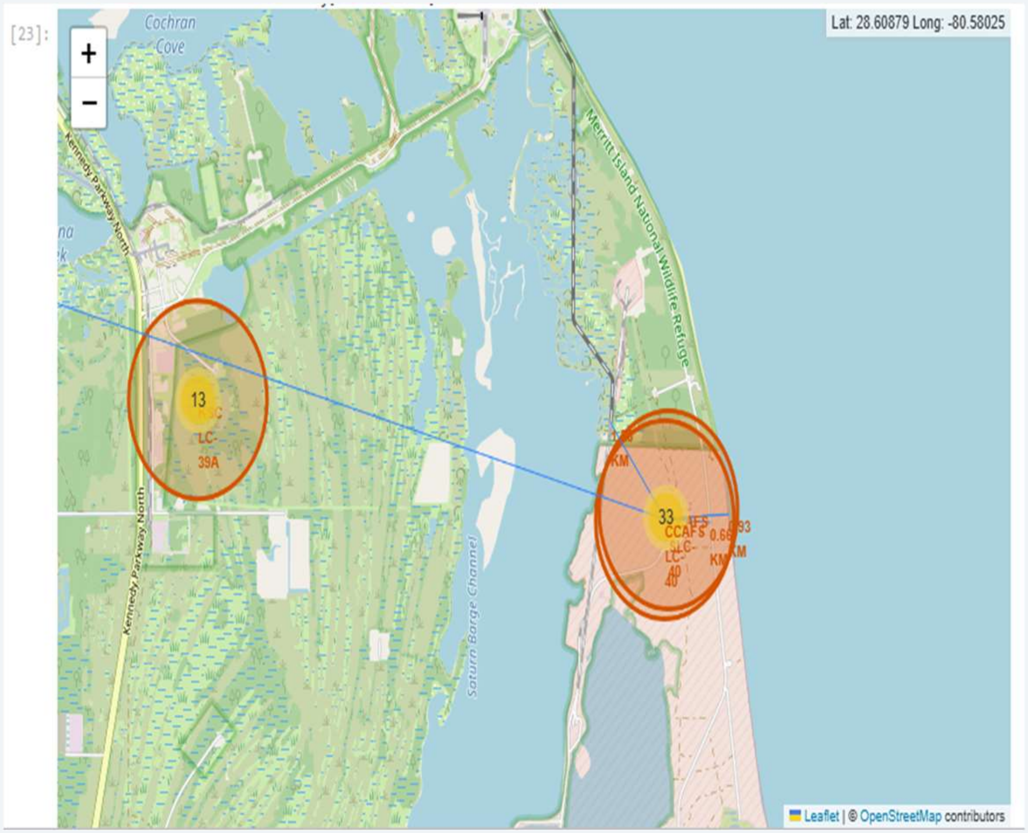

[15]:
+
−
Vandenberg State Marine Reserve
La Purisima Mission State Historic Park
Lompoc Airport
Lompoc
CA 246
CA 1
VAFB
Leaflet | © OpenStreetMap contributors

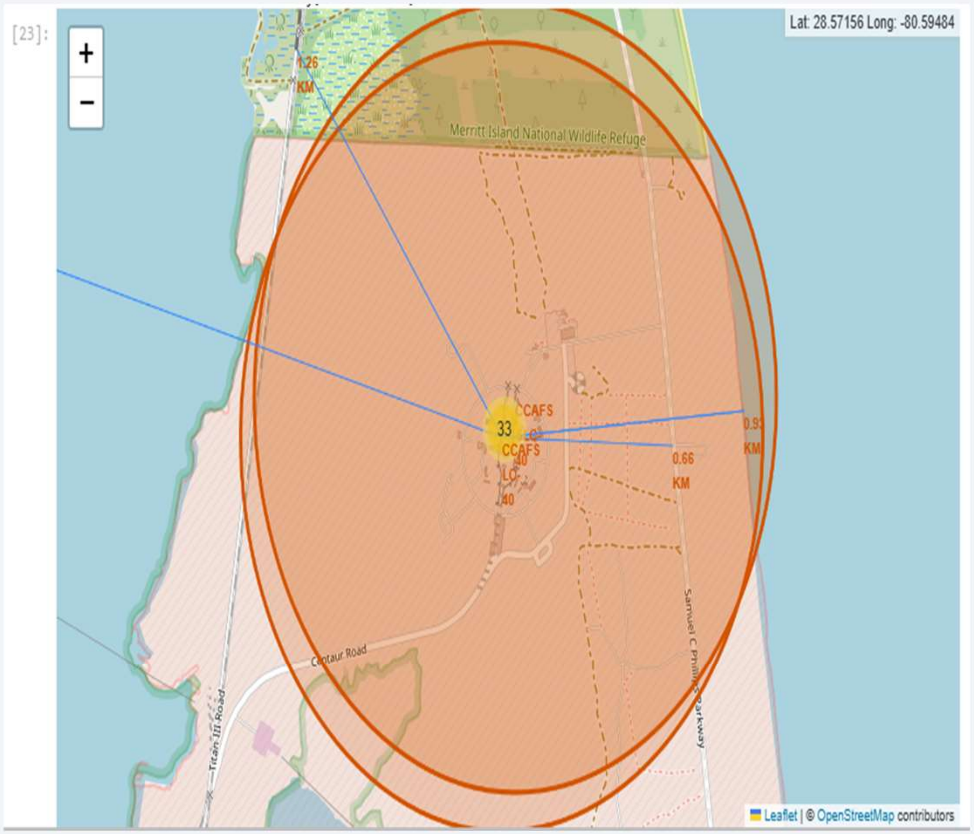


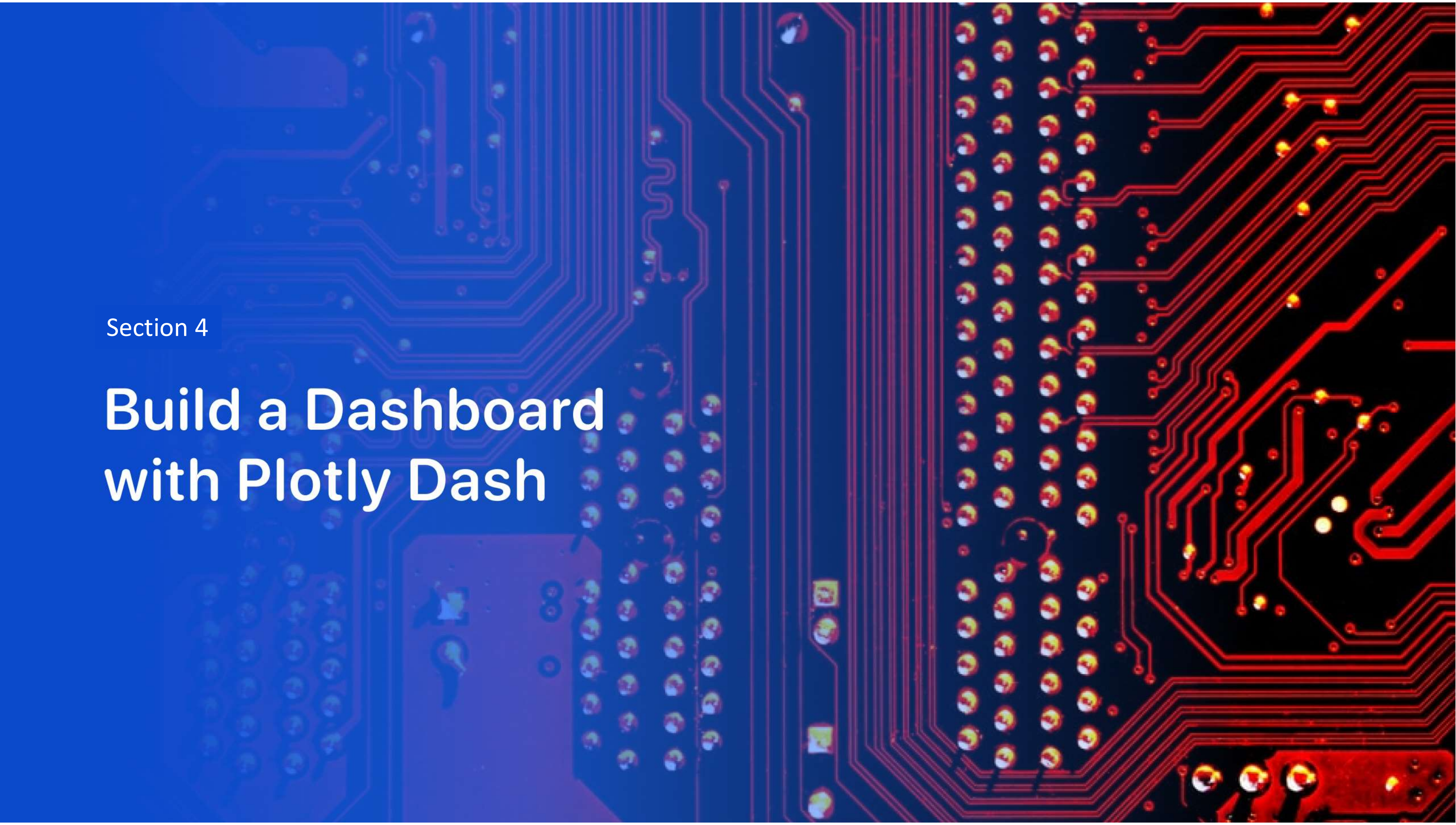

[15]:
+
−
Cochran Cove
33
CCAFS
SLC-
40
Cape Canaveral Space Force
Leaflet | © OpenStreetMap contributors



Proximities of Launch Sites


[23]:
+
−
Lat: 28.60879 Long: -80.58025
Cochran Cove
Kennedy Parkway North
Merritt Island National Wildlife Refuge
Saturn Barge Channel
13
LC-
39A
33
CCAFS
LC-
40
0.66
KM
0.93
KM
Leaflet | © OpenStreetMap contributors

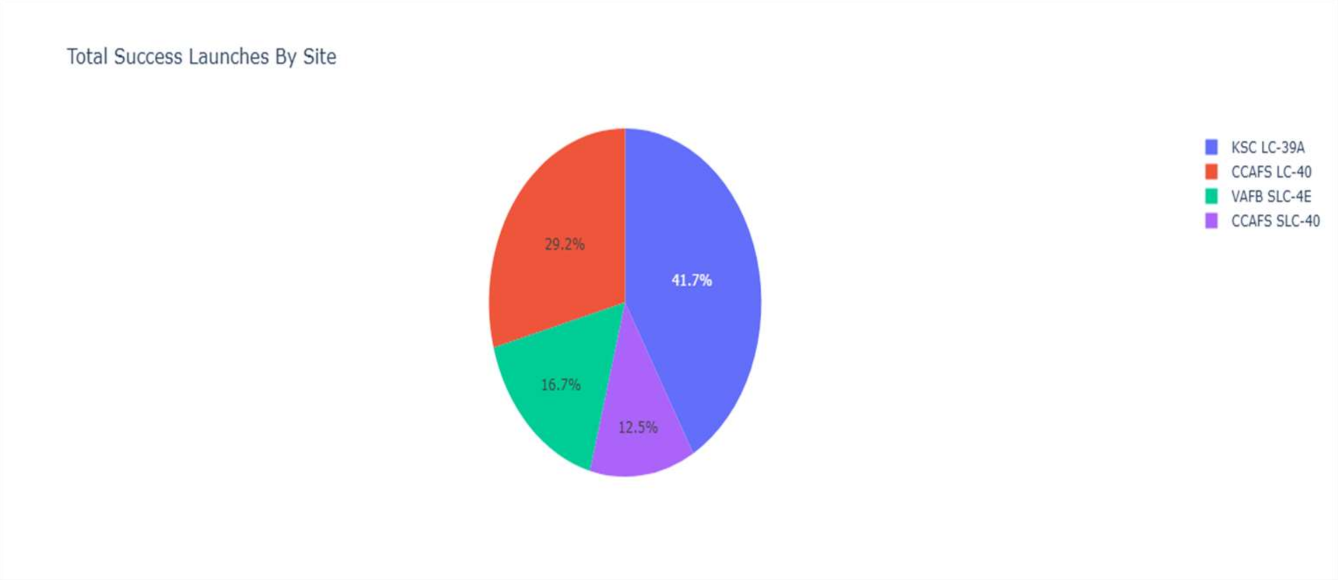

[23]:
+
−
Lat: 28.57156 Long: -80.59484
1.26
KM
Merritt Island National Wildlife Refuge
33
CCAFS
LC-
40
0.66
KM
0.93
KM
Centaur Road
Titan III Road
Samuel C Phillips Parkway
Leaflet | © OpenStreetMap contributors



Section 4

Build a Dashboard with Plotly Dash

Total Successful Launches by all sites

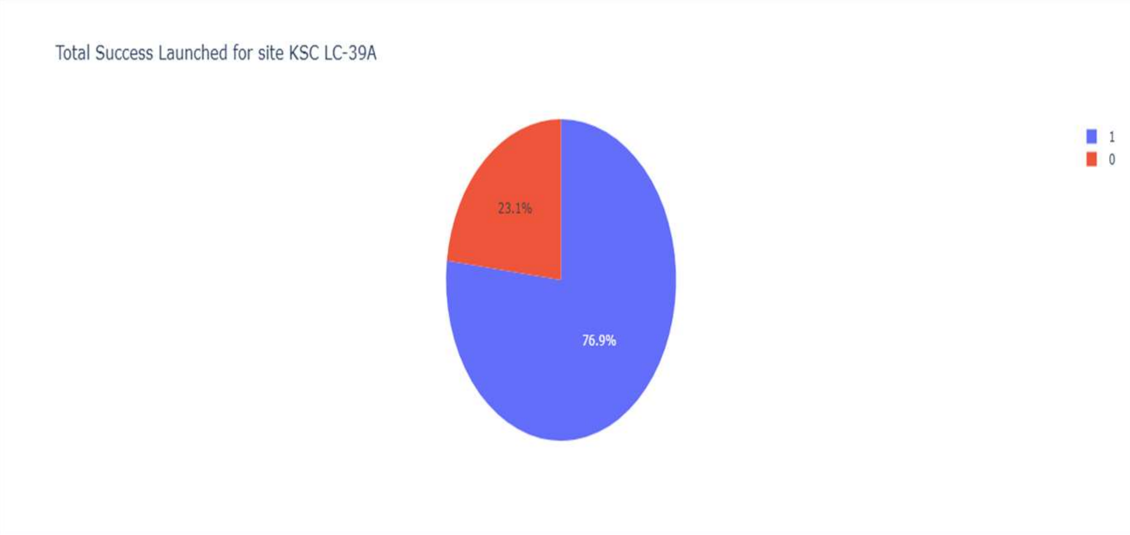


- In the left dashboard screenshot we can see the success rate of various launch sites.
- Among them KSLC-39A records the most launch success among all sites.
- CCAFS SLC- 40 records the least success comparing to others

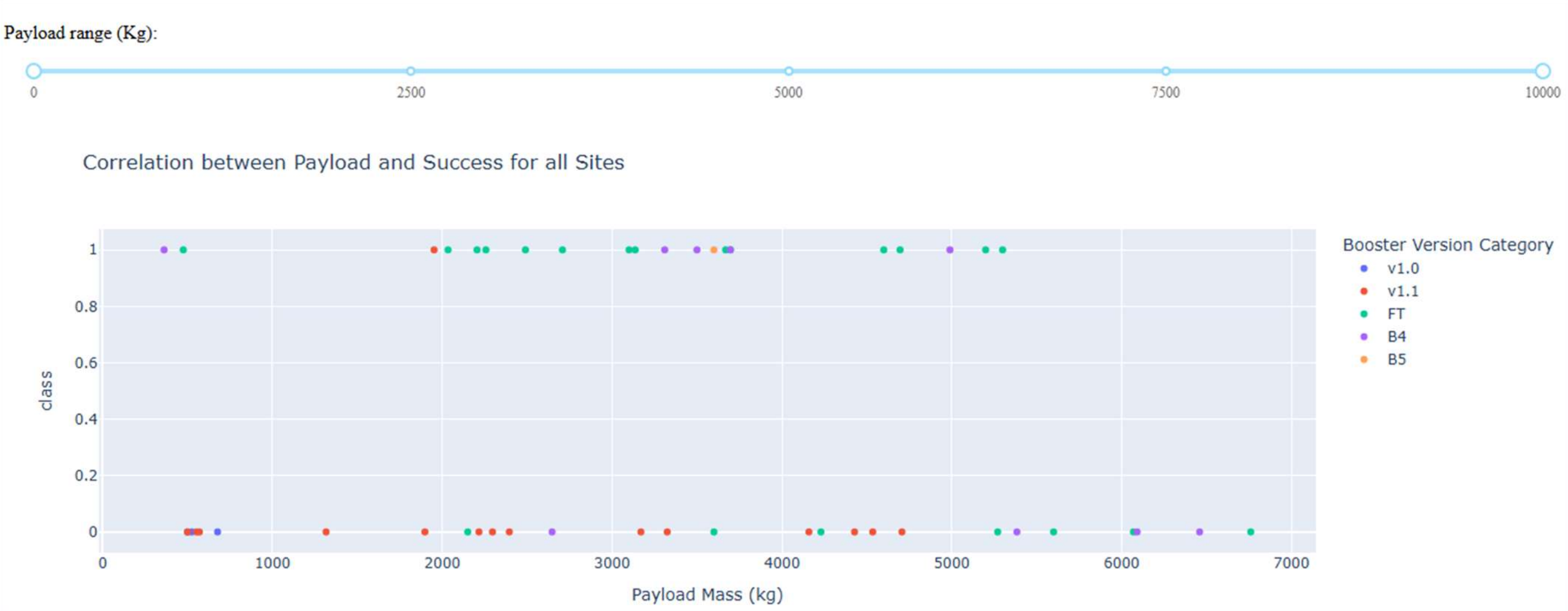
Launch Site with Highest Launch Success Ratio

- KSLC-39A records the most launch success among all sites.

- KSLC-39A has the highest success rate with 10 landing successes (76.9%) and 3 landing failures (23.1%)



Payload vs. Launch Outcome Scatter Plot for all sites



- The above image is screenshot of Payload vs. Launch Outcome scatter plot for all sites with payload slider.

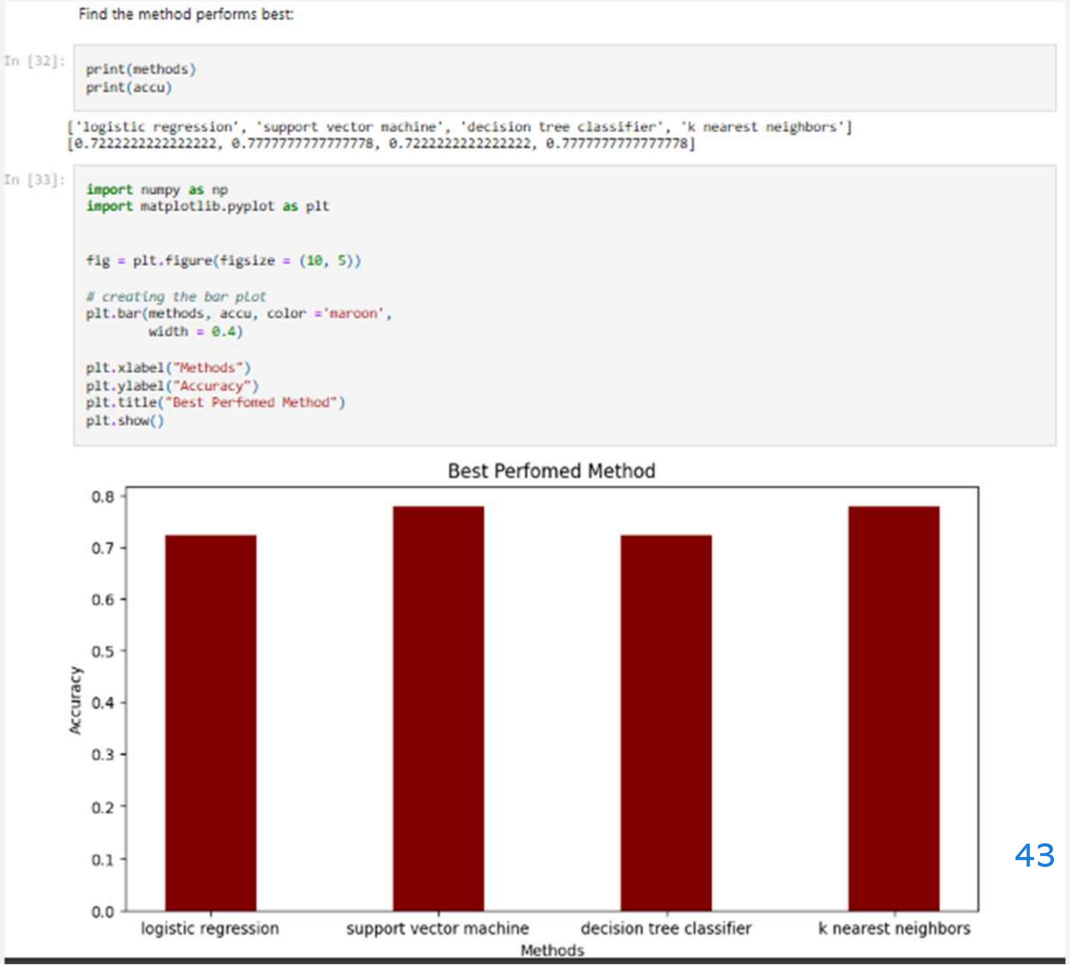
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- In the test set, the accuracy of two models was virtually the same at 77.78% and accuracy of other two models was same at 72.22%

- The highest classification accuracy models are support vector machine and K nearest neighbors



Find the method performs best:

In [32]:

```
print(methods)
print(accu)
```

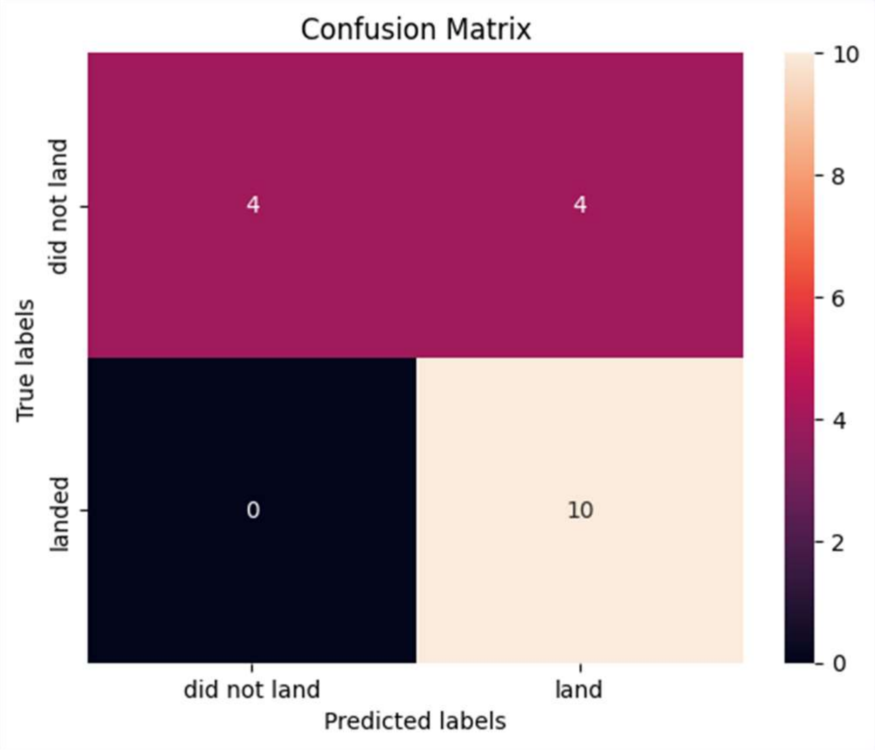
```
['logistic regression', 'support vector machine', 'decision tree classifier', 'k nearest neighbors']
[0.7222222222222222, 0.7777777777777778, 0.7222222222222222, 0.7777777777777778]
```

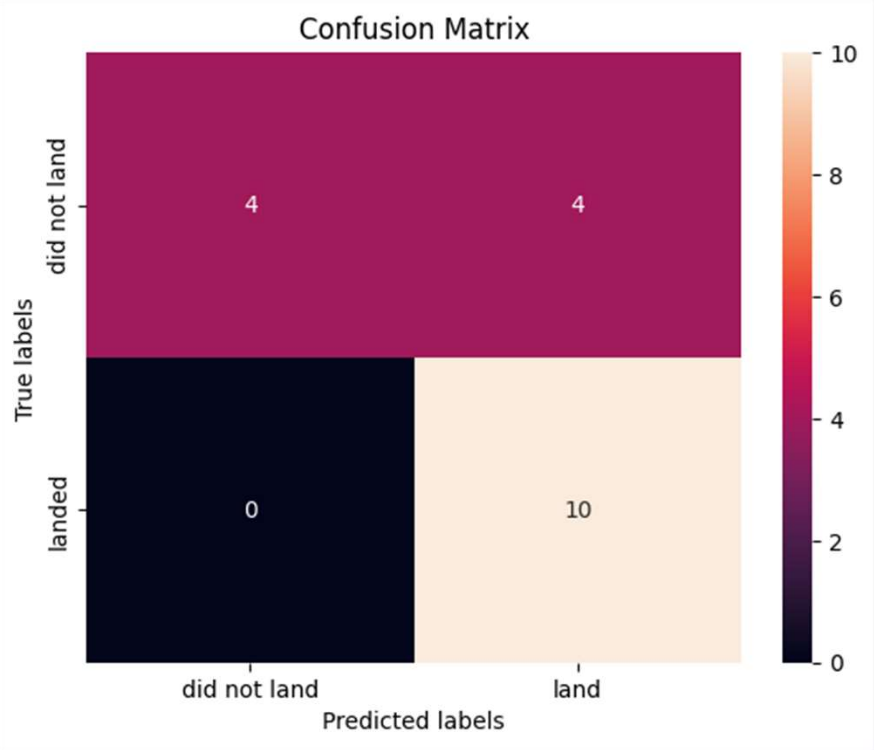
In [33]:

```
import numpy as np
import matplotlib.pyplot as plt


fig = plt.figure(figsize = (10, 5))

# creating the bar plot
plt.bar(methods, accu, color ='maroon',
        width = 0.4)

plt.xlabel("Methods")
plt.ylabel("Accuracy")
plt.title("Best Perfomed Method")
plt.show()
```

Confusion Matrix

- Two models predicted 10 successful landings when the true label was successful and 4 failed landings when the true label was failure. But there were also 4 predictions that said successful landings when the true label was failure (false positive).





Conclusions

- As the number of flights increased, the success rate increased, and recently it has exceeded 80%.
- Orbit types ES-L1, GEO, HEO, SSO, and have the highest success rates (100%).
- The launch success rate of low weighted payloads is higher than that of heavy weighted payloads
- The launch site is close to railways, highways, and coastline, but far from cities.
- The highest classification accuracy models are support vector machine and K nearest neighbors

Appendix

- GITHUB URL

  https://github.com/SR000777/Data-Science-and-Machine-Learning-Capstone-Project.git

- SPECIAL THANKS TO ALL THE INSTRUCTORS OF THE COURSE
- SPECIAL THANKS TO IBM SKILLS NETWORK TEAM
- SPECIAL THANKS TO EDX TEAM

Thank you!