IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

DEEPIKA CHARLY
12-03-2024

Outline

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

Executive Summary

PROJECT OVERVIEW:

The commercial space industry is booming, with companies like SpaceX leading the way in affordable space travel. In this project, we focus on Space Y, a new entrant founded by billionaire industrialist Allon Musk, aiming to compete with SpaceX. The objective is to determine the cost of each launch by predicting the success of landing and reusing the first stage of the Falcon 9 rocket.

- ✓ WEEK 1: The project kicks off with data collection using the SpaceX REST API and web scraping Wiki pages for Falcon 9 launch data. The data includes information on rocket specifications, payload details, launch and landing outcomes. We address challenges like null values and filter out Falcon 1 launches. The Payload Mass null values are imputed with the mean, preparing the dataset for analysis.

- ✓ WEEK 2: Exploratory Data Analysis (EDA) reveals insights into success rates, launch sites' impact, and correlations between different features. Notable findings include improved success rates since 2013 and variations among launch sites. The EDA sets the stage for building a predictive model.

- ✓ WEEK 3: Building an interactive dashboard using Plotly Dash for analyzing launch records and a Folium map to assess launch site proximity. The dashboard incorporates pie charts, scatter plots, and an interactive map to aid stakeholders in making informed decisions about optimal launch sites.

- ✓ WEEK 4: Machine learning techniques, including SVM, Classification Trees, and Logistic Regression, are employed to predict the success of landing the Falcon 9's first stage. The data is split into training and test sets, and hyperparameter optimization is performed to enhance model performance. The best-performing model is identified through testing.

This project equips Space Y with valuable insights to compete in the evolving commercial space industry. The predictive model and interactive dashboard empower decision-makers to optimize launch strategies and costs

Introduction

In the ever-evolving landscape of commercial space exploration, where pioneers like SpaceX are rewriting the rules, a new contender emerges – Space Y. Founded by the visionary billionaire industrialist Allon Musk, Space Y is poised to challenge the status quo and redefine the economics of space travel. As the space industry endeavors to make access to space more affordable and sustainable, understanding the intricacies of rocket launches becomes paramount.

The primary focus of this project is on unraveling the cost dynamics associated with space travel, particularly honing in on the pivotal factor – the successful landing and reuse of the first stage of the Falcon 9 rocket. SpaceX's Falcon 9 has set a precedent by demonstrating the reusability of its first stage, leading to significant cost reductions compared to traditional rocket launches. Our objective is twofold: to ascertain the cost implications of each launch and to predict the success of landing the first stage.

Problems to find:

1. Cost Determination: The commercial space age introduces a paradigm shift in cost structures, with companies like SpaceX offering more economical space travel. However, understanding the exact cost per launch remains a complex puzzle. Our project aims to decipher this puzzle by delving into the factors influencing launch costs, particularly focusing on the successful recovery of the Falcon 9's first stage.

2. First Stage Landing Prediction: The ability to predict whether the first stage of the Falcon 9 will successfully land is pivotal for optimizing launch strategies and costs. By leveraging machine learning and data analytics, we seek to develop a predictive model that foretells the outcome of each launch, enabling Space Y to make informed decisions about reusability and cost-effectiveness.

In essence, this project is at the intersection of cutting edge technology, data science, and the pioneering spirit of space exploration. As we embark on this journey, our aim is to equip SpaceY with the insights needed to compete in this fast paced industry and contribute to the ongoing revolution in commercial space travel.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - ❑ Using SpaceX Rest API
  - ❑ Using Web Scrapping from Wikipedia
- Perform data wrangling
  - ❑ Filtering the data
  - ❑ Dealing with missing values
  - ❑ Using One Hot Encoding to prepare the data to a binary classification
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

  - Building, tuning and evaluation of classification models to ensure the best results

Data Collection

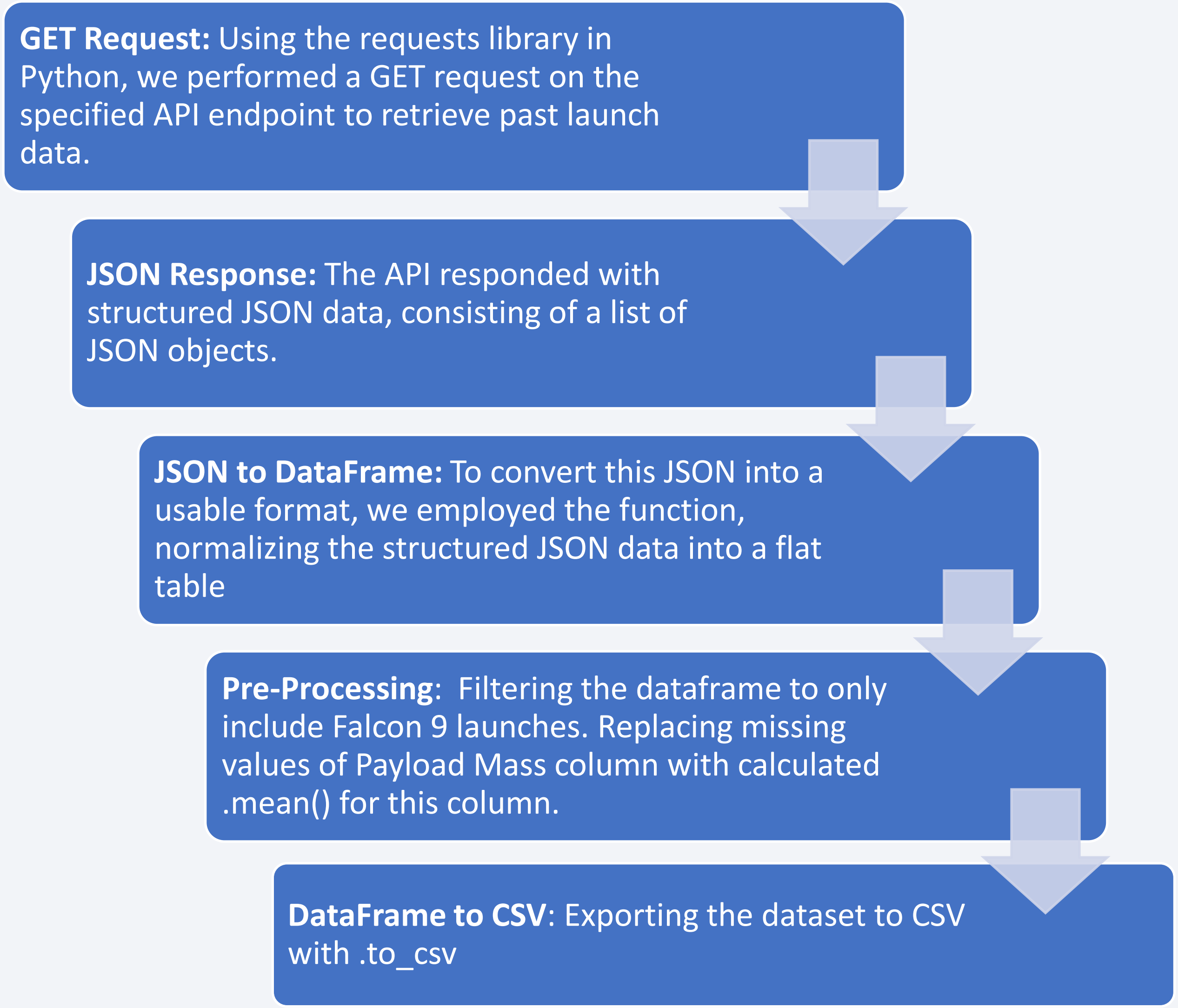
Our data collection process for the Space Y project involved retrieving valuable information on SpaceX launches, specifically focusing on Falcon 9 missions. We employed two primary methods: utilizing the **SpaceX REST API** and **Web Scraping relevant Wiki pages**.

Our meticulous data collection process ensured that we obtained a clean and meaningful dataset, paving the way for further visualization, analysis, and predictive modeling in subsequent stages of the project.

Data Collection – SpaceX API

Git Hub URL:

Data Collection API



GET Request: Using the requests library in Python, we performed a GET request on the specified API endpoint to retrieve past launch data.

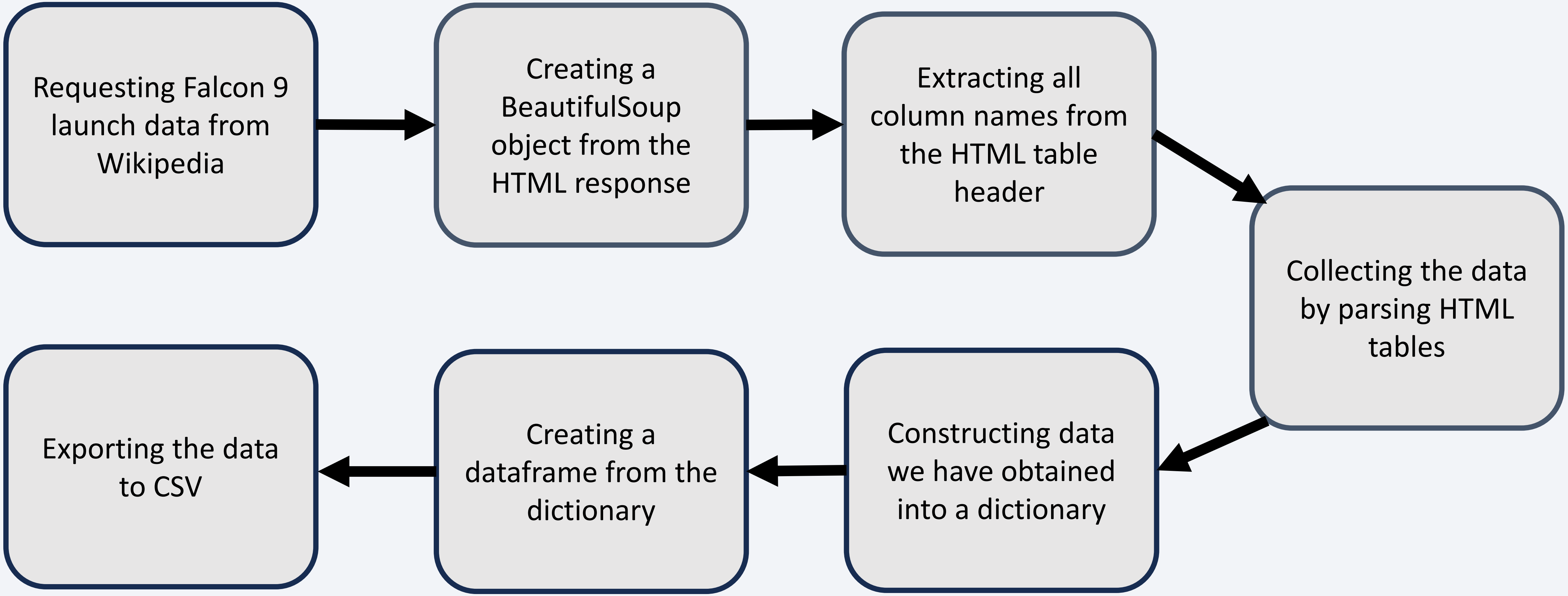
JSON Response: The API responded with structured JSON data, consisting of a list of JSON objects.

JSON to DataFrame: To convert this JSON into a usable format, we employed the function, normalizing the structured JSON data into a flat table

Pre-Processing: Filtering the dataframe to only include Falcon 9 launches. Replacing missing values of Payload Mass column with calculated .mean() for this column.

DataFrame to CSV: Exporting the dataset to CSV with .to_csv

Data Collection – Web Scraping


Requesting Falcon 9 launch data from Wikipedia

Creating a BeautifulSoup object from the HTML response

Extracting all column names from the HTML table header

Collecting the data by parsing HTML tables

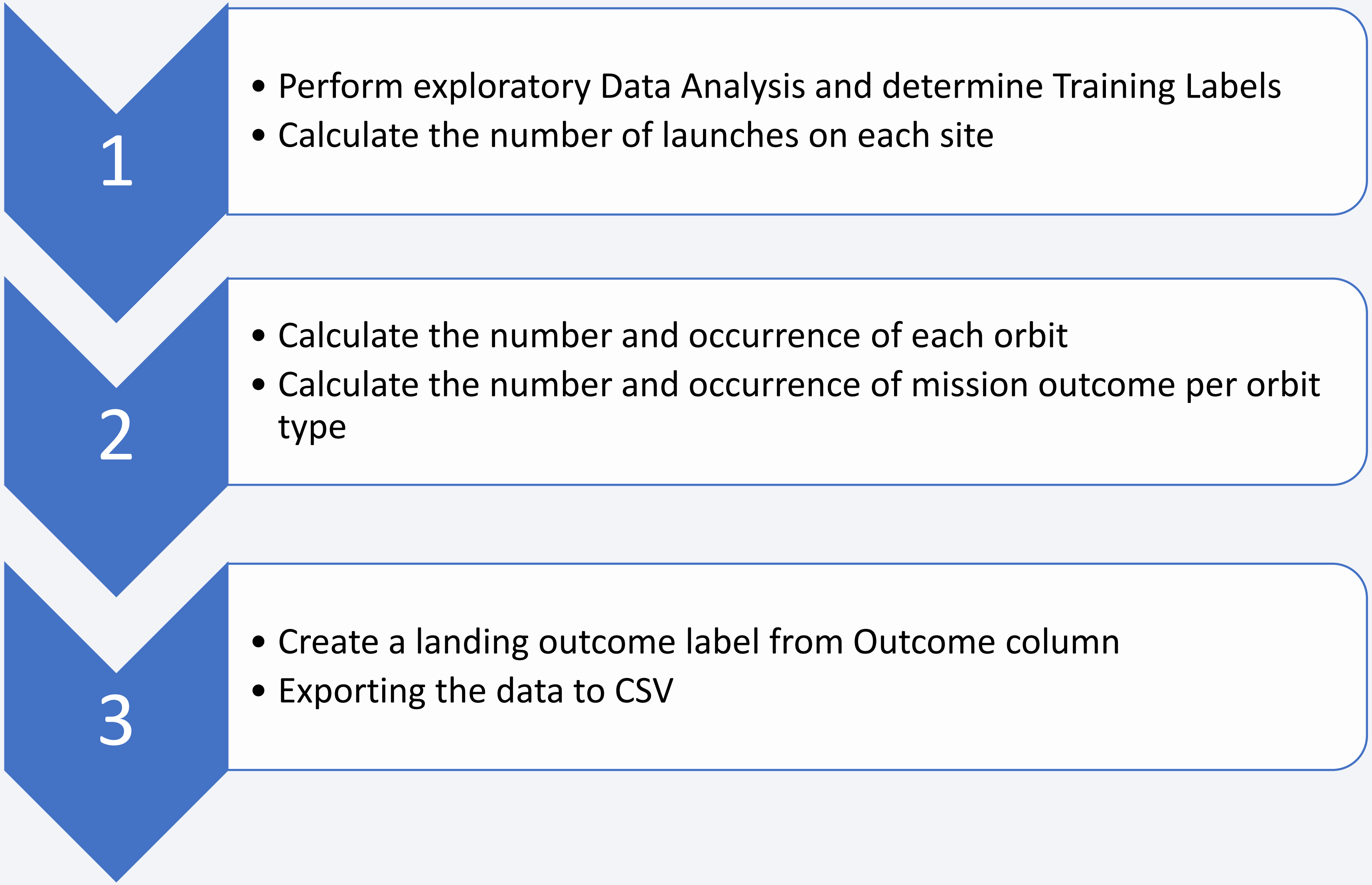
Constructing data we have obtained into a dictionary

Creating a dataframe from the dictionary

Exporting the data to CSV

Git Hub URL: [Data Collection – Web Scraping]

Data Wrangling



1

- Perform exploratory Data Analysis and determine Training Labels
- Calculate the number of launches on each site

2

- Calculate the number and occurrence of each orbit
- Calculate the number and occurrence of mission outcome per orbit type

3

- Create a landing outcome label from Outcome column
- Exporting the data to CSV

Git Hub URL:

Data Wrangling

EDA with Data Visualization

Charts plotted:

1. Flight Number vs. Payload Mass
2. Flight Number vs. Launch Site
3. Payload Mass vs. Launch Site
4. Orbit Type vs. Success Rate
5. Flight Number vs. Orbit Type
6. Payload Mass vs Orbit Type
7. Success Rate Yearly Trend

Scatter plots show the relationship between variables. If a relationship exists, they could be used in machine learning model.
Bar charts show comparisons among discrete categories. The goal is to show the relationship between the specific categories being compared and a measured value.
Line charts show trends in data over time (time series).

Git Hub URL:

EDA with Data Visualization

EDA with SQL

Performed SQL queries:

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string ‘CCA'
- 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 ground pad was achieved
- Listing the names of the boosters which have success in drone ship 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 the months in year 2015
- 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

GIT HUB URL: EDA With SQL

Build an Interactive Map with Folium

Markers of all Launch Sites:

- Added Marker with Circle, Popup Label and Text Label of NASA Johnson Space Center using its latitude and longitude coordinates as a start location.

- Added Markers with Circle, Popup Label and Text Label of all Launch Sites using their latitude and longitude coordinates to show their geographical locations and proximity to Equator and coasts.

Coloured Markers of the launch outcomes for each Launch Site:

- Added coloured Markers of success (Green) and failed (Red) launches using Marker Cluster to identify which launch sites have relatively high success rates.

Distances between a Launch Site to its proximities:

- Added coloured Lines to show distances between the Launch Site KSC LC-39A (as an example) and its proximities like Railway, Highway, Coastline and Closest City.

GIT HUB URL: Folium Interactive Map

Build a Dashboard with Plotly Dash

Launch Sites Dropdown List:

- Added a dropdown list to enable Launch Site selection.

Pie Chart showing Success Launches (All Sites/Certain Site):

- Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site, if a specific Launch Site was selected.

Slider of Payload Mass Range:

- Added a slider to select Payload range.

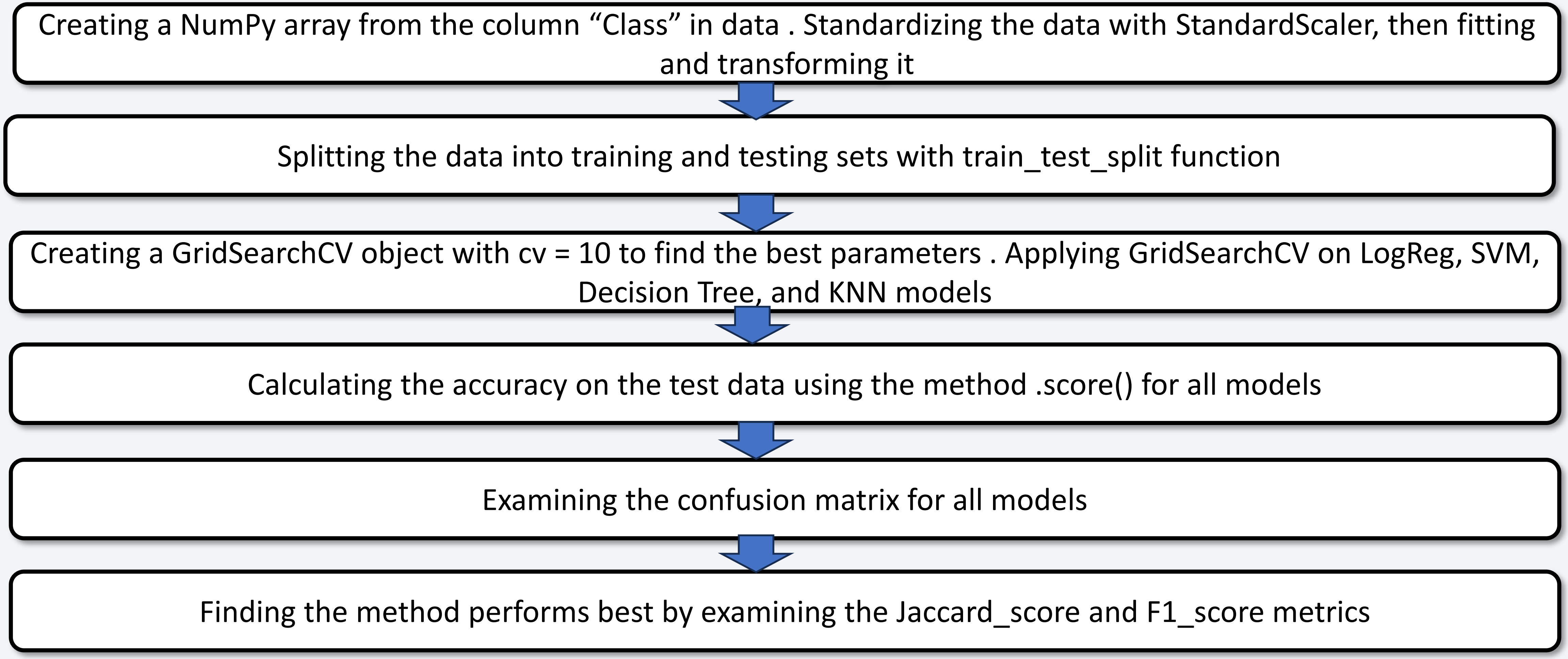
Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions:

- Added a scatter chart to show the correlation between Payload and Launch Success

GIT HUB URL:

[Dashboard with Plotly Dash]()

Predictive Analysis (Classification)



Creating a NumPy array from the column “Class” in data . Standardizing the data with StandardScaler, then fitting and transforming it

Splitting the data into training and testing sets with train_test_split function

Creating a GridSearchCV object with cv = 10 to find the best parameters . Applying GridSearchCV on LogReg, SVM, Decision Tree, and KNN models

Calculating the accuracy on the test data using the method .score() for all models

Examining the confusion matrix for all models

Finding the method performs best by examining the Jaccard_score and F1_score metrics

GIT HUB URL: Classification

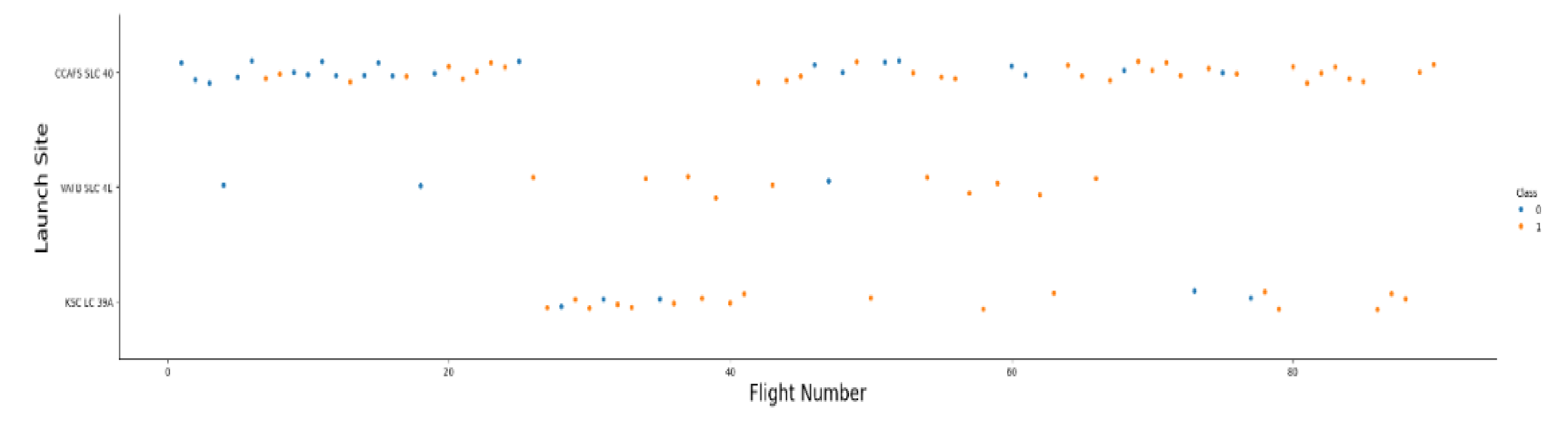
Results

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

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site



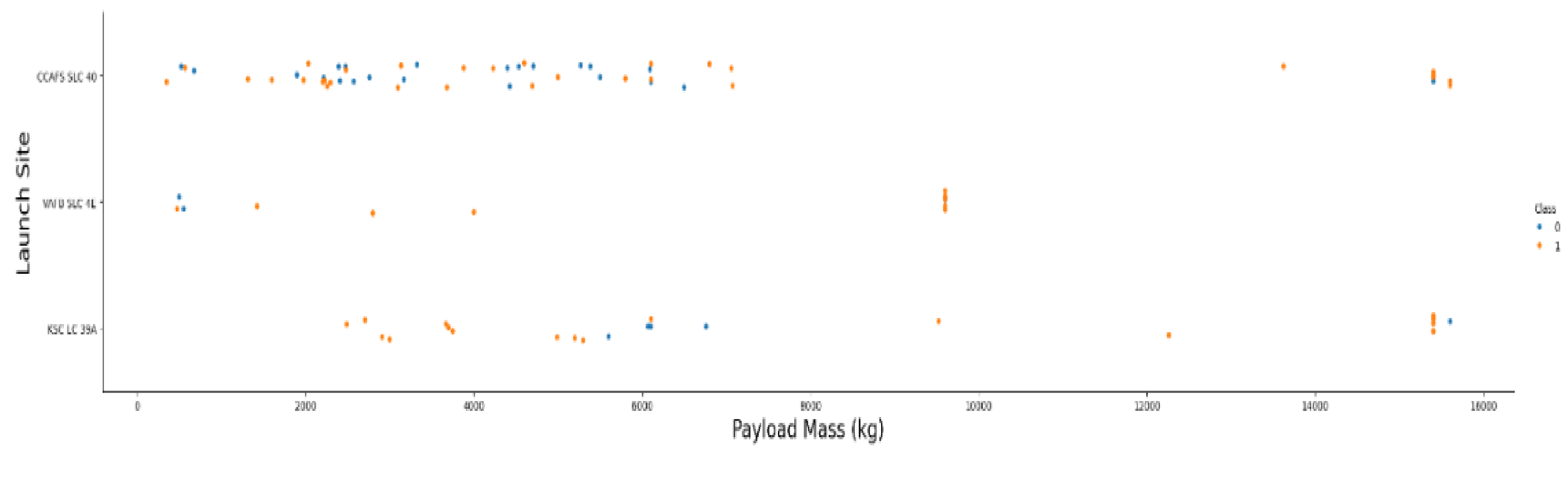
Explanation:

- The earliest flights all failed while the latest flights all succeeded.
- The CCAFS SLC 40 launch site has about a half of all launches.
- VAFB SLC 4E and KSC LC 39A have higher success rates.
- It can be assumed that each new launch has a higher rate of success.

Payload vs. Launch Site

Explanation:

- For every launch site the higher the payload mass, the higher the success rate.
- Most of the launches with payload mass over 7000 kg were successful.
- KSC LC 39A has a 100% success rate for payload mass under 5500 kg too.



Success Rate vs. Orbit Type

Explanation:

• Orbits with 100% success rate:

- ES-L1, GEO, HEO, SSO

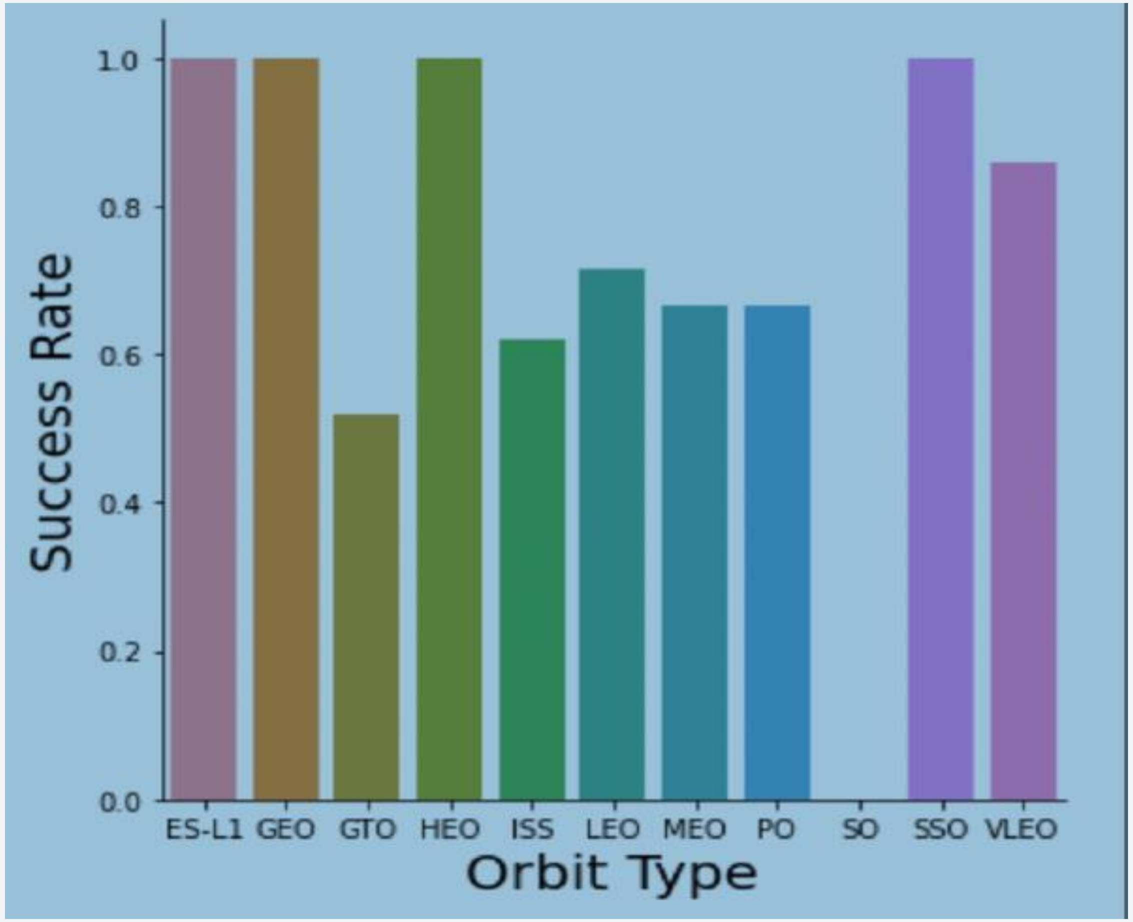
• Orbits with 0% success rate:

- SO

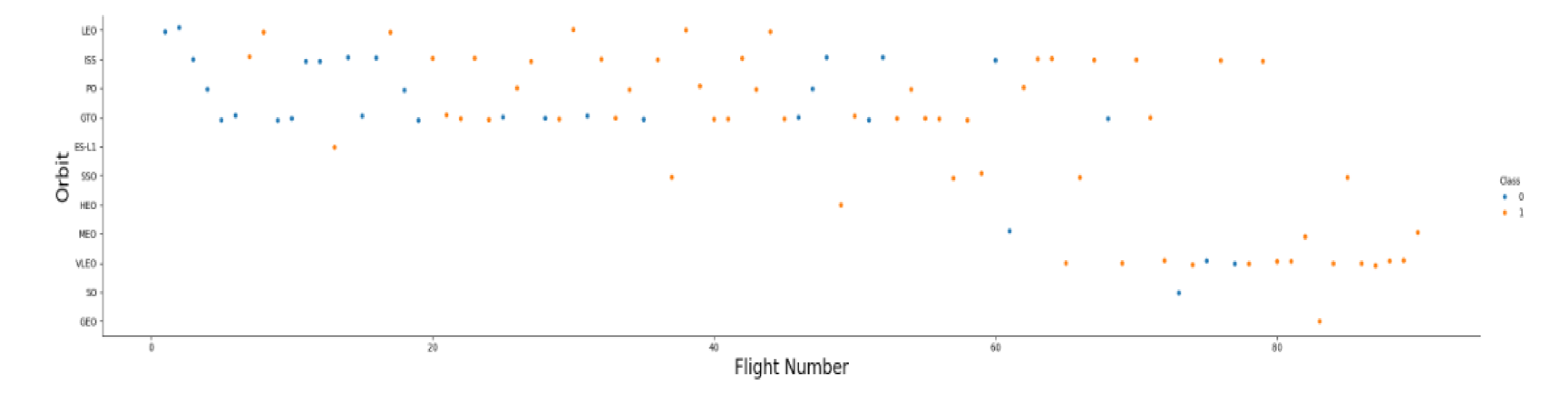
• Orbits with success rate

between 50% and 85%:

- GTO, ISS, LEO, MEO, PO



Flight Number vs. Orbit Type



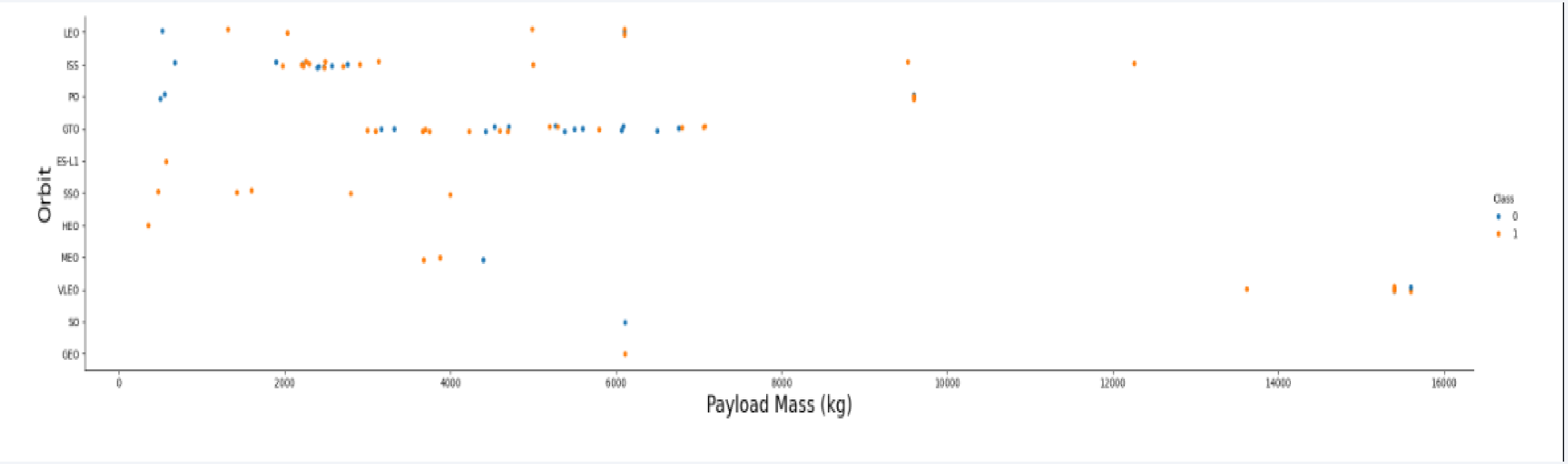
Explanation:

• 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.

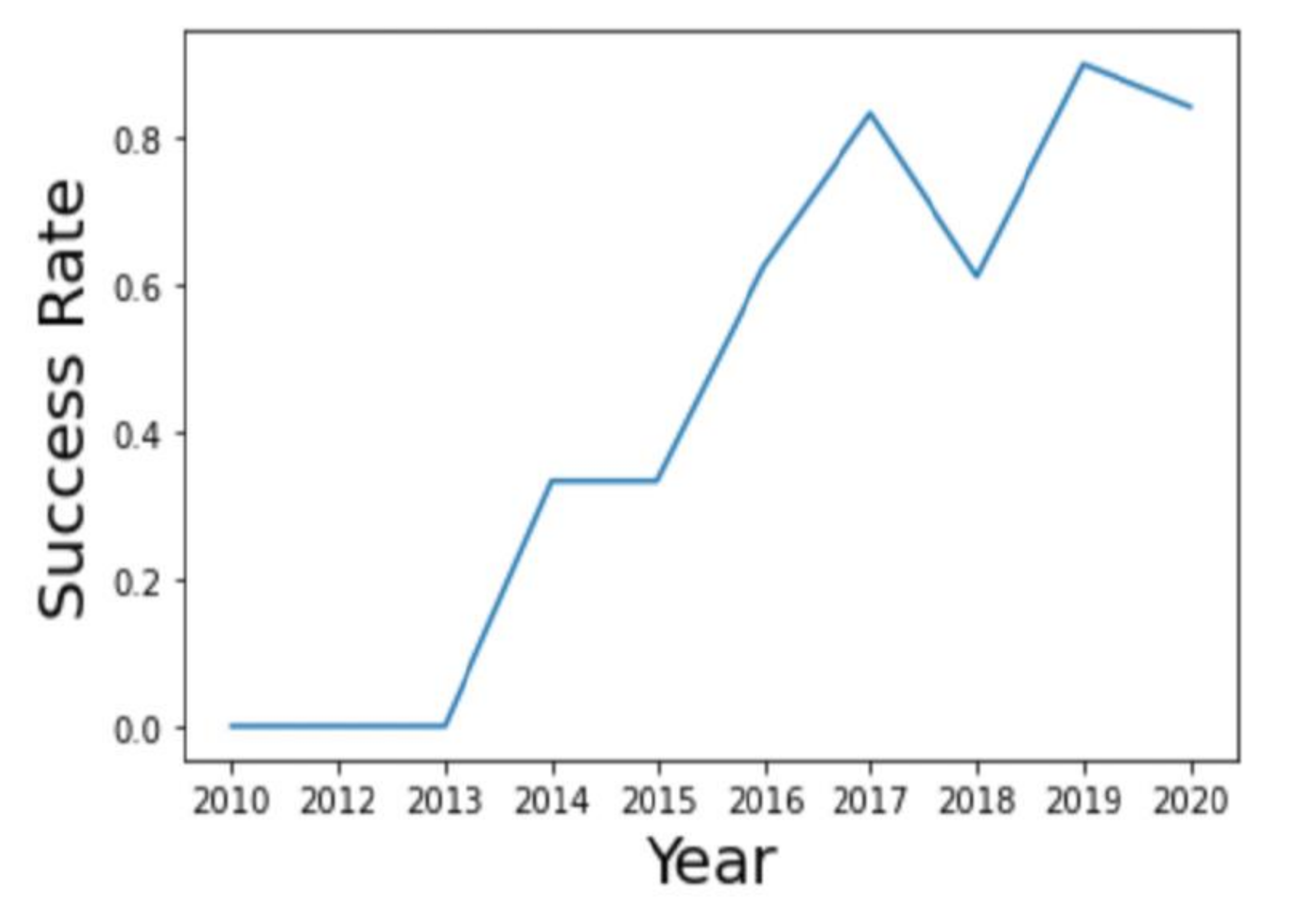
Payload vs. Orbit Type

Explanation:

- Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.



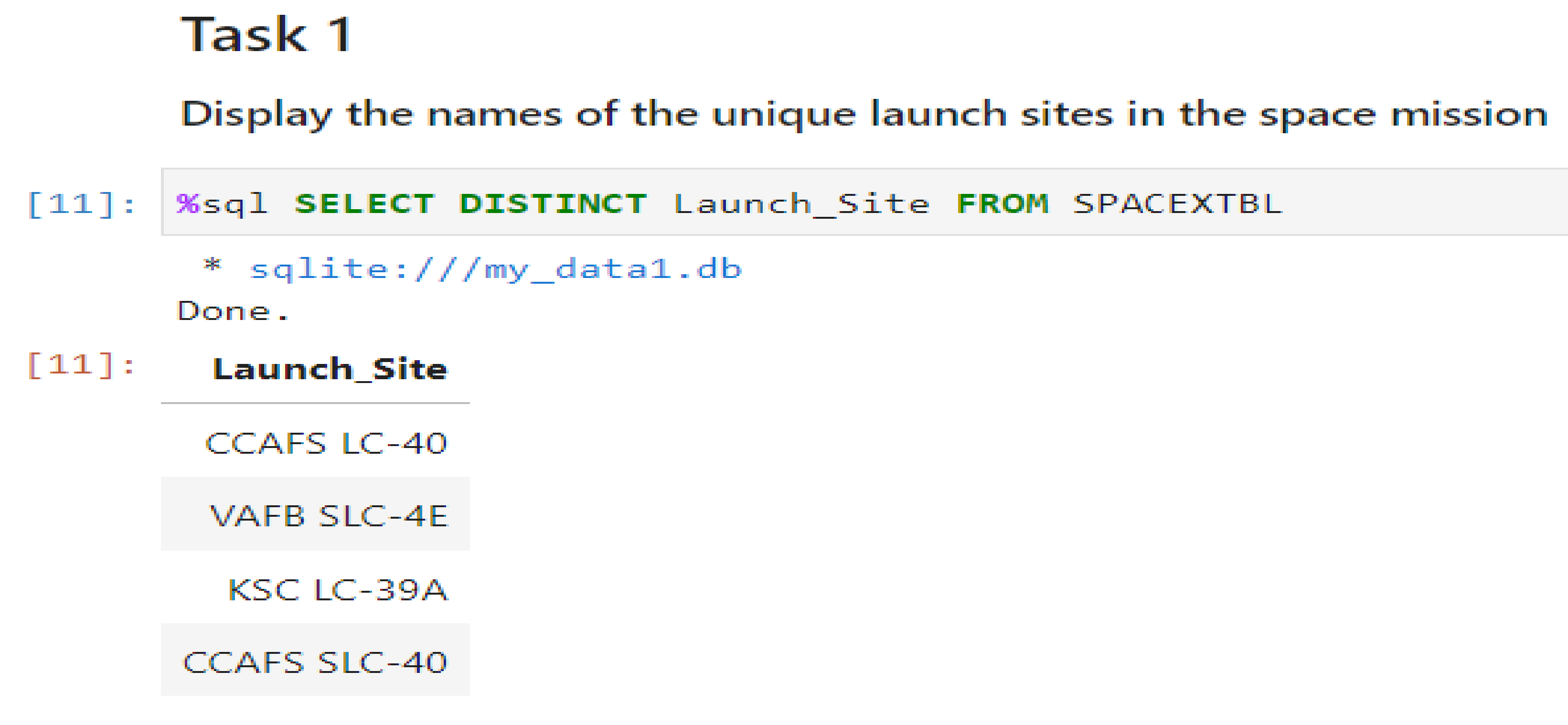
Launch Success Yearly Trend



Explanation:

- The success rate since 2013 kept increasing till 2020.

All Launch Site Names



Task 1

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

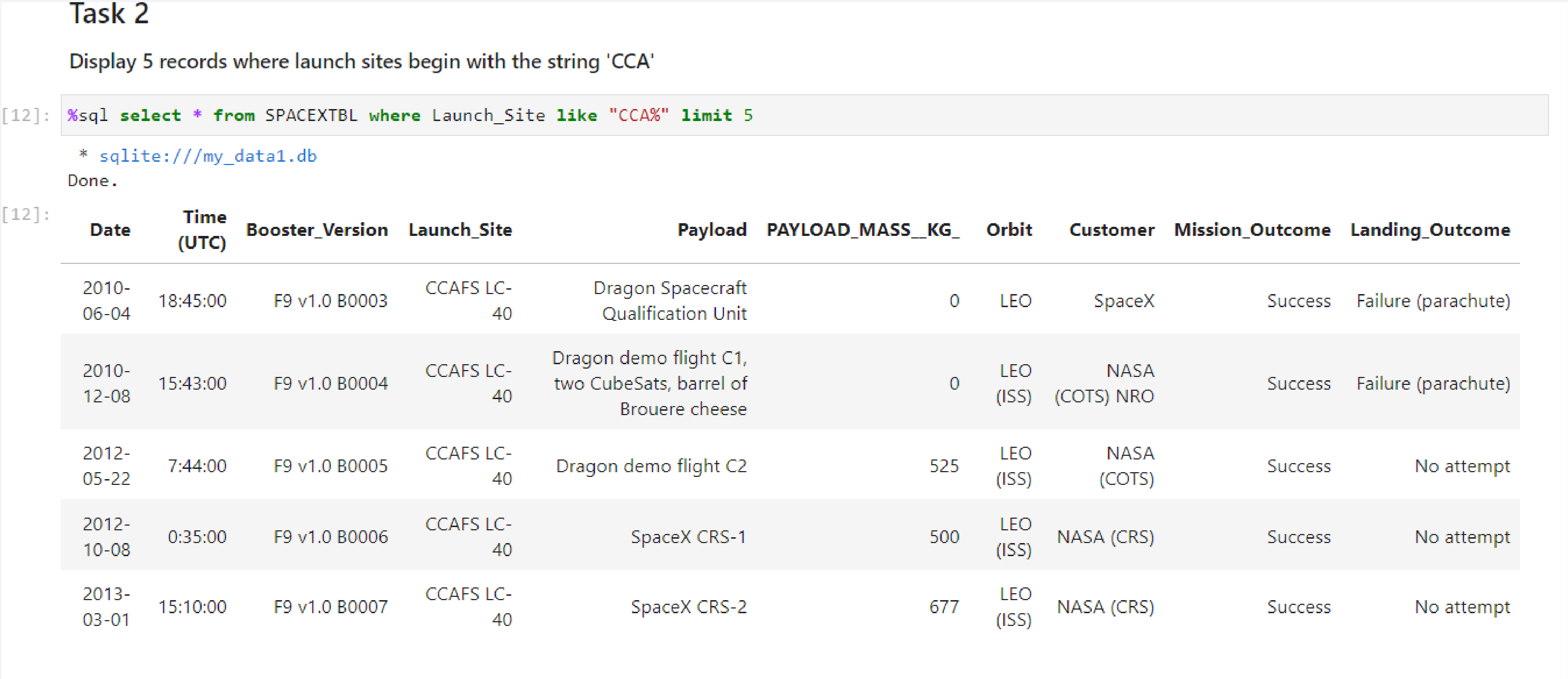
```
[11]: %sql SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

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

[11]:

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'

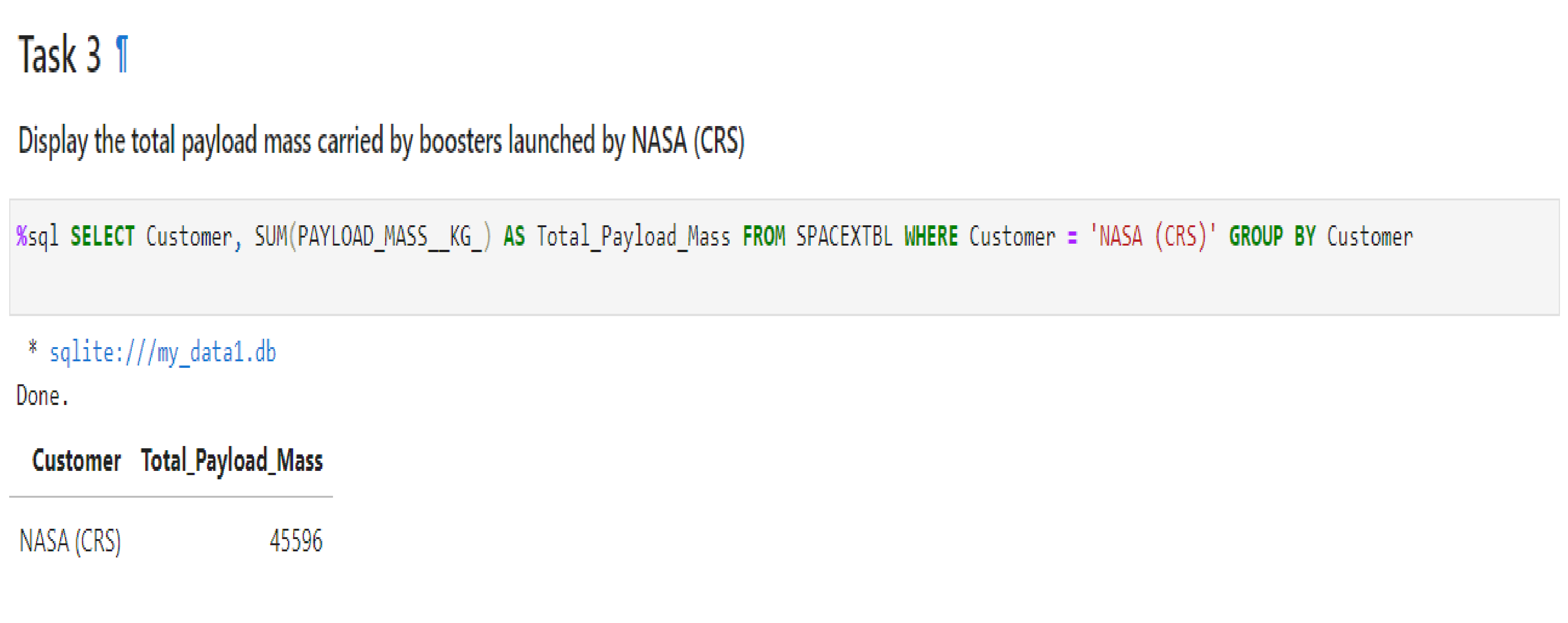
```
[12]: %sql select * from SPACEXTBL where Launch_Site like "CCA%" limit 5
```

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

[12]:

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 ¶

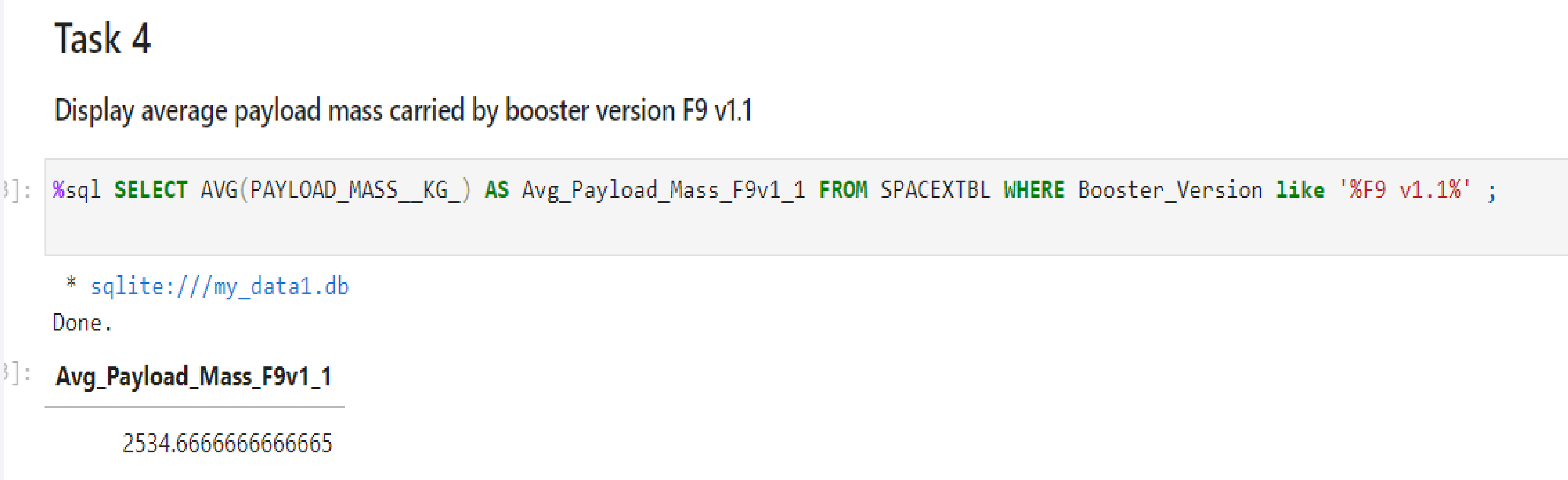
Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT Customer, SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass FROM SPACEXTBL WHERE Customer = 'NASA (CRS)' GROUP BY Customer
```

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

Customer	Total_Payload_Mass
NASA (CRS)	45596

Average Payload Mass by F9 v1.1



Task 4

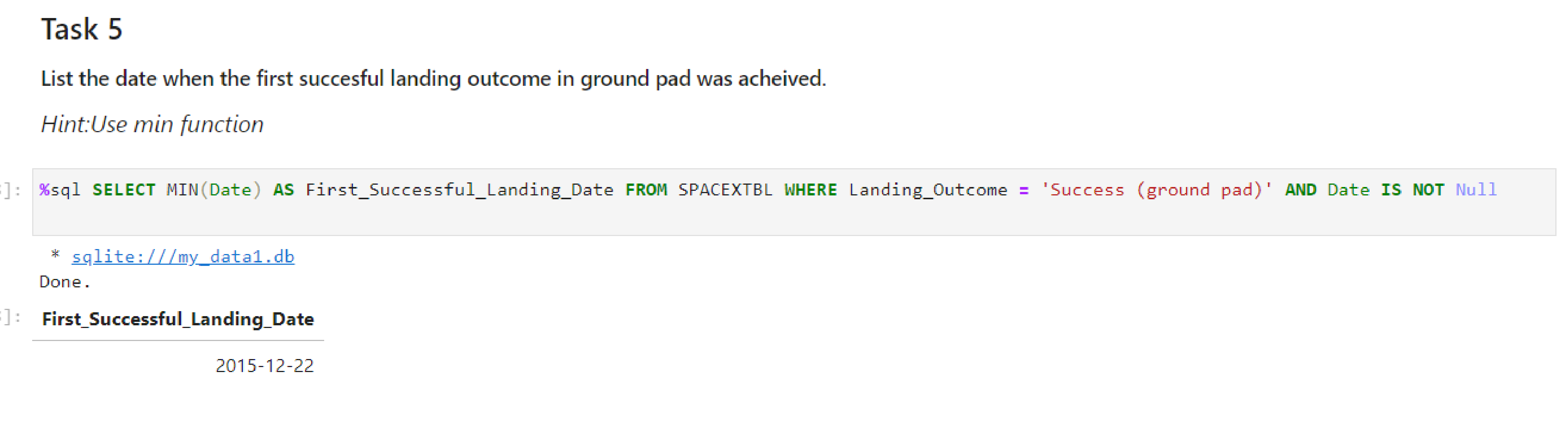
Display average payload mass carried by booster version F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS Avg_Payload_Mass_F9v1_1 FROM SPACEXTBL WHERE Booster_Version like '%F9 v1.1%' ;
```

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

Avg_Payload_Mass_F9v1_1
2534.6666666666665

First Successful Ground Landing Date



Task 5

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

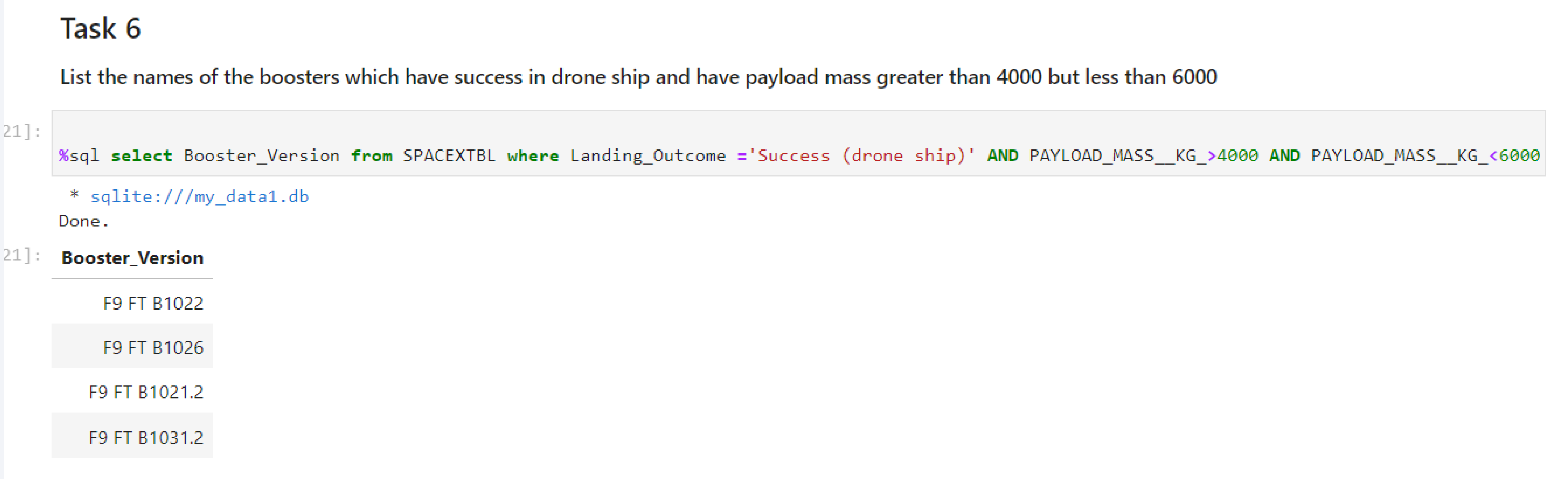
Hint:Use min function

```
%sql SELECT MIN(Date) AS First_Successful_Landing_Date FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)' AND Date IS NOT Null
```

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

First_Successful_Landing_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

21]:

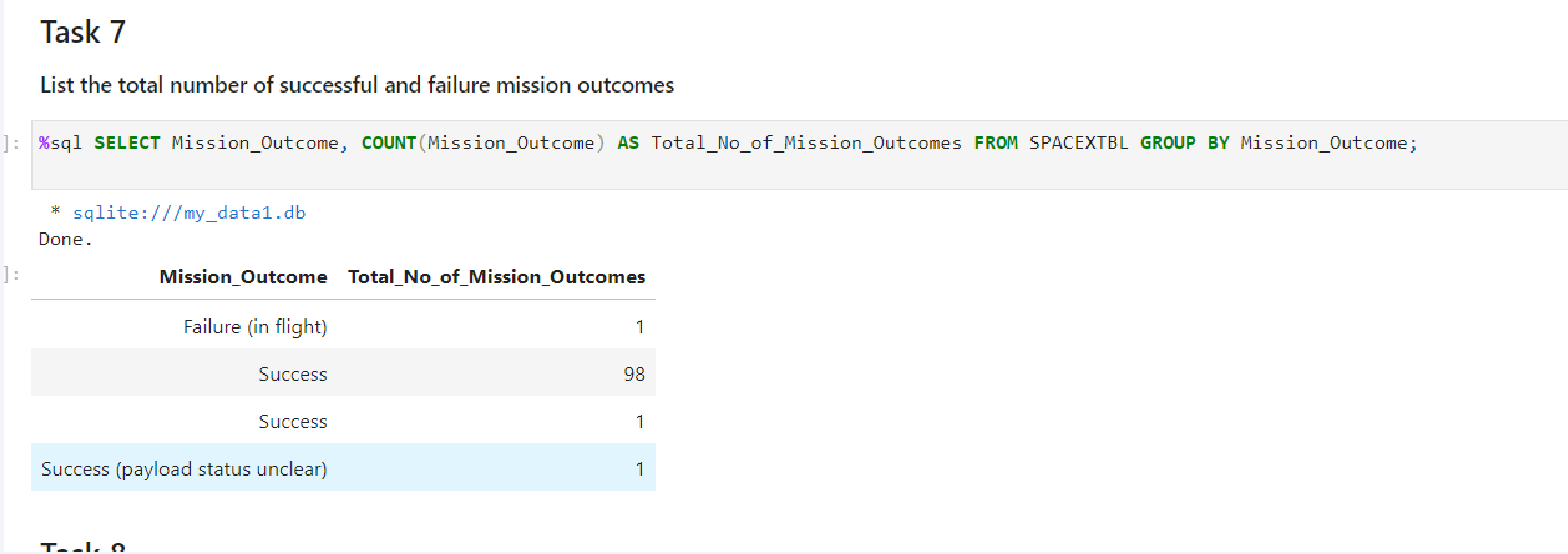
```
%sql select Booster_Version from SPACEXTBL where Landing_Outcome ='Success (drone ship)' AND PAYLOAD_MASS__KG_>4000 AND PAYLOAD_MASS__KG_<6000
```

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

21]:

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

Total Number of Successful and Failure Mission Outcomes



Task 7

List the total number of successful and failure mission outcomes

```
]: %sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS Total_No_of_Mission_Outcomes FROM SPACEXTBL GROUP BY Mission_Outcome;
```

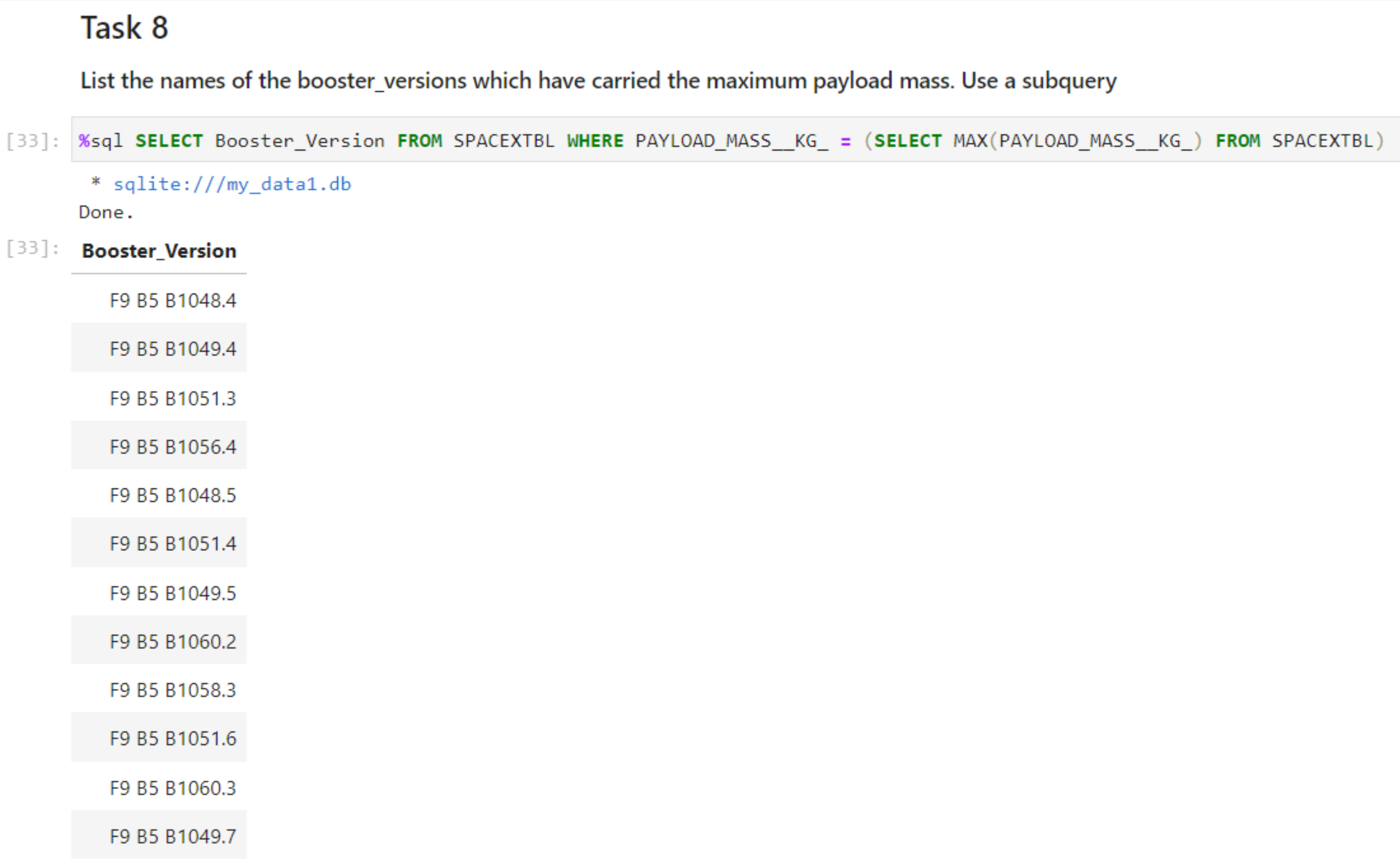
```
 * sqlite:///my_data1.db
Done.
```

]:

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

Task 8

Boosters Carried Maximum Payload



Task 8

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

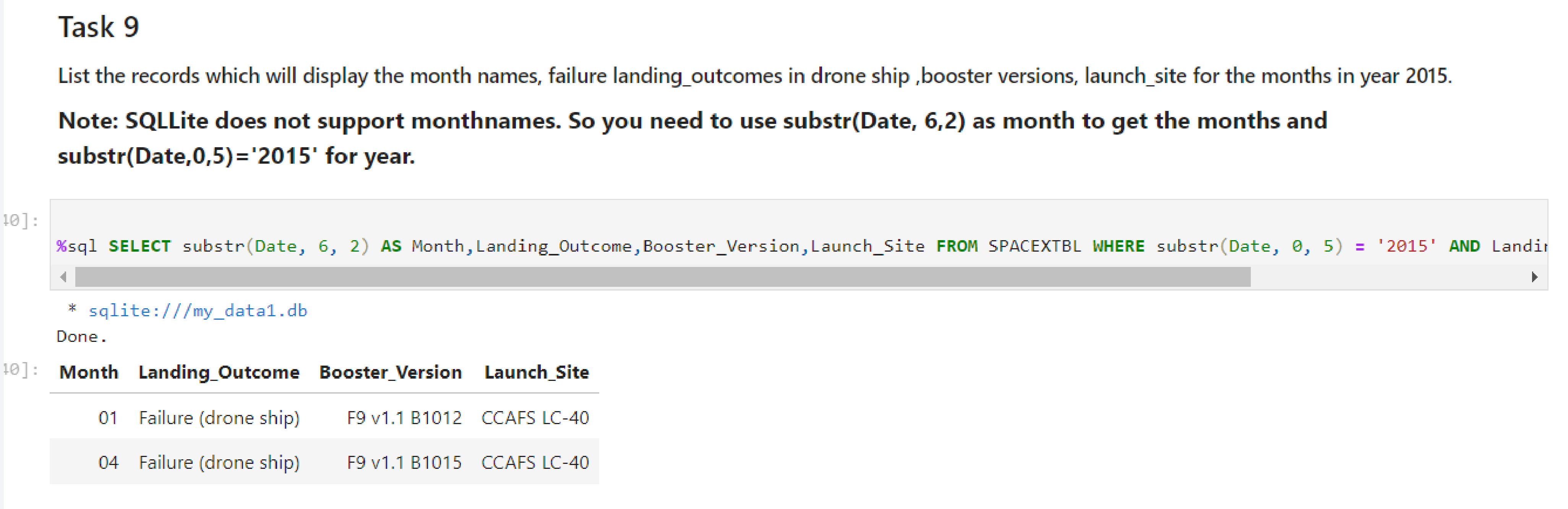
```
[33]: %sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

[33]:

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.

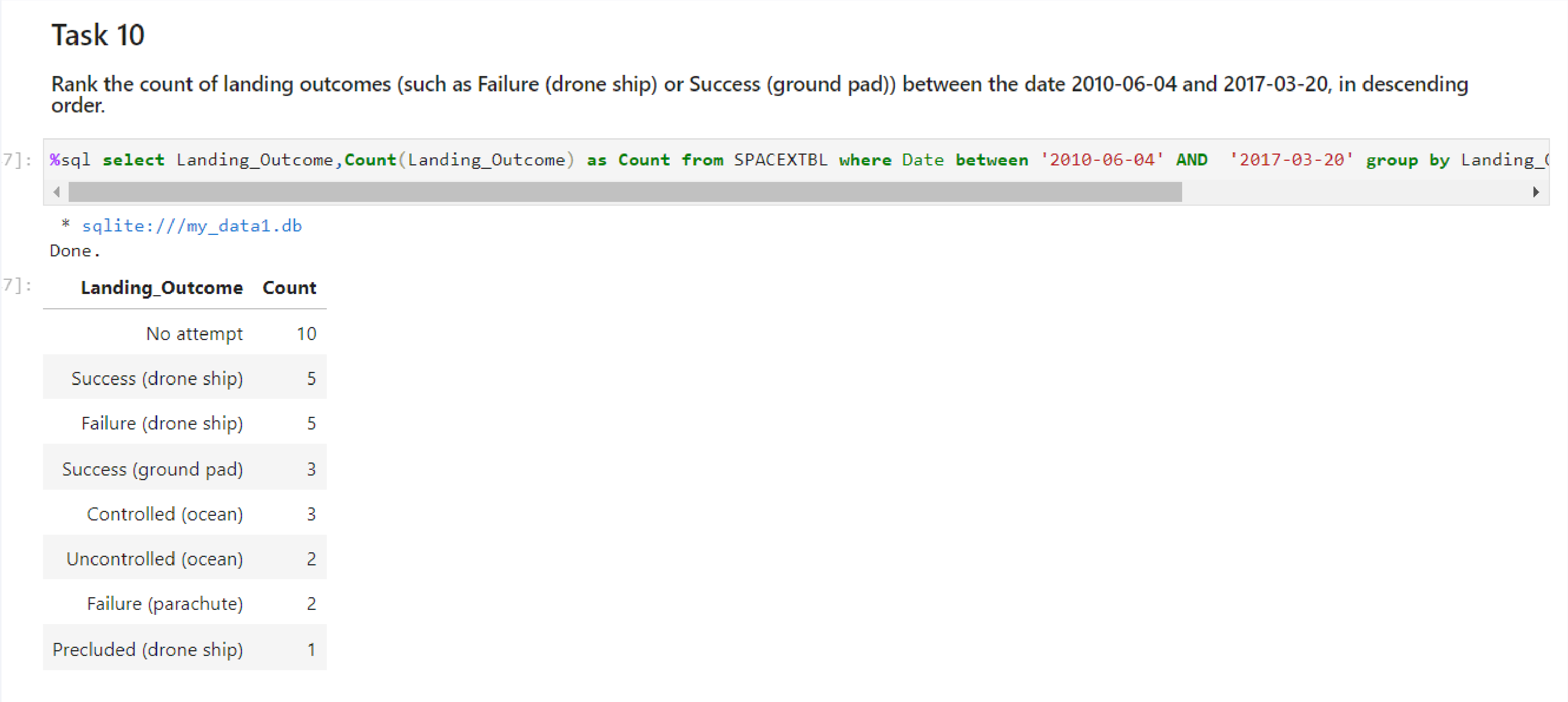
```
10]:
%sql SELECT substr(Date, 6, 2) AS Month,Landing_Outcome,Booster_Version,Launch_Site FROM SPACEXTBL WHERE substr(Date, 0, 5) = '2015' AND Landir
```

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

10]:

Month	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

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.

```
7]: %sql select Landing_Outcome,Count(Landing_Outcome) as Count from SPACEXTBL where Date between '2010-06-04' AND  '2017-03-20' group by Landing_(
```

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

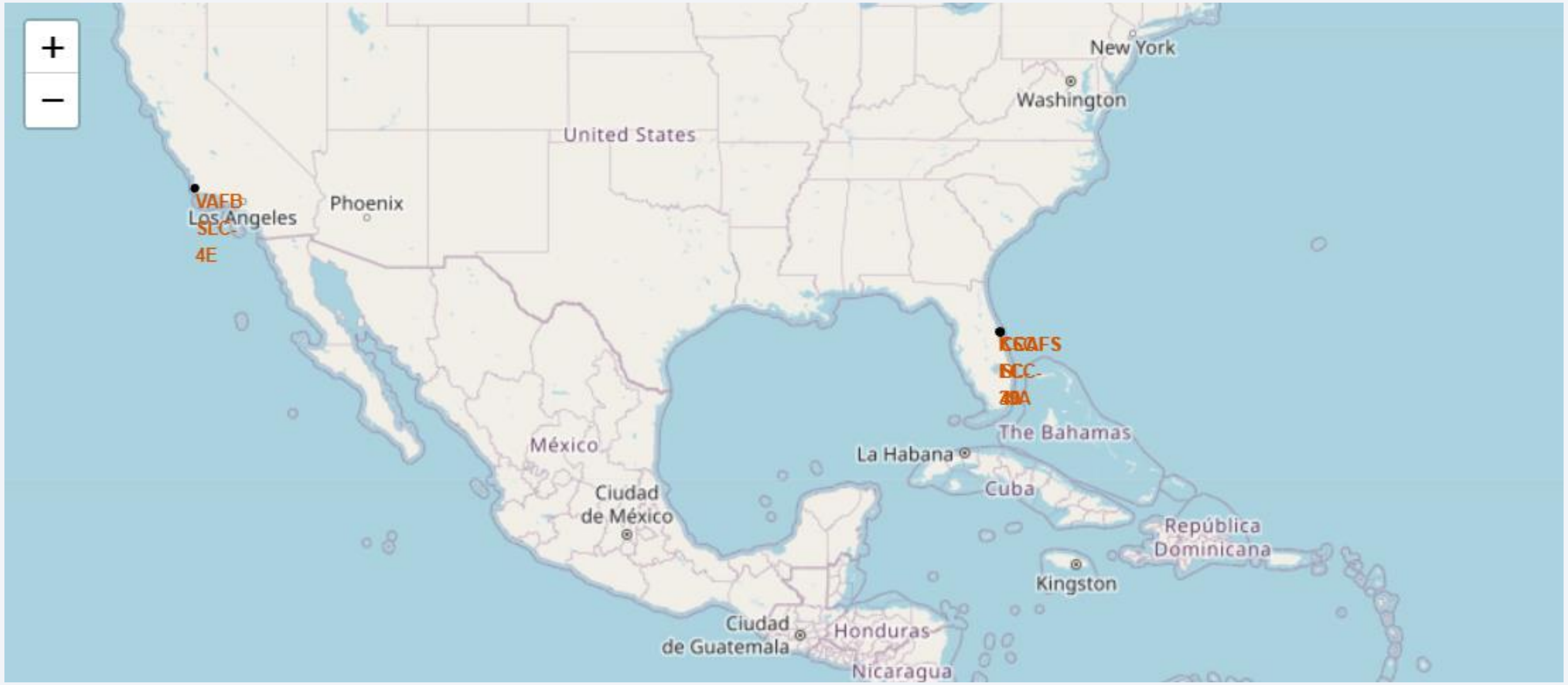
7]:

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

All launch sites’ location markers on a global map



Explanation:

- Most of Launch sites are in proximity to the Equator line.
- All launch sites are in very close proximity to the coast,

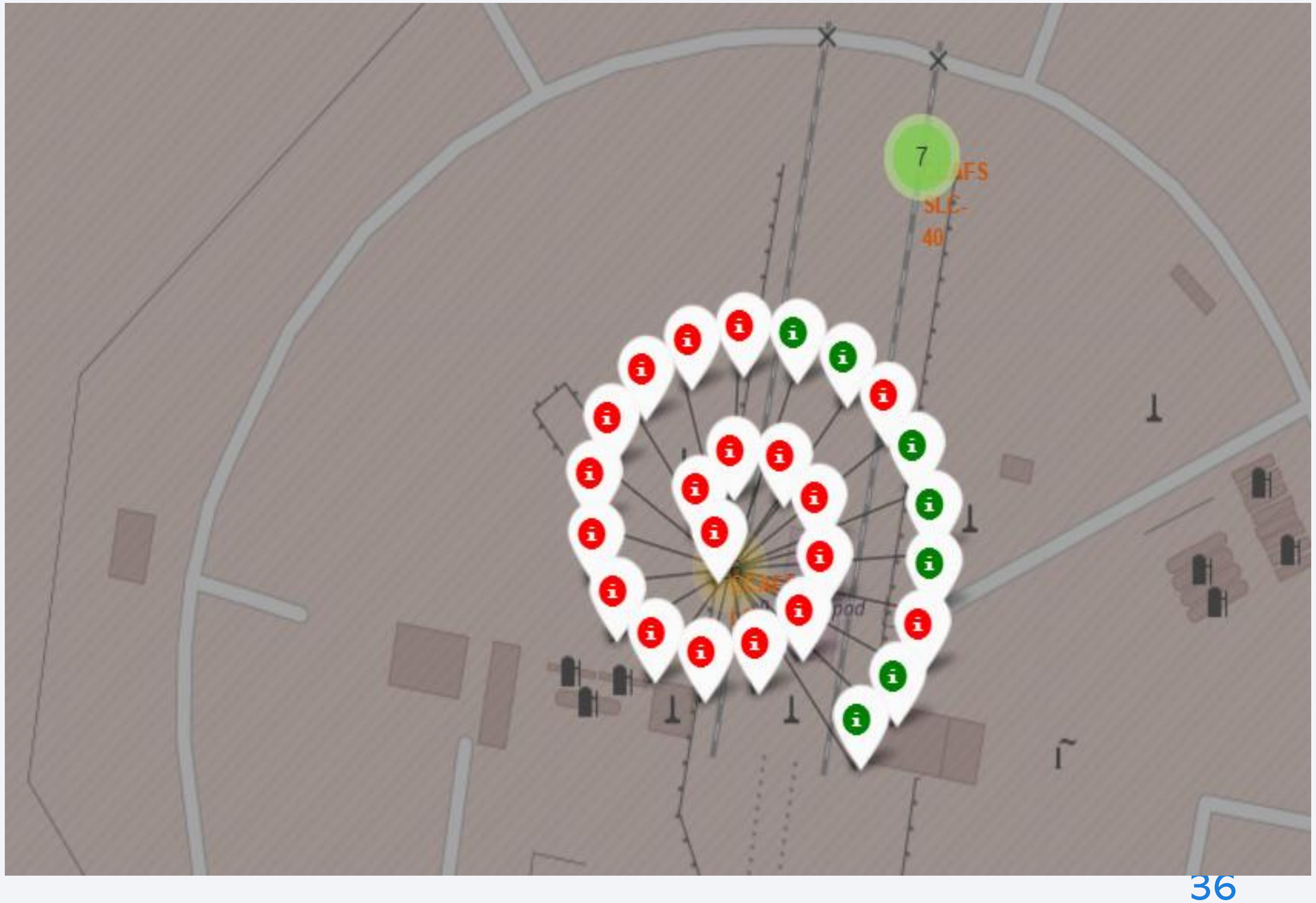
Color-labeled launch records on the map

- Explanation:

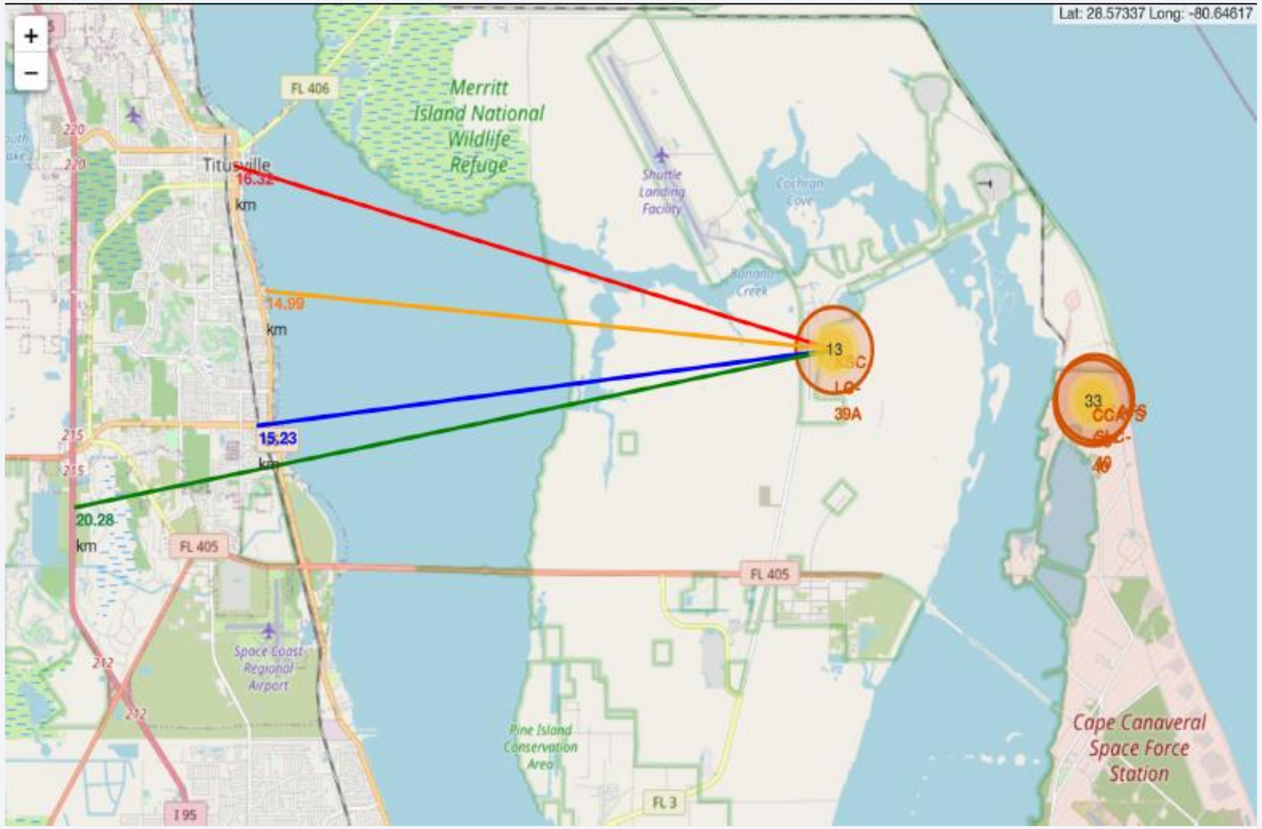
From the color-labeled markers we should be able to easily identify which launch sites have relatively high success rates.

- Green Marker =Successful Launch

- Red Marker = Failed Launch



Distance from the launch site KSC LC-39A to its proximities



Lat: 28.57337 Long: -80.64617
+
−
FL 406
Merritt Island National Wildlife Refuge
Shuttle Landing Facility
Cochran Cove
Banana Creek
Titusville
16.32 km
14.99 km
15.23 km
20.28 km
13
KSC LC-39A
33
CCAFS SLC-40
FL 405
FL 405
Space Coast Regional Airport
Pine Island Conservation Area
Cape Canaveral Space Force Station
FL 3
I 95

Explanation:

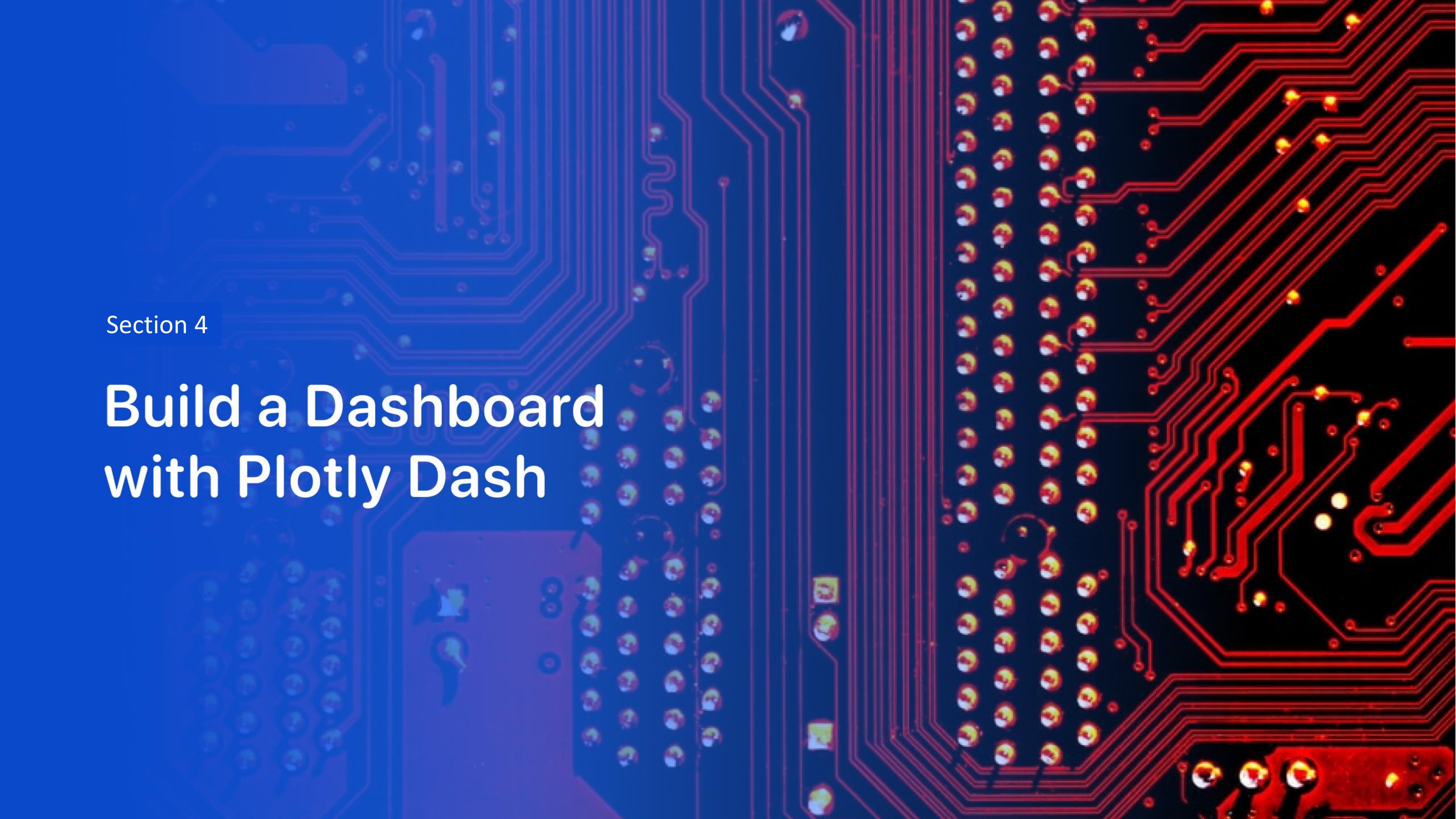
- From the visual analysis of the launch site KSC LC-39A we can clearly see that it is:

  - relative close to railway (15.23 km)
  - relative close to highway (20.28 km)
  - relative close to coastline (14.99 km)

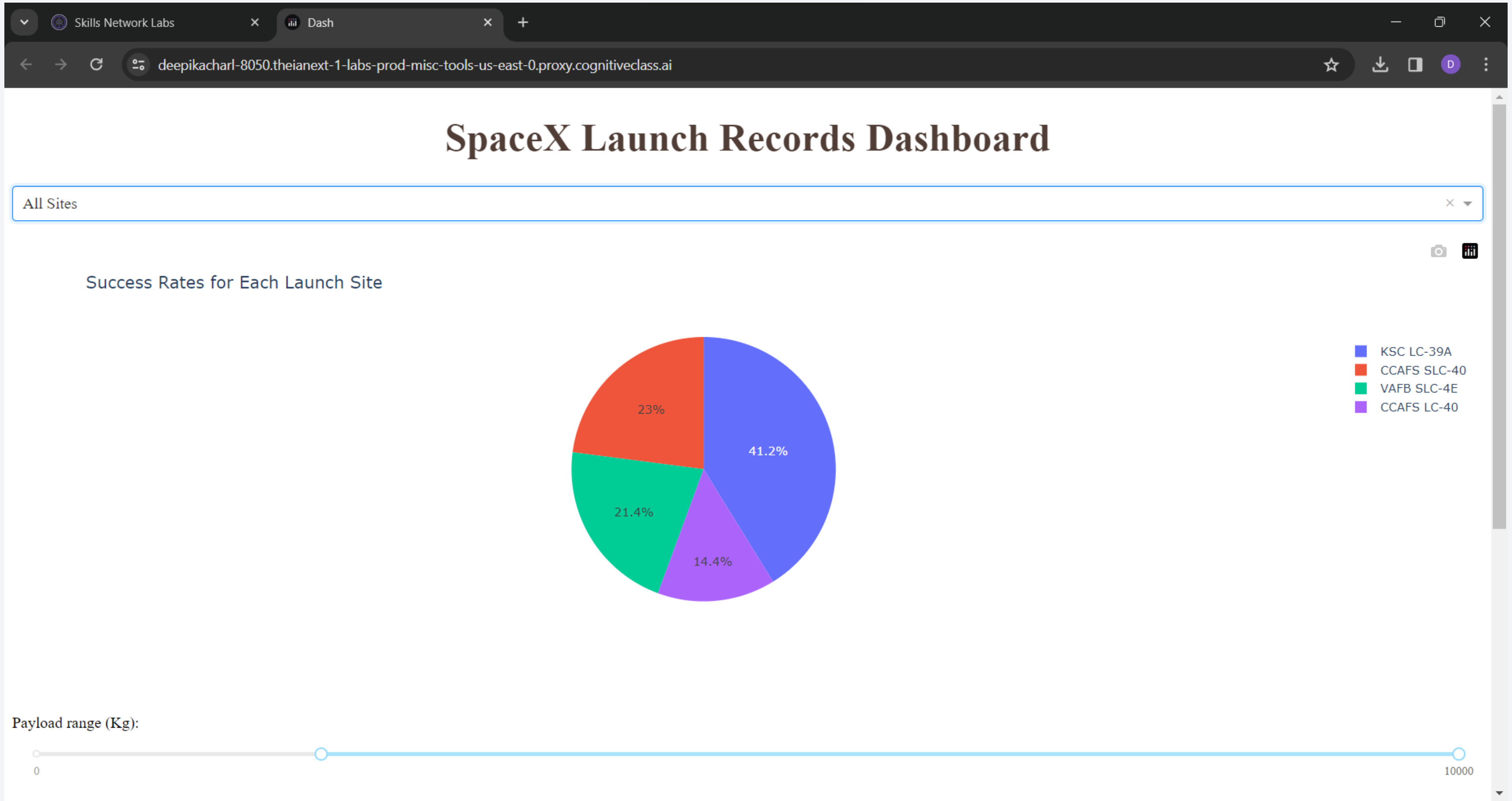
- Also the launch site KSC LC-39A is relative close to its closest city Titusville (16.32 km).
- Failed rocket with its high speed can cover distances like 15-20 km in few seconds. It could be potentially dangerous to populated areas.

Section 4

Build a Dashboard with Plotly Dash



Launch success count for all sites



Skills Network Labs
Dash
deepikacharl-8050.theianext-1-labs-prod-misc-tools-us-east-0.proxy.cognitiveclass.ai
SpaceX Launch Records Dashboard
All Sites
Success Rates for Each Launch Site
23%
41.2%
21.4%
14.4%
KSC LC-39A
CCAFS SLC-40
VAFB SLC-4E
CCAFS LC-40
Payload range (Kg):
0
10000

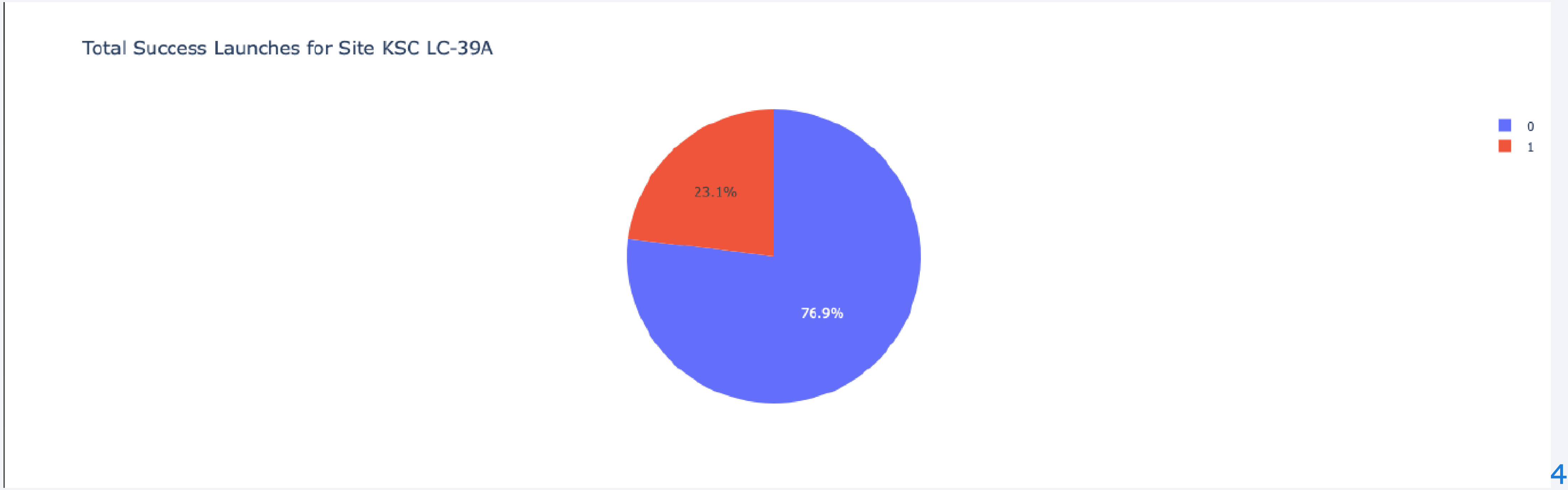
Explanation:

- The chart clearly shows that from all the sites, KSC LC-39A has the most successful launches

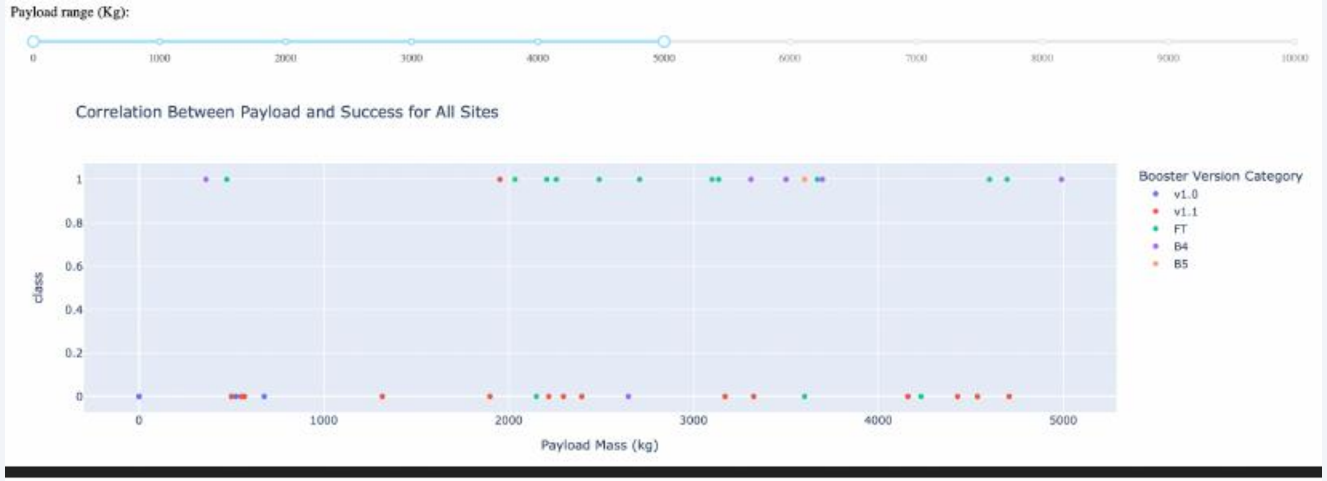
Launch site with highest launch success ratio

Explanation:

- KSC LC-39A has the highest launch success rate (76.9%) with 10 successful and only 3 failed landings.

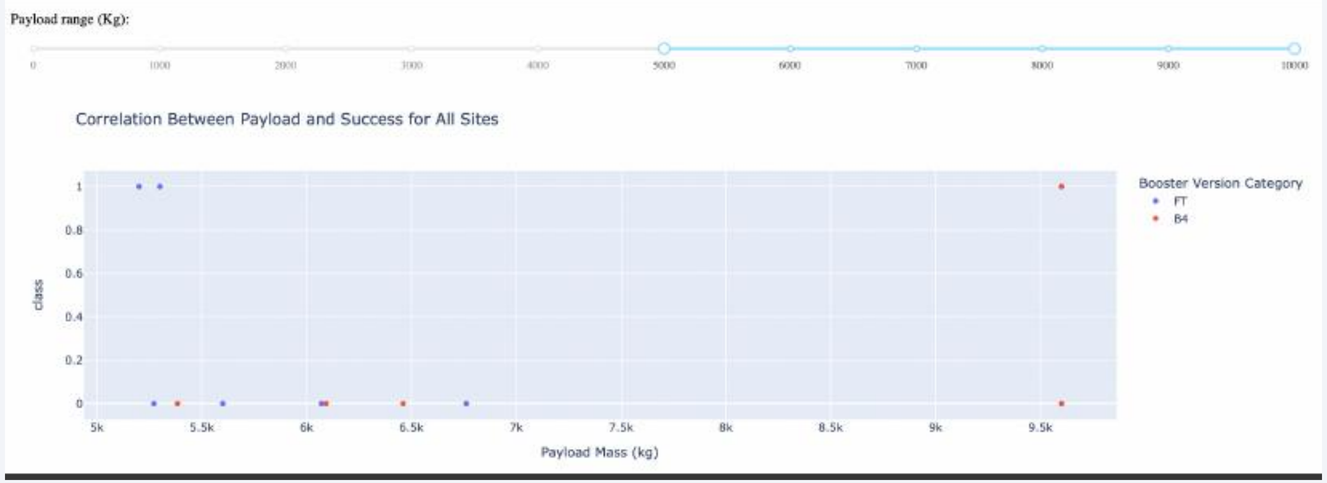


Payload Mass vs. Launch Outcome for all sites



Payload range (Kg):

Correlation Between Payload and Success for All Sites



Payload range (Kg):

Correlation Between Payload and Success for All Sites

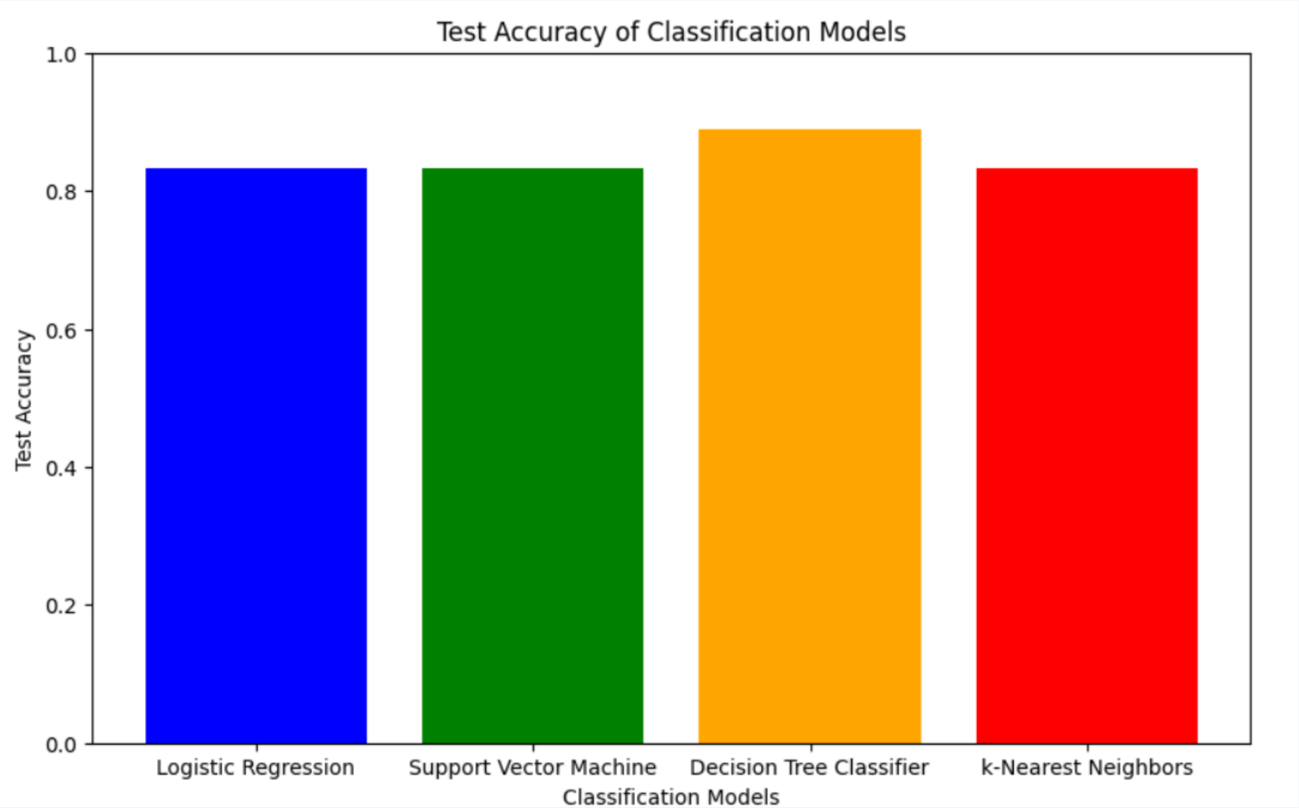
Explanation:

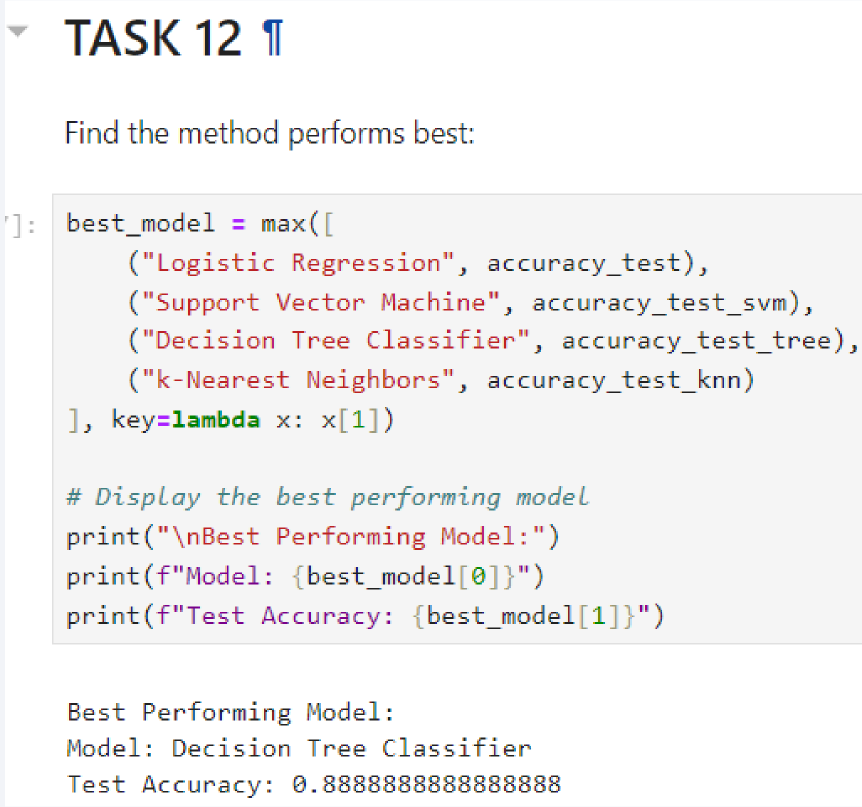
- The charts show that payloads between 2000 and 5500 kg have the highest success rate.

Section 5

Predictive Analysis (Classification)

Classification Accuracy





TASK 12 ¶

Find the method performs best:

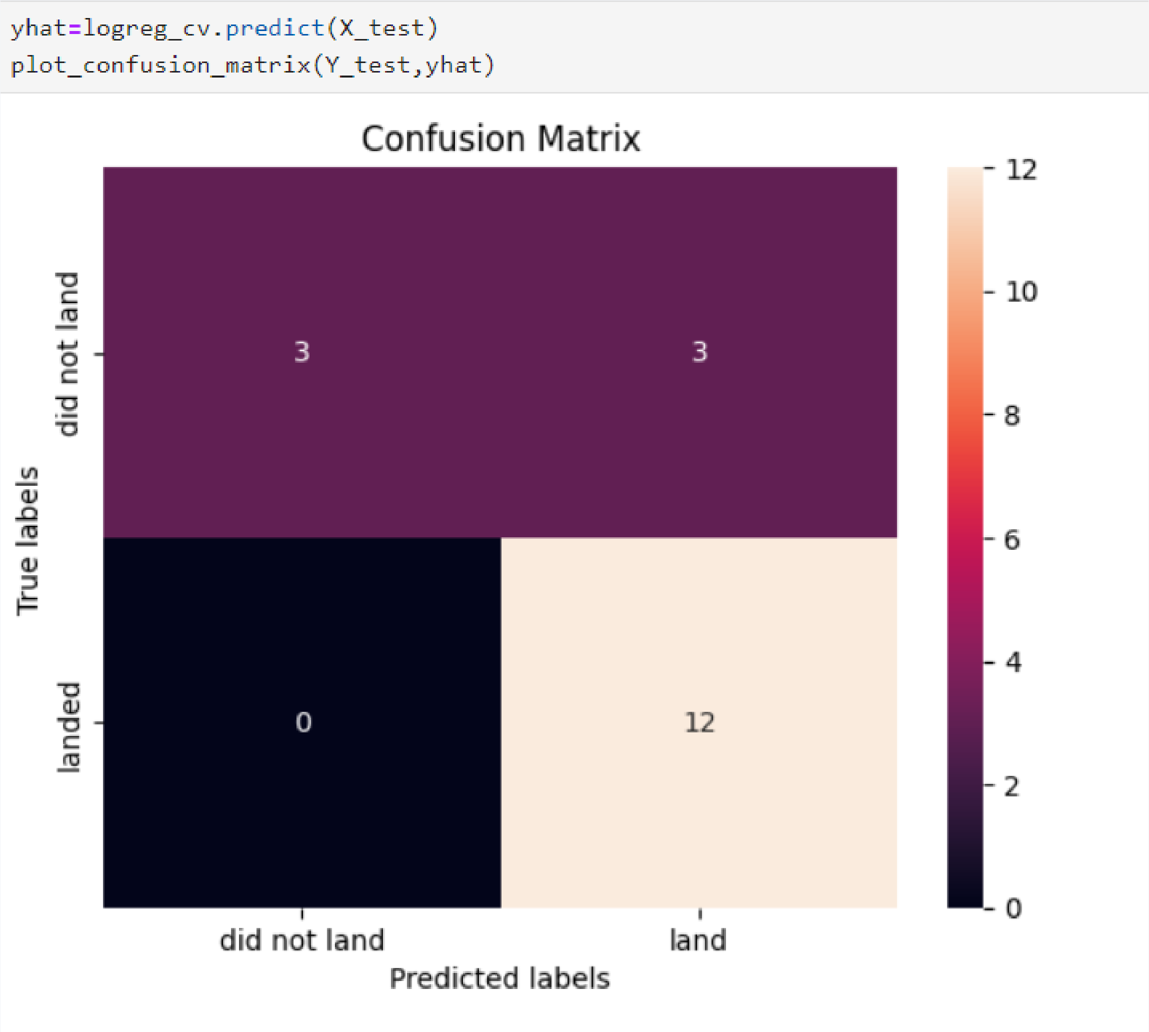
```
best_model = max([
    ("Logistic Regression", accuracy_test),
    ("Support Vector Machine", accuracy_test_svm),
    ("Decision Tree Classifier", accuracy_test_tree),
    ("k-Nearest Neighbors", accuracy_test_knn)
], key=lambda x: x[1])

# Display the best performing model
print("\nBest Performing Model:")
print(f"Model: {best_model[0]}")
print(f"Test Accuracy: {best_model[1]}")
```

```
Best Performing Model:
Model: Decision Tree Classifier
Test Accuracy: 0.8888888888888888
```

The scores of the whole Dataset confirm that the best model is the **Decision Tree Model**.

Confusion Matrix

Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the major problem is false positives



```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

Conclusions

- Decision Tree Model is the best algorithm for this dataset.
- Launches with a low payload mass show better results than launches with a larger payload mass.
- Most of launch sites are in proximity to the Equator line and all the sites are in very close proximity to the coast.
- The success rate of launches increases over the years.
- KSC LC-39A has the highest success rate of the launches from all the sites.
- Orbits ES-L1, GEO, HEO and SSO have 100% success rate.

Appendix

Special Thanks to:

- Instructors
- Coursera
- IBM

Thank you!