IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Shahnaj Ullah

February 18, 2023

Outline

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

Executive Summary

- Data was collected through SpaceX REST API and web scraping from a Wiki Page table. To clean data we filtered for records specific to Falcon 9 launch records, we replaced missing payload records with its mean value. Initially the data was missing interpretable data as it had ID numbers, we retrieved records that are associated to those ID. We classified landing outcomes with binary numbers to represent successful/failed landing. We carried out analysis using SQL and data visualization. We used Folium and Plotly dash to gain interactive analytic insights. Lastly we used machine learning models to create a model that can predict the possibility of the first stage successfully landing for falcon 9 ship.
- Our exploratory data analysis showed launches made from launch site KSC LC-39A had higher success rate. Booster version FT also had higher success rate along with payload masses ranging from 2000-3250kg had increased success odds.

Introduction

- Companies are trying to make travelling to space accessible by making it affordable. If a launch is successfully executed, SpaceX is able to reuse its first stage components of Falcon 9 thereby making it significantly cheaper per launch in comparison to its competitors. The competitors can cost upwards of 165 million dollars for a launch whereas SpaceX is upwards of 62 million dollars.
- Here, we would like to create a predictive model that can classify the landing outcome of a launch. We will use SpaceX's falcon 9 historical data to train the model. With this info we can estimate the cost of needed for the next launch knowing whether we can reuse the first stage from the launch.

Section 1

Methodology

Methodology

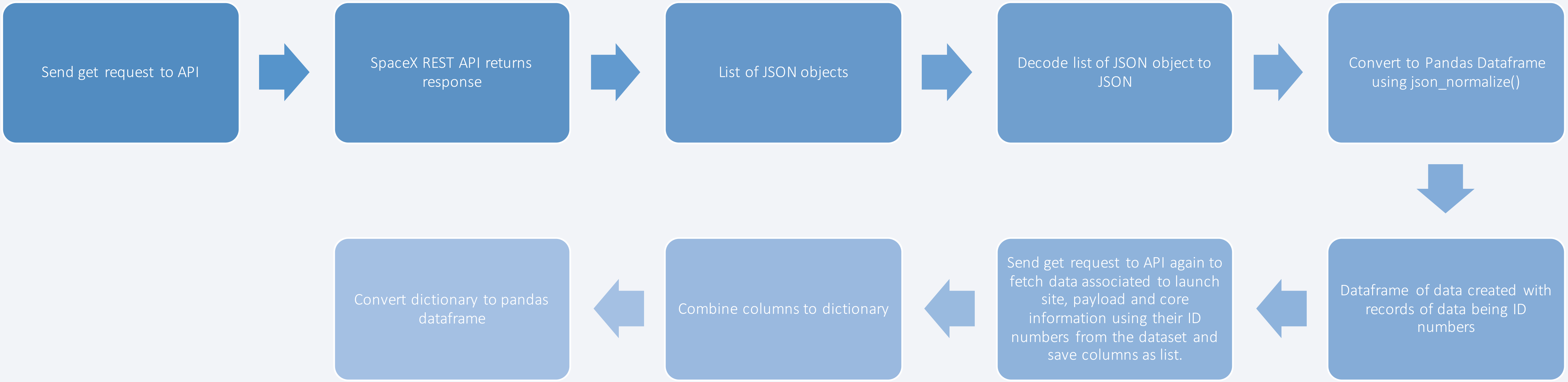
Executive Summary

- Data collection methodology:
  - SpaceX REST API was used to gather data on rocket used, payload delivered, launch specifications, landing specifications and landing outcome for SpaceX's launches.
  - Additionally, web scraping was used to gather more data on the Falcon 9 launches from a Wiki page's table. This web page contained useful data such as launch sites, flight number, payload mass, booster version and more.
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection – SpaceX API

- A **get request** was sent to a URL to target a specific endpoint of SpaceX's **REST API** which returns the data of past launches in the form of a **list of JSON objects**. The JSON objects was then converted to a dataframe using the **json_normalize** function.
- This led to obtaining a dataframe who's records consisted of data represented by ID numbers in place of useful information. Several functions were created to send multiple get requests to the API again to extract actual data on the BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude and Latitude using the ID numbers from the rocket, payload, launchpad and cores dataset. These data were saved as a list and then saved as a dictionary to be converted to a pandas dataframe.
- See process flowchart on slide

Data Collection – SpaceX API



Send get request to API

SpaceX REST API returns response

List of JSON objects

Decode list of JSON object to JSON

Convert to Pandas Dataframe using json_normalize()

Dataframe of data created with records of data being ID numbers

Send get request to API again to fetch data associated to launch site, payload and core information using their ID numbers from the dataset and save columns as list.

Combine columns to dictionary

Convert dictionary to pandas dataframe

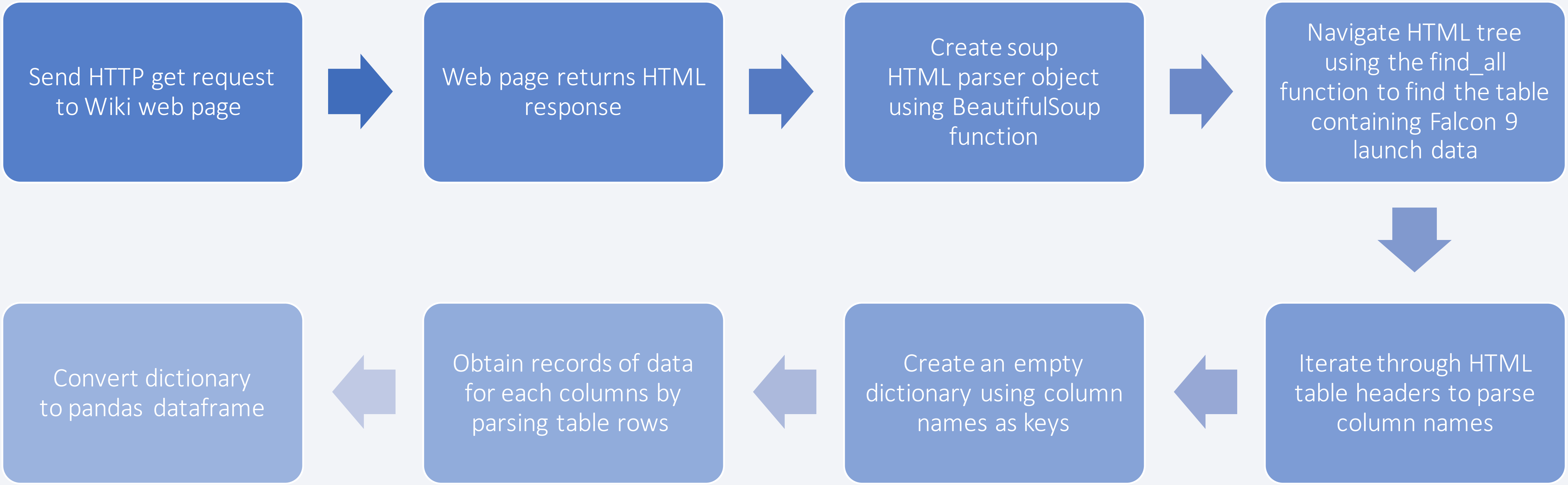
Github API notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/208a76df5c8bea68ee2397acaf0161e5fb3c2278/Data%20Collection%20API.ipynb

Data Collection – Scraping

- Python's **BeautifulSoup** package was used to **web scrape** an HTML table from a Wiki page containing records associated to Falcon 9 launches.
- The data was obtained by sending an **HTTP request** to the Wiki page's URL which returns a **response** with the **HTML contents** of the web page. Then a **beautiful soup object** was created to create an **HTML parser** which nested the HTML data to make it more navigable. The **HTML parser** was used to extract the relevant column headers and records using the find_all function on the table's HTML tag to save the each column as a dictionary. Then the dictionary was converted to a pandas data frame.
- See process flowchart on slide

Data Collection - Scraping



Send HTTP get request to Wiki web page

Web page returns HTML response

Create soup HTML parser object using BeautifulSoup function

Navigate HTML tree using the find_all function to find the table containing Falcon 9 launch data

Iterate through HTML table headers to parse column names

Create an empty dictionary using column names as keys

Obtain records of data for each columns by parsing table rows

Convert dictionary to pandas dataframe

Github Webscraping notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

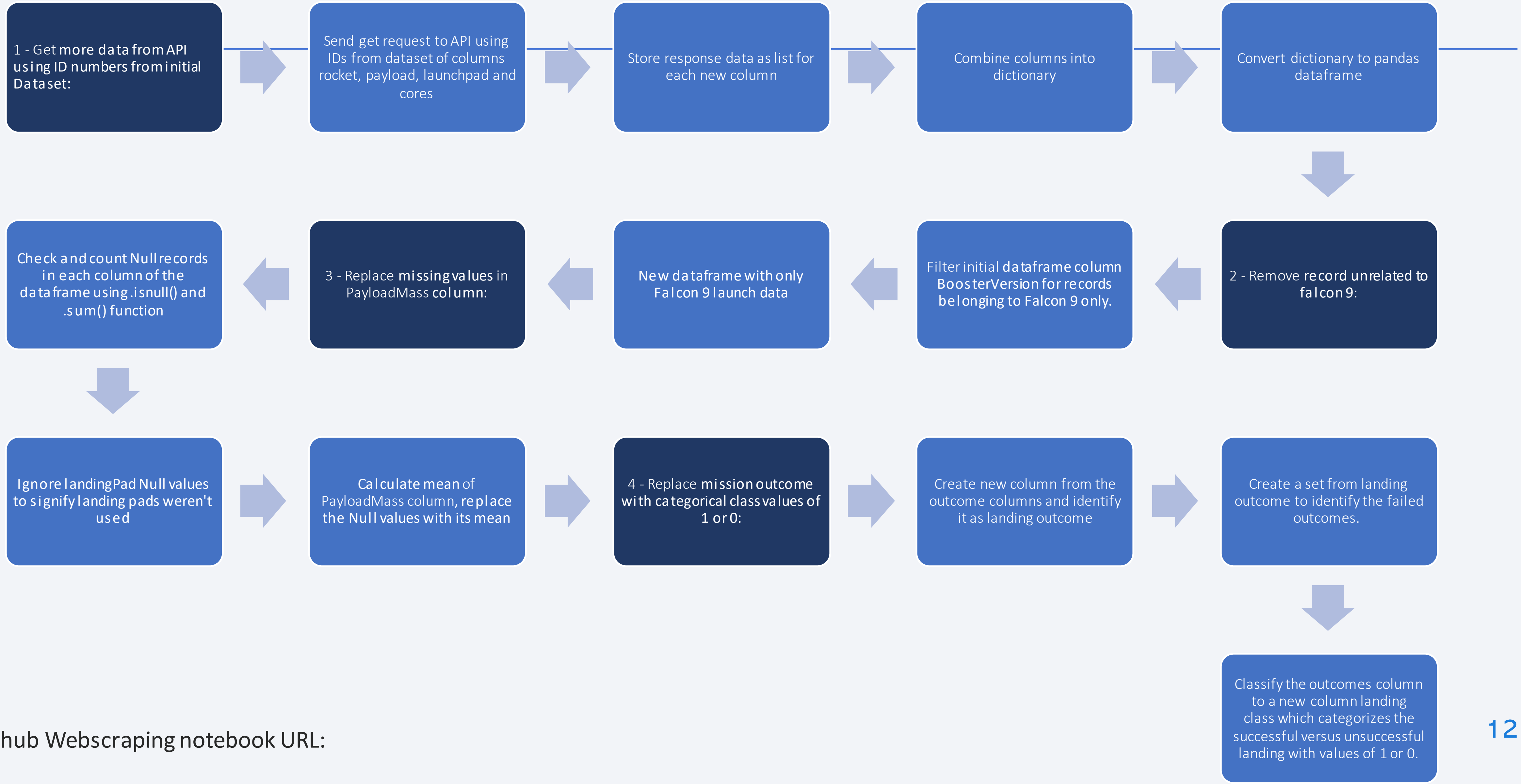
Data Wrangling

- The initial dataset from the API held records of ID numbers in the columns of rocket, payloads, launchpad and cores. These IDs are linked to useful launch data such as BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude and Latitude. A get request was sent to a different endpoint of the API again to extract those data using the ID numbers.
- Using the data from the BoosterVersion we also filtered to remove any records not relevant to Falcon 9 from the dataframe.
- Data collected through the REST API had some missing values in the PayloadMass and LandingPad columns as well. To deal with this we substituted the PayloadMass missing values by the mean of PayloadMass values and for the LandingPad it retained values of None to signify landing pads were not used.
- Furthermore, we converted the mission outcomes to either successfully landed or unsuccessful landing of the booster using categorical classes of 1 or 0, with 1 meaning successful and 0 not. True Ocean, True RTLS and True ASDS are successful landing values and False Ocean, False RTLS, False ASDS , None ASDS and None None are unsuccessful.

- GitHub Notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/Data%20Wrangling.ipynb

Data Wrangling



1 - Get more data from API using ID numbers from initial Dataset:

Send get request to API using IDs from dataset of columns rocket, payload, launchpad and cores

Store response data as list for each new column

Combine columns into dictionary

Convert dictionary to pandas dataframe

2 - Remove record unrelated to falcon 9:

Filter initial dataframe column BoosterVersion for records belonging to Falcon 9 only.

New dataframe with only Falcon 9 launch data

3 - Replace missing values in PayloadMass column:

Check and count Null records in each column of the dataframe using .isnull() and .sum() function

Ignore landingPad Null values to signify landing pads weren't used

Calculate mean of PayloadMass column, replace the Null values with its mean

4 - Replace mission outcome with categorical class values of 1 or 0:

Create new column from the outcome columns and identify it as landing outcome

Create a set from landing outcome to identify the failed outcomes.

Classify the outcomes column to a new column landing class which categorizes the successful versus unsuccessful landing with values of 1 or 0.

Github Webscraping notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

EDA with Data Visualization

- Scatter plot: Flight Number vs. Payload Mass to see the odds of successfully landing increases with continuous launch attempts and the influence of the size of payload on the likelihood of the first stage returning.
- Scatter plot: Flight Number vs. Launch Site to visualize correlation between selecting a particular Launch Site and successfully landing.
- Scatter plot: Payload and Launch Site to visualize correlation between launching from a particular site with a specific payload mass.
- Bar chart: Visualize correlation between orbit type and successfully landing.
- Scatter plot: Visualize correlation between continuous launch i.e. Flight Number and Orbit type.
- Scatter plot: Visualize correlation between Payload size and Orbit type.
- Line plot: Visualize trends between successful landing and the year.
- Github notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/EDA%20using%20Data%20VIsualization.ipynb

EDA with SQL

1. Query list of all unique launch sites in the space mission
2. Query five records in which launch site begin with string 'CCA'
3. Query the total payload mass carried by boosters launched by NASA (CRS)
4. Query the average payload mass carried by booster versions F9 v1.1
5. Query the date in which the first successful ground pad landing outcome was achieved
6. Query the list of names of boosters that have successful landing in drone ship and a payload mass of greater than 4000kg but less than 6000kg
7. Query the count of the total number of successful and failed mission outcomes
8. Query the list of booster version names that have carried the maximum payload mass
9. Query the list of month, failed landing outcome in drone ship, booster version and launch sites for the months in the year 2015
10. Query the rank count of successful landing outcomes in descending order between the dates 04-06-2010 and 20-03-2017

- Github notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/SQLlite%20Exploratory%20Data%20Analysis.ipynb

Build an Interactive Map with Folium

1. Blue circle markers were used to identify the four mission launch sites.

2. All the launch outcomes were clustered by launch site and then further identified by green and red popup icon markers to signify successful and unsuccessful landing outcomes respectively. This can help identified if certain sites have higher success rate than others.

3. A mouse pointer object was also created to be able to hover over a point on the map and easily obtain the latitude and longitude of that location of interest. This was used to later identify the distance between the railways, highways, city and coast in a launch site's close proximity.

4. Lastly we identified the coast, city, highway and railway closest to launch site KSC LC-39A using a blue line to map the distance of these points. The distances were also calculated and marked on the map. This is helpful in understanding how sites are usually located i.e. for example far from cities or nearby coasts, etc.

- GitHub notebook URL:

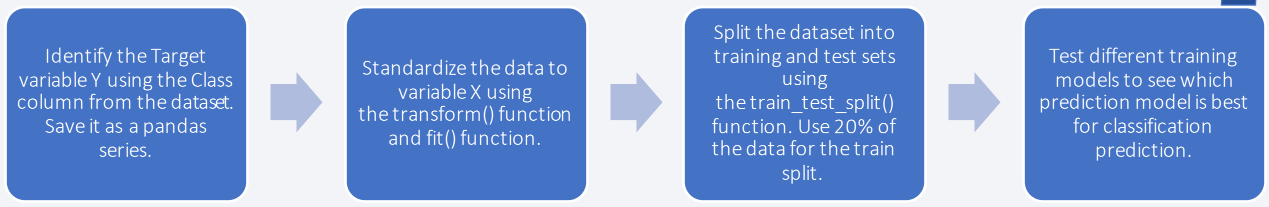
https://github.com/ShahnajRUllah/Space_Y/blob/master/Location%20site%20analysis%20with%20folium.ipynb

Build a Dashboard with Plotly Dash

- Using plotly dash we were able to gain insights with regards to which launch sites experiences the most success in its launches. A dropdown menu was created allowing users to view either the number of successful launches across all the sites or to view the ratio in the number of successful versus failed launches of a given site.
- Secondly, a scatter plot is also created to visualize the correlation between using a particular payload mass, a specific booster version and having a successful landing outcome. Using the dropdown menu we can select to gain the insight across all launch sites or a particular one. Additionally, a slider tool was added to narrow the x-axis payload mass range to get a closer look on the booster versions in the specified range.
- GitHub space app URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

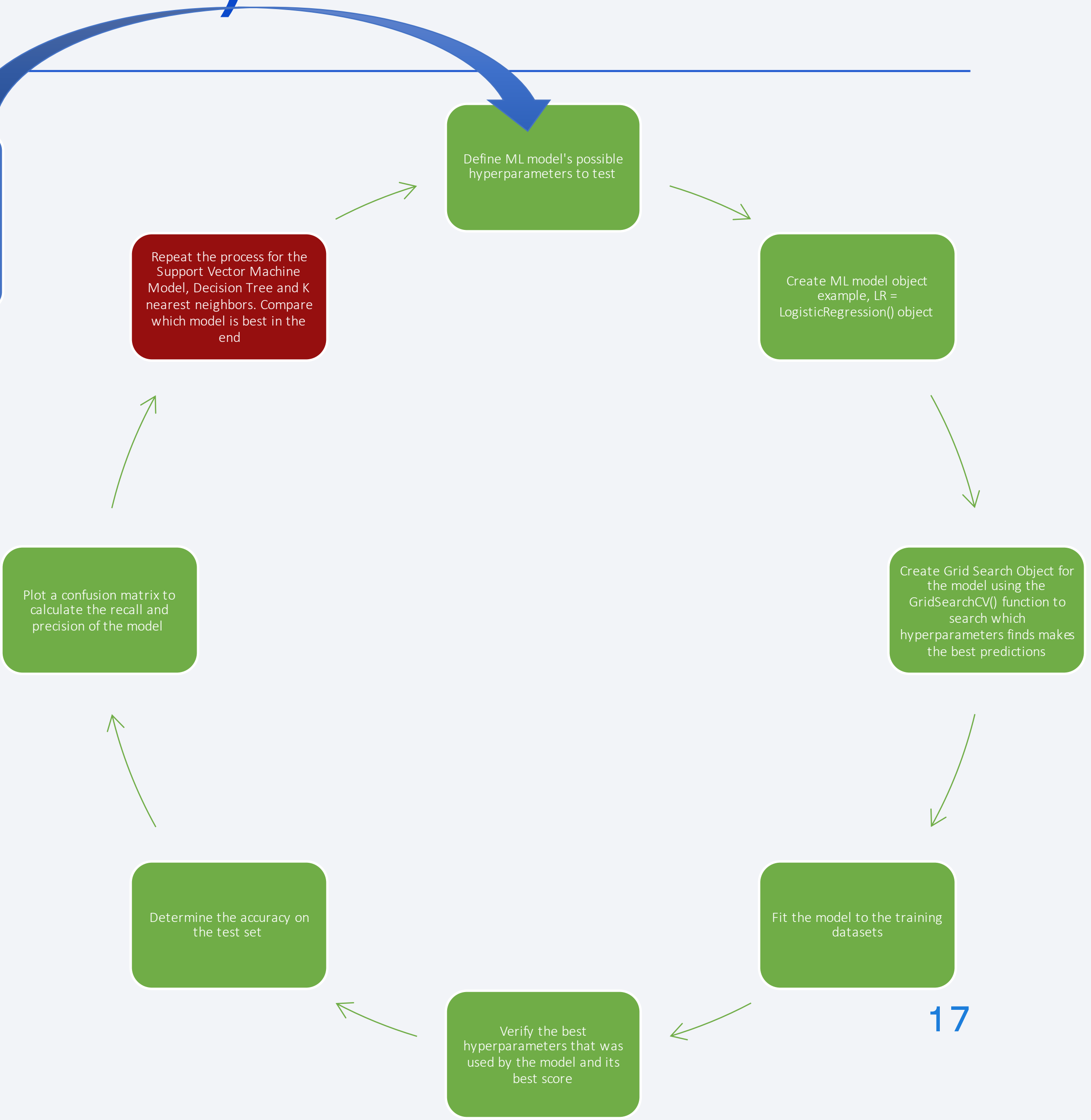


Identify the Target variable Y using the Class column from the dataset. Save it as a pandas series.

Standardize the data to variable X using the transform() function and fit() function.

Split the dataset into training and test sets using the train_test_split() function. Use 20% of the data for the train split.

Test different training models to see which prediction model is best for classification prediction.



Define ML model's possible hyperparameters to test

Create ML model object example, LR = LogisticRegression() object

Create Grid Search Object for the model using the GridSearchCV() function to search which hyperparameters finds makes the best predictions

Fit the model to the training datasets

Verify the best hyperparameters that was used by the model and its best score

Determine the accuracy on the test set

Plot a confusion matrix to calculate the recall and precision of the model

Repeat the process for the Support Vector Machine Model, Decision Tree and K nearest neighbors. Compare which model is best in the end

- First the Target and Feature variables were identified. Then the dataset was split into training and test datasets.

- After the split of the dataset, we trained four different machine learning models to see which models made the best classification predictions for the landing outcomes. The models compared to was the Support Vector machine, Decision Tree, Logistic Regression and K nearest neighbor model.

- We performed a grid Search to find out which hyperparameters is best for the models to yield the highest accuracy.

- The recall and precision was calculated from each model's respective confusion matrix. The model with the highest recall and precision deemed the best model for our prediction.

- The accuracy calculated using the best_score() function for each model was also compared in a bar chart to also determine the best model for our objective.

- GitHub notebook URL:

https://github.com/ShahnajRUllah/Space_Y/blob/master/SpaceX_ML_Prediction.ipynb

Results from Data Visualization EDA

Result Summary from Data Visualization EDA

Launch site related observation:

1. Launch sites VAFB SLC-4E and KSC LC-39A shows a higher success rate in comparison to site CCAFS SLC-40.
2. Site KSC LC-39A has higher success with lower payload masses of below 5500kg while site VAFB SLC-4E is successful with payload masses lower than 10000kg. Site CCAFS SLC-40 experiences more success with heavy payloads of approximately 16000kg.

Orbit type related observation:

1. Orbit ES-L1, GEO and SSO have higher success rates versus orbit SO which has none.
2. Orbit LEO experiences an increased success rate as the launch attempt number increases, i.e. flight number.
3. Orbit LEO and ISS have higher successful landing rate with heavier payloads.

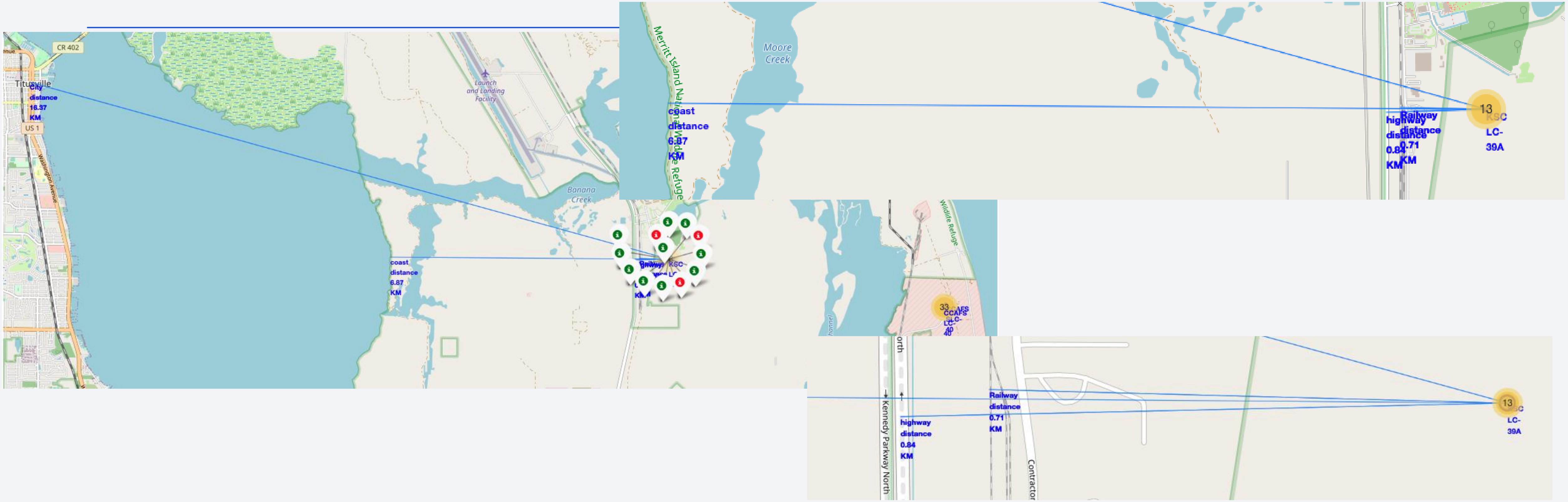
Yearly landing outcome trend shows the success rate increases after the year 2013.

Results from EDA using SQL

Result summary from EDA using SQL:

1. There are four launch sites in total. CCAFS LC-40, CCAFS SLC-40, KSC LC-39A and VAFB SLC-4E.
2. NASA (CRS) booster launched a total of 45596kg payload.
3. Booster version f9 v1.1 carried an average of 2534.67kg
4. First successful landing outcome in ground pad was on 22-12-2015
5. Boosters FT B1022, B1026, B1021.2 and B1031.2 have had successful landing on drone ships while carrying payloads between 4000-6000kg.
6. A total of 100 successful mission outcome was counted and 1 failed in flight
7. There were 12 B5 booster versions that carried the maximum payload.
8. In January and April 2015, there has been two failed drone ship landing outcomes both were launched from CCAFS LC-40.
9. There has been 10 landing outcomes with no attempts and 5 successful/5 failed drone ship landing as well. 3 successful ground pas landing, 3 controlled/2 uncontrolled ocean landing. 2 parachute failed landing and 1 precluded drone ship landing.

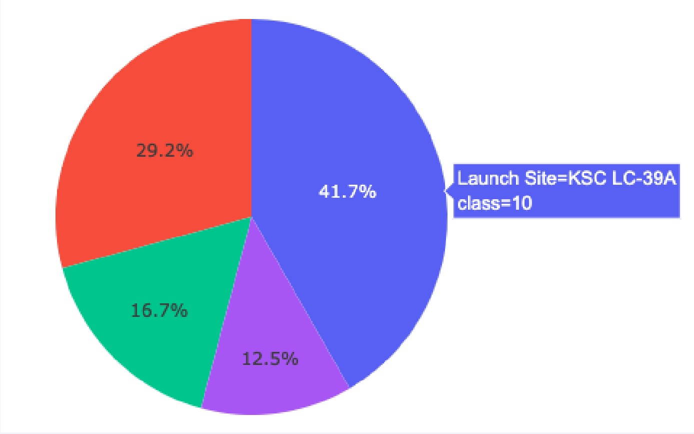
Results from Folium Interactive Map Analysis

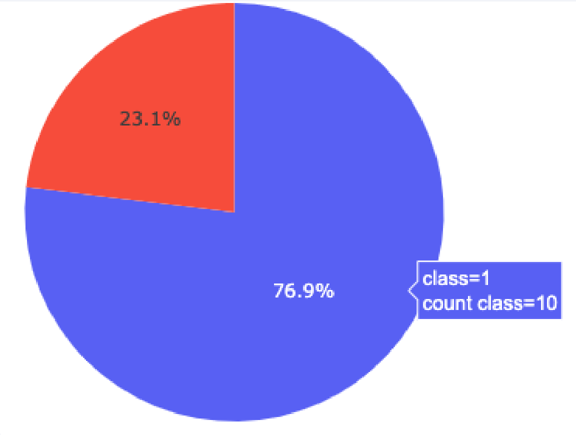


From the interactive folium map we've highlighted the most popular launch site, i.e. KSC LC-39A. We can see the closest city is 16.37km away. The coast is 6.87km, and there is a highway and railway 0.84km and 0.71km away respectively.

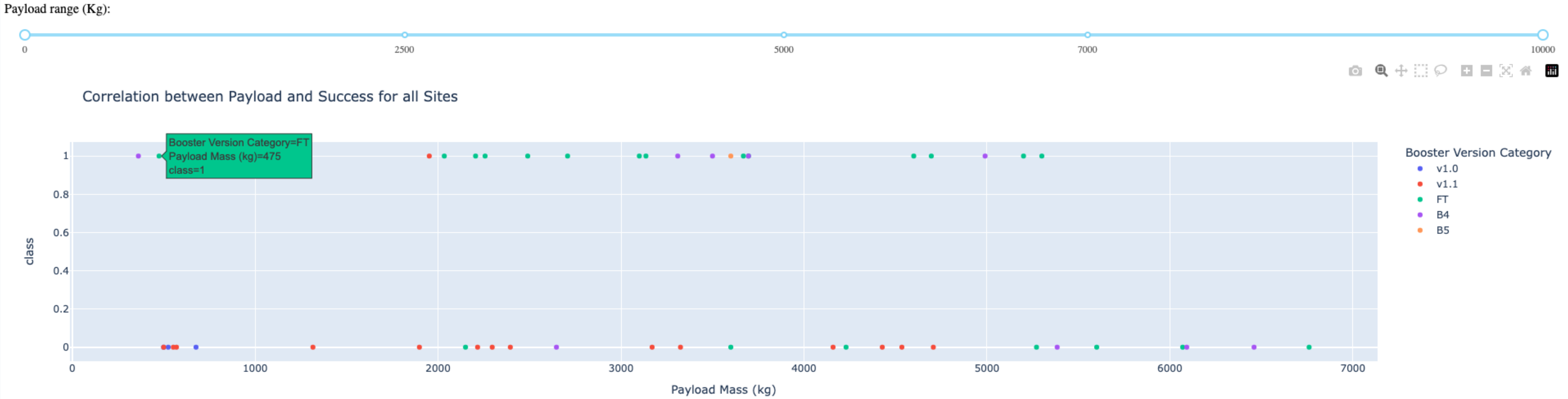
Looking at all launch sites, the city tends to be further away from the site while they are all close to a coastline, a highway and railway.

Results from Plotly Dash





- Dash pie chart shows Launch site KSC LC-39A has the most success rate.
- Dash scatter plot shows booster version FT has the most success and v1.1 has more fails. Payload masses between 1900-3700kg have higher success rate as well.



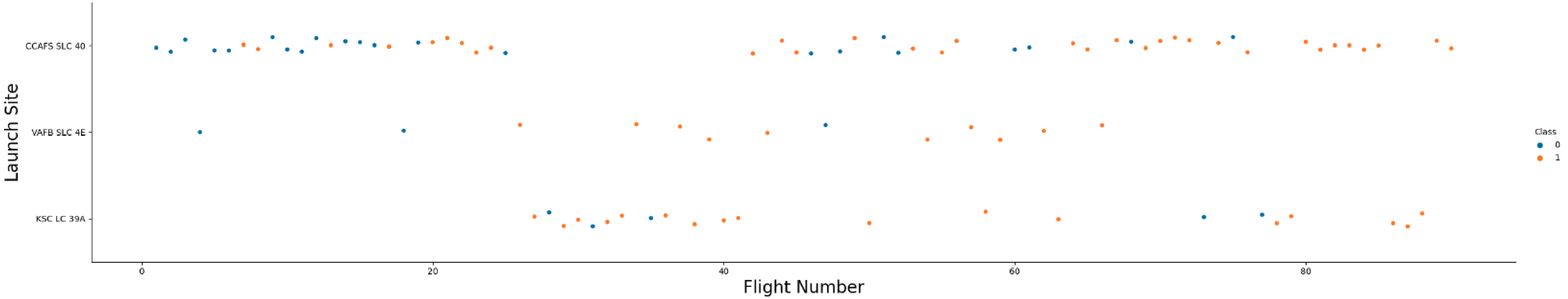
Results from Predictive Analysis

- From our predictive analysis process we concluded the best model to make our classification prediction of whether a launch will successfully land or not is the Decision tree model with the hyperparameters set to the following:
    - criterion:entropy,
    - max_depth:8,
    - max_features:sqrt,
    - min_samples_leaf:2,
    - min_samples_split:2,
    - splitter:random.
- This model exhibits a 89% accuracy.

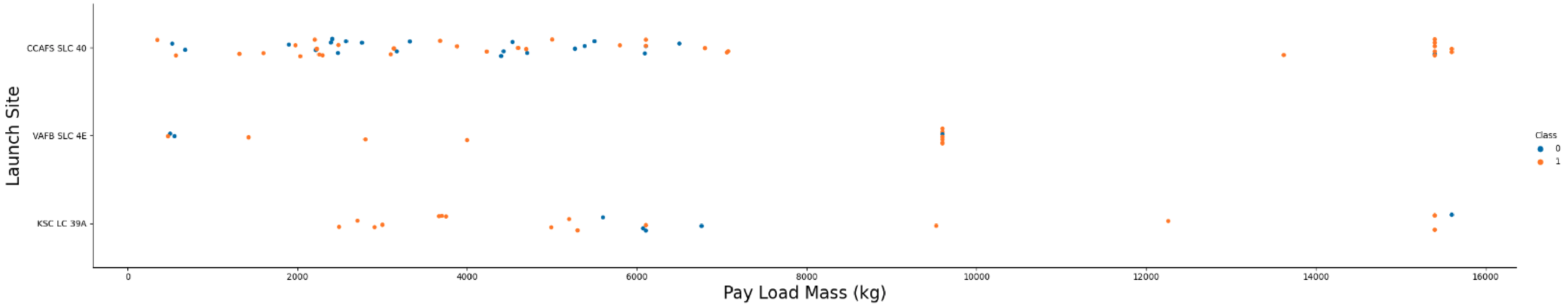
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

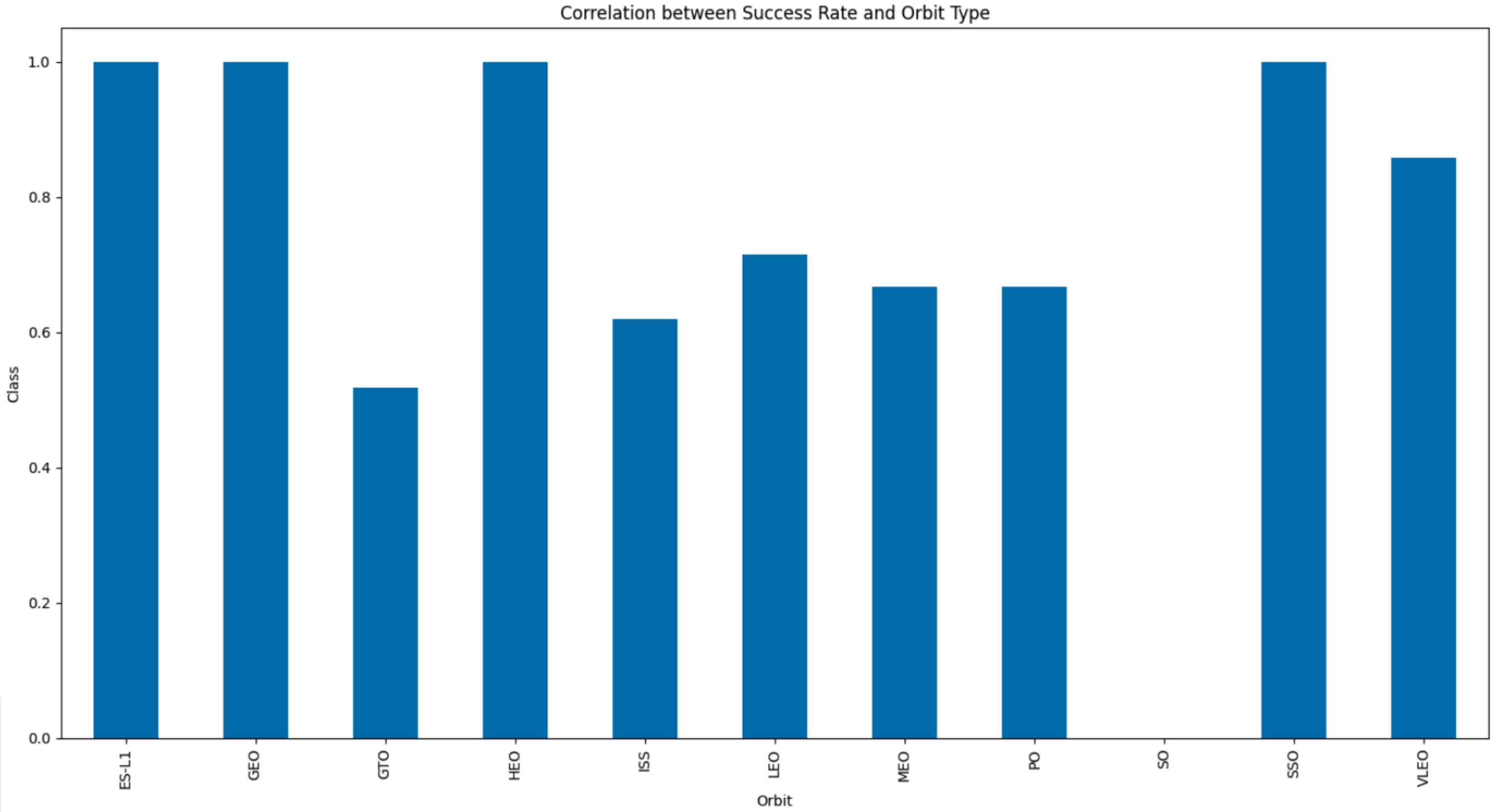


- CCAFS SLC-40: 33/55 successful launches ~ 60% success rate
- VAFB SLC-4E: 10/13 successful launches ~ 77% success rate
- KSC LC-39A: 17/22 successful launches ~77% success rate
- Launch sites VAFB SLC 4E and KSC LC-39A have a success rate of 77% in comparison to CCAFS SLC-40 which has a 60% success rate for landing successfully.



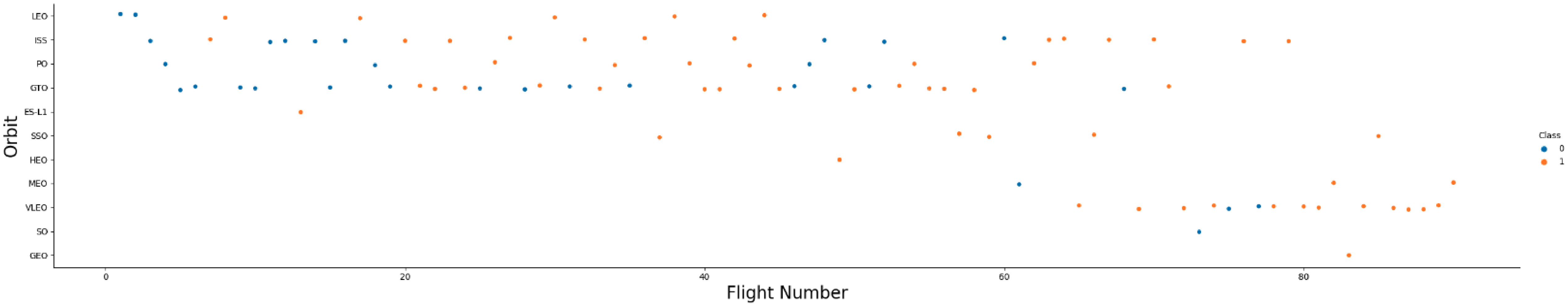
Payload vs. Launch Site

- We can see less attempts have been made with payload mass of over 10000kg on sites KSC LC-39A and CCAFS SLC-40. Site VAFB SLC 4E had no launch attempts with payload masses of above 10000kg.
- Site KSC LC-39A has more successful launches when using payload masses below 5500kg.
- Site VAFB SLC 4E has more successful launches when using heavier payloads above 1000kg and below 10000kg.
- Site CCAFS SLC-40 exhibits more successful launches when using heavier payload masses of between 6500kg and 16000kg.



Success Rate vs. Orbit Type

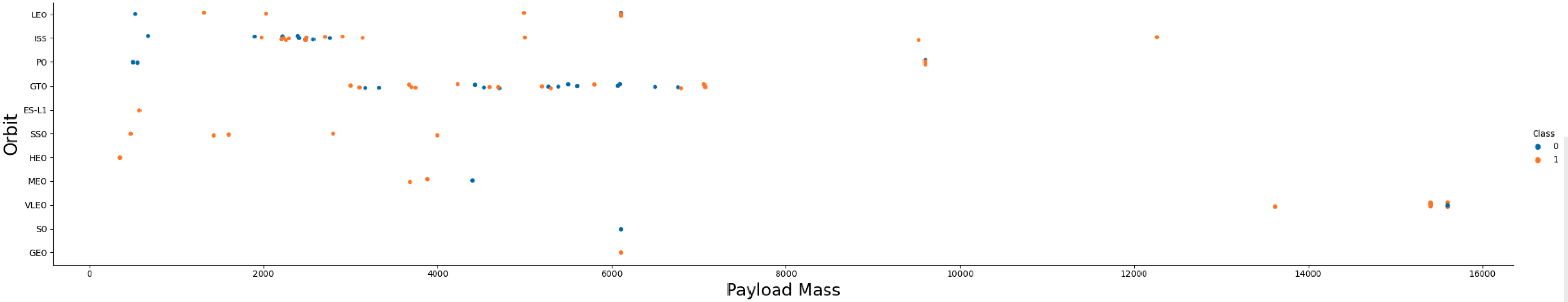
- Orbit ES-L1, GEO, HEO and SSO exhibit higher success rates while orbit SO exhibits no success.



Flight Number vs. Orbit Type

- We can see after the initial first two launches, orbit LEO's success is related to number of flights. With more launch attempts, there is increased success.
- When it comes to orbits GTO, ISS and PO, no correlation is seen between the launch attempt flight number.

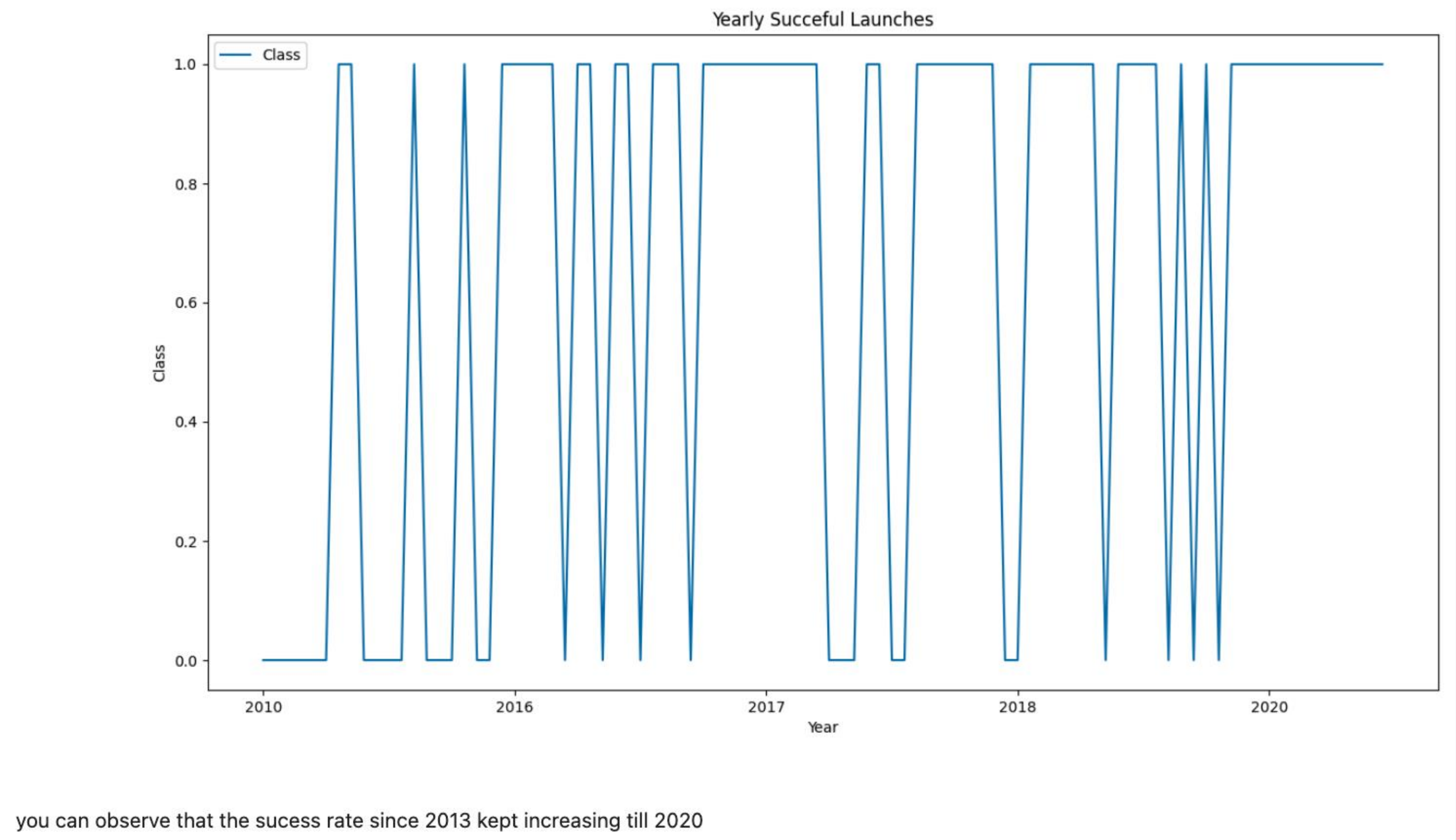
Payload vs. Orbit Type



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

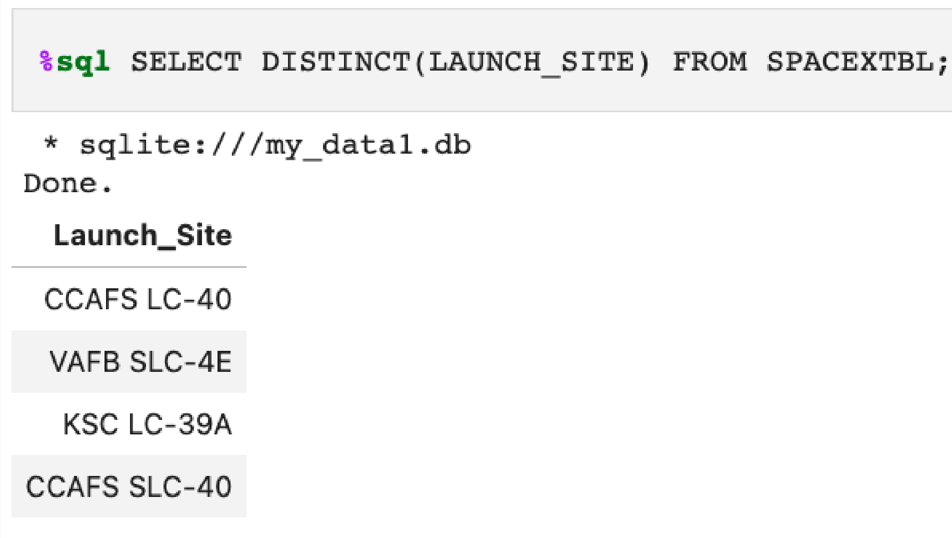
However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names



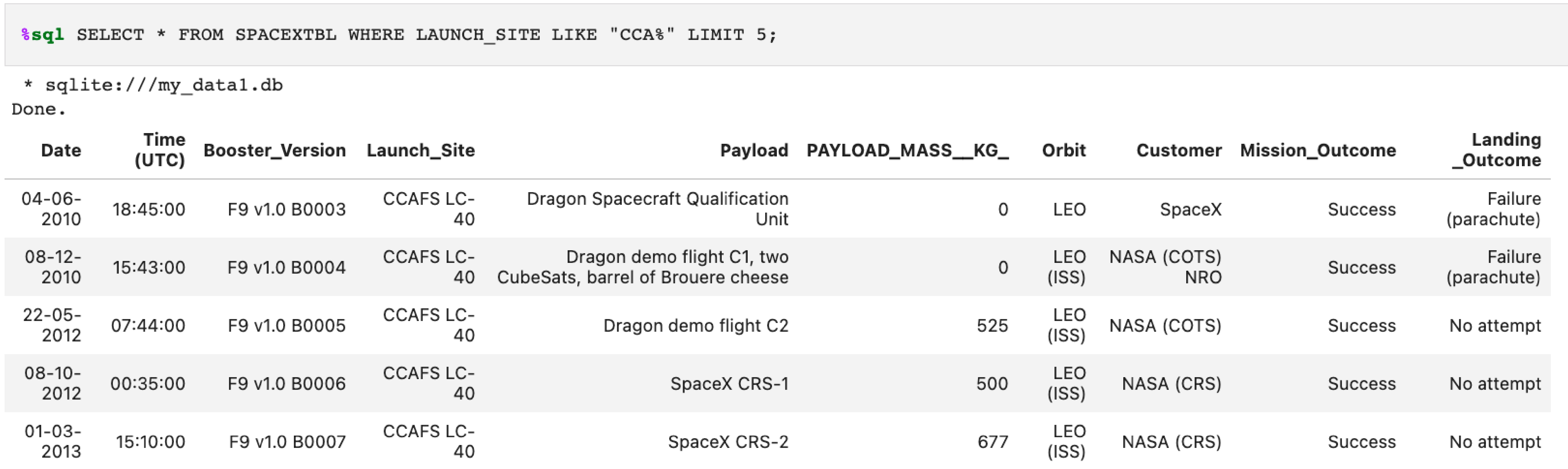
```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL;
```

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

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

- The result of the query shows all the unique space launch sites for falcon 9. There are four sites.

Launch Site Names Begin with 'CCA'



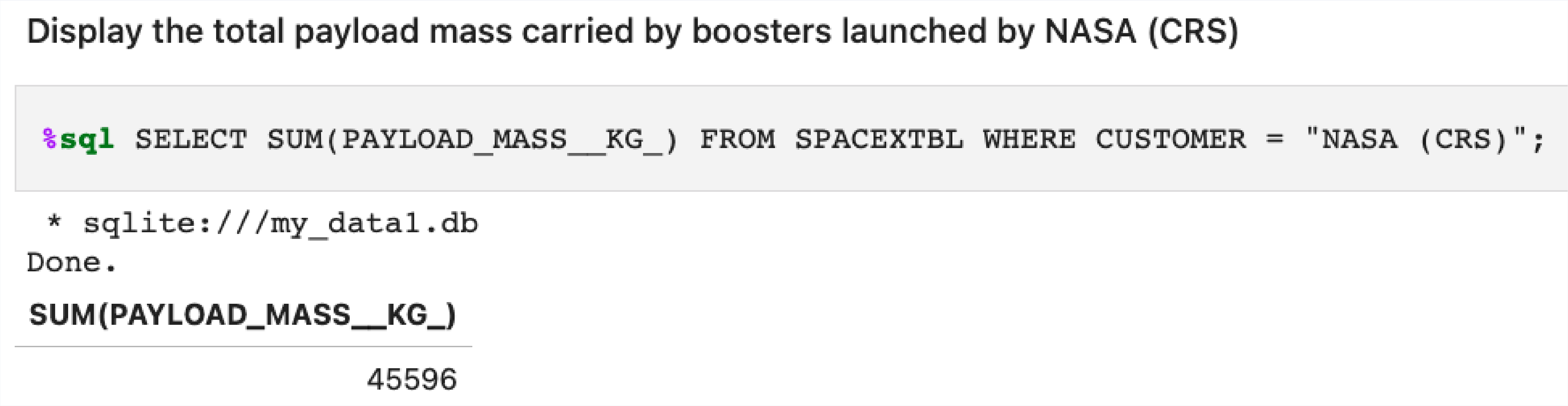
```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE "CCA%" LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- The query here lists the first five records from the data in which the name of the launch sites begins with "CCA". Our earlier visualization EDA had excluded launch site CCAFS LC-40 which this table reveals.

Total Payload Mass



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

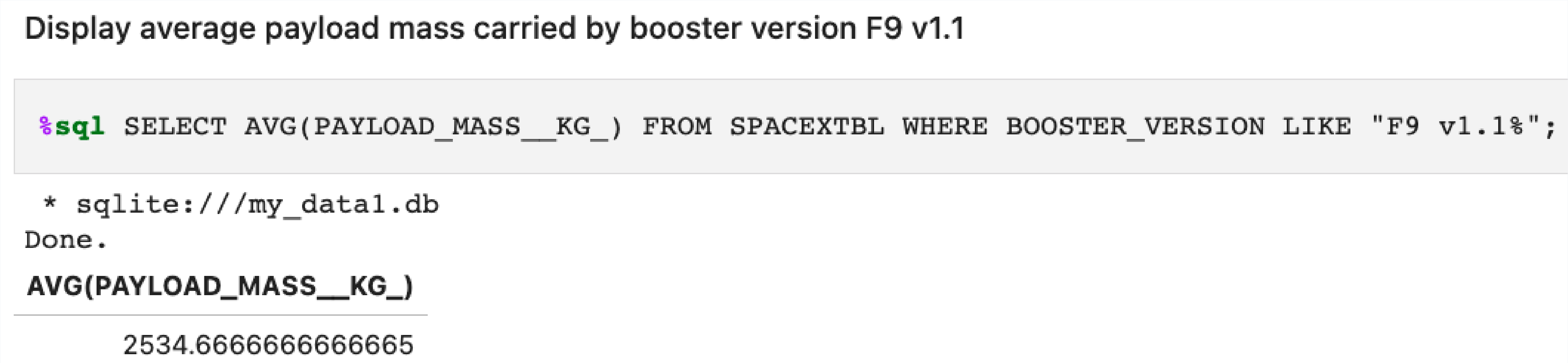
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = "NASA (CRS)";
```

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

SUM(PAYLOAD_MASS__KG_)
45596

- The query here shows the total payload mass carried by boosters carried launched by NASA (CRS).

Average Payload Mass by F9 v1.1



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

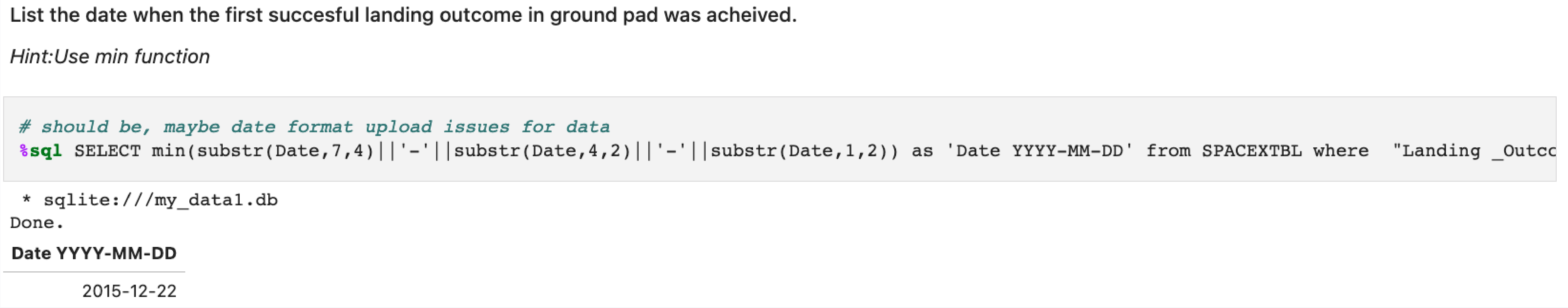
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION LIKE "F9 v1.1%";
```

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

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

- This query shows the booster version F9 v1.1 carried an average mass of 2534.67kg payload.

First Successful Ground Landing Date



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

Hint:Use min function

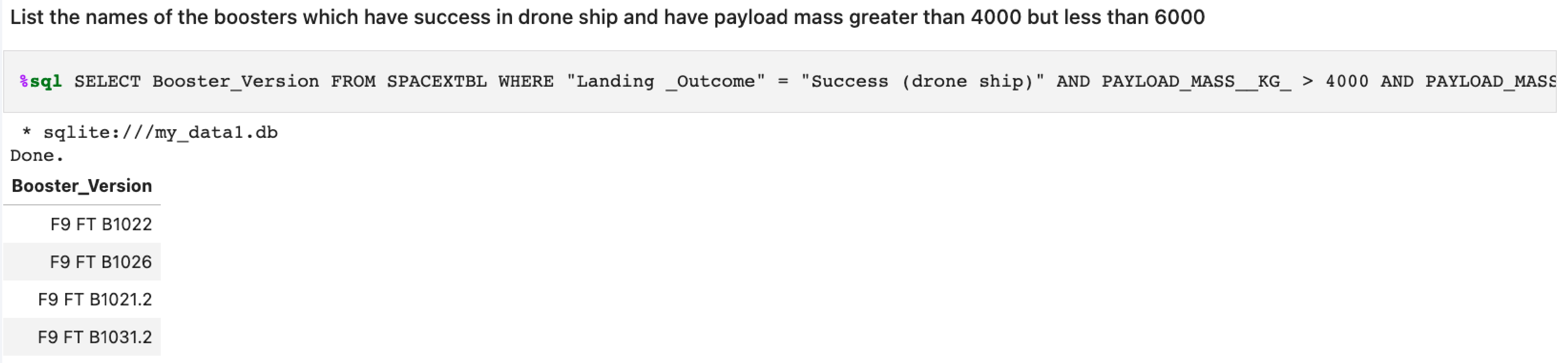
```
# should be, maybe date format upload issues for data
%sql SELECT min(substr(Date,7,4)||'-'||substr(Date,4,2)||'-'||substr(Date,1,2)) as 'Date YYYY-MM-DD' from SPACEXTBL where  "Landing _Outcc
```

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

Date YYYY-MM-DD
2015-12-22

- This query shows the first successful landing outcome in ground pad was achieved on December 22nd, 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000



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

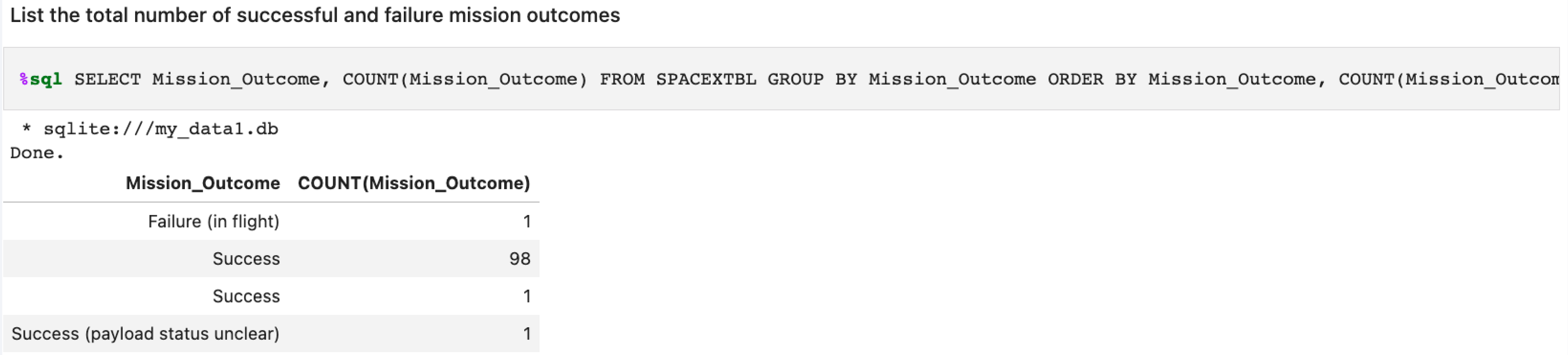
```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE "Landing _Outcome" = "Success (drone ship)" AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS
```

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

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

- This query shows the list of boosters who have had successful landing outcomes on drone ships while carrying payloads of more than 4000kg and less than 6000kg.

Total Number of Successful and Failure Mission Outcomes



List the total number of successful and failure mission outcomes

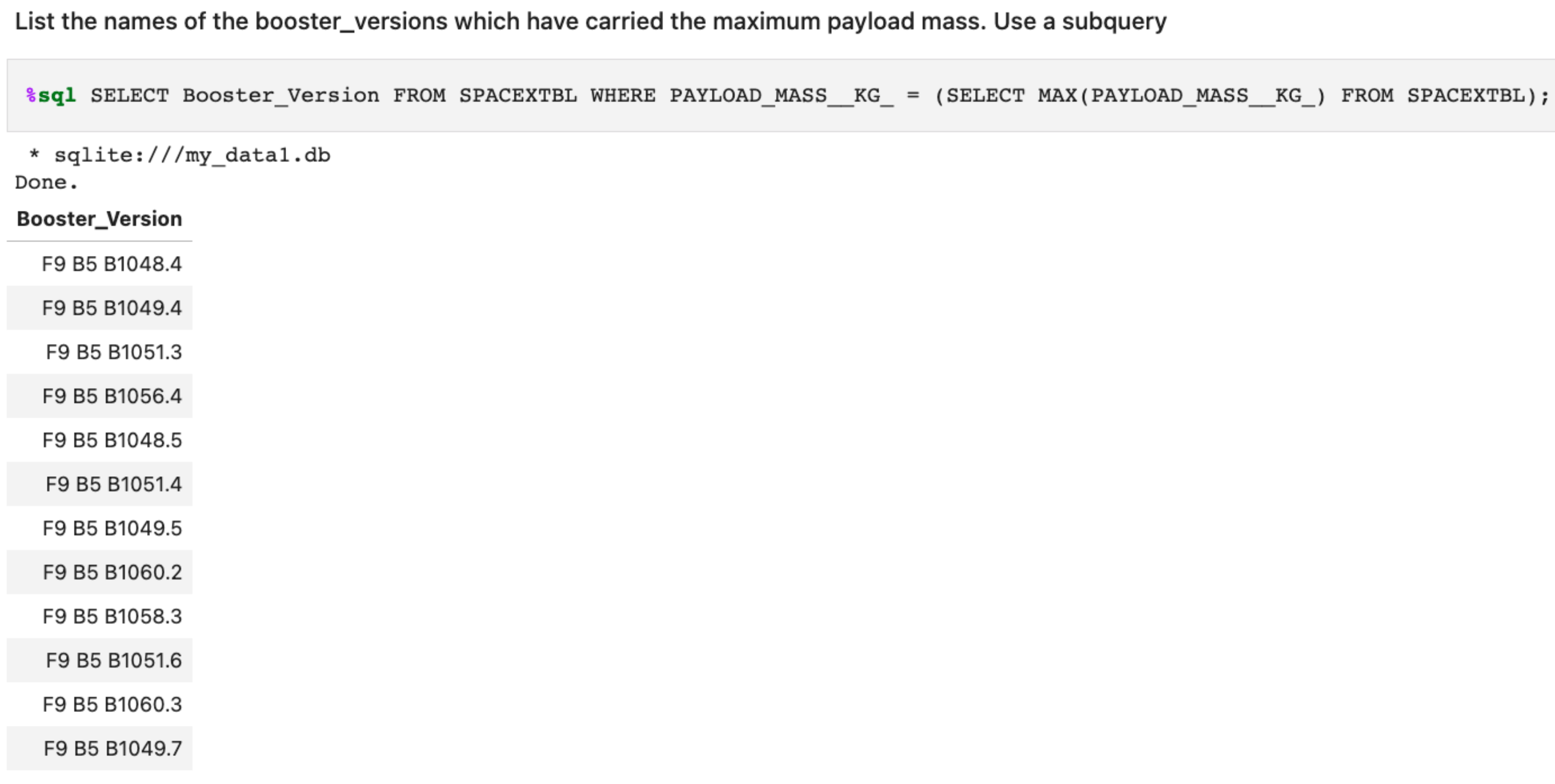
```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) FROM SPACEXTBL GROUP BY Mission_Outcome ORDER BY Mission_Outcome, COUNT(Mission_Outcom
```

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

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

- This query summarizes the total number of mission outcomes, i.e. the total number of successful outcomes and failed outcomes.

Boosters Carried Maximum Payload



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

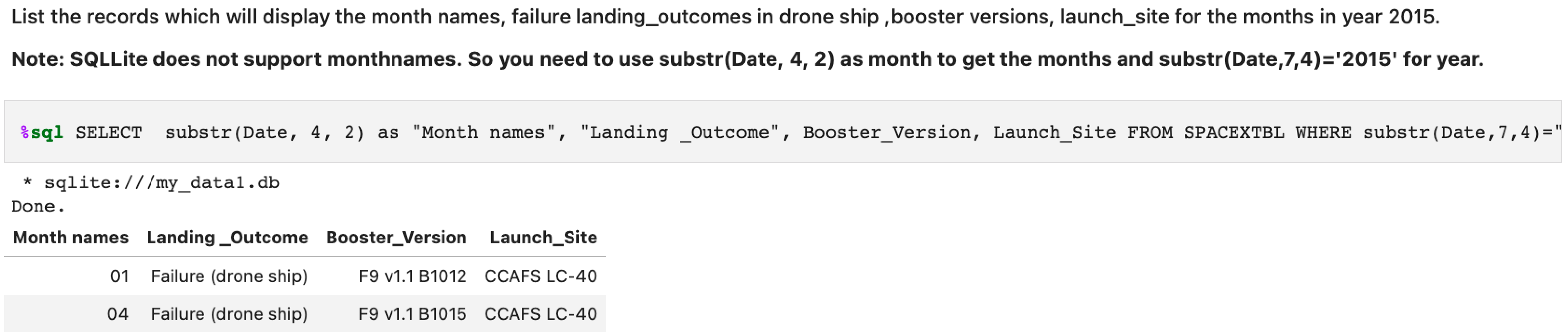
```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

```
 * 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

- This query shows the list of booster versions which carried the maximum payload.

2015 Launch Records



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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

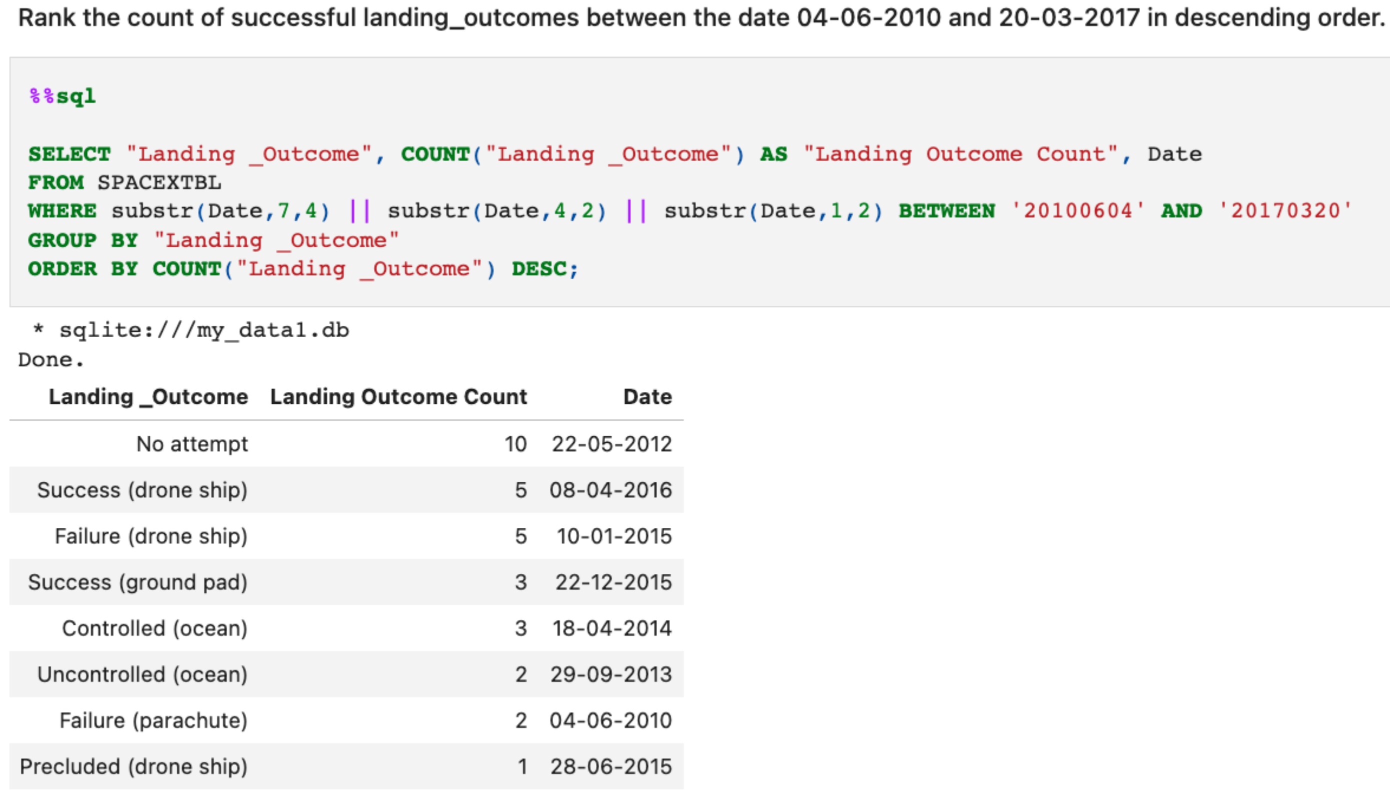
```
%sql SELECT  substr(Date, 4, 2) as "Month names", "Landing _Outcome", Booster_Version, Launch_Site FROM SPACEXTBL WHERE substr(Date,7,4)="
```

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

Month names	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

- Here we summarized the launch records who've had failed landing outcomes on drone ships for the year 2015. The details include the month of the launch, the booster version used and the launch site.

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



Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%%sql

SELECT "Landing _Outcome", COUNT("Landing _Outcome") AS "Landing Outcome Count", Date
FROM SPACEXTBL
WHERE substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2) BETWEEN '20100604' AND '20170320'
GROUP BY "Landing _Outcome"
ORDER BY COUNT("Landing _Outcome") DESC;
```

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

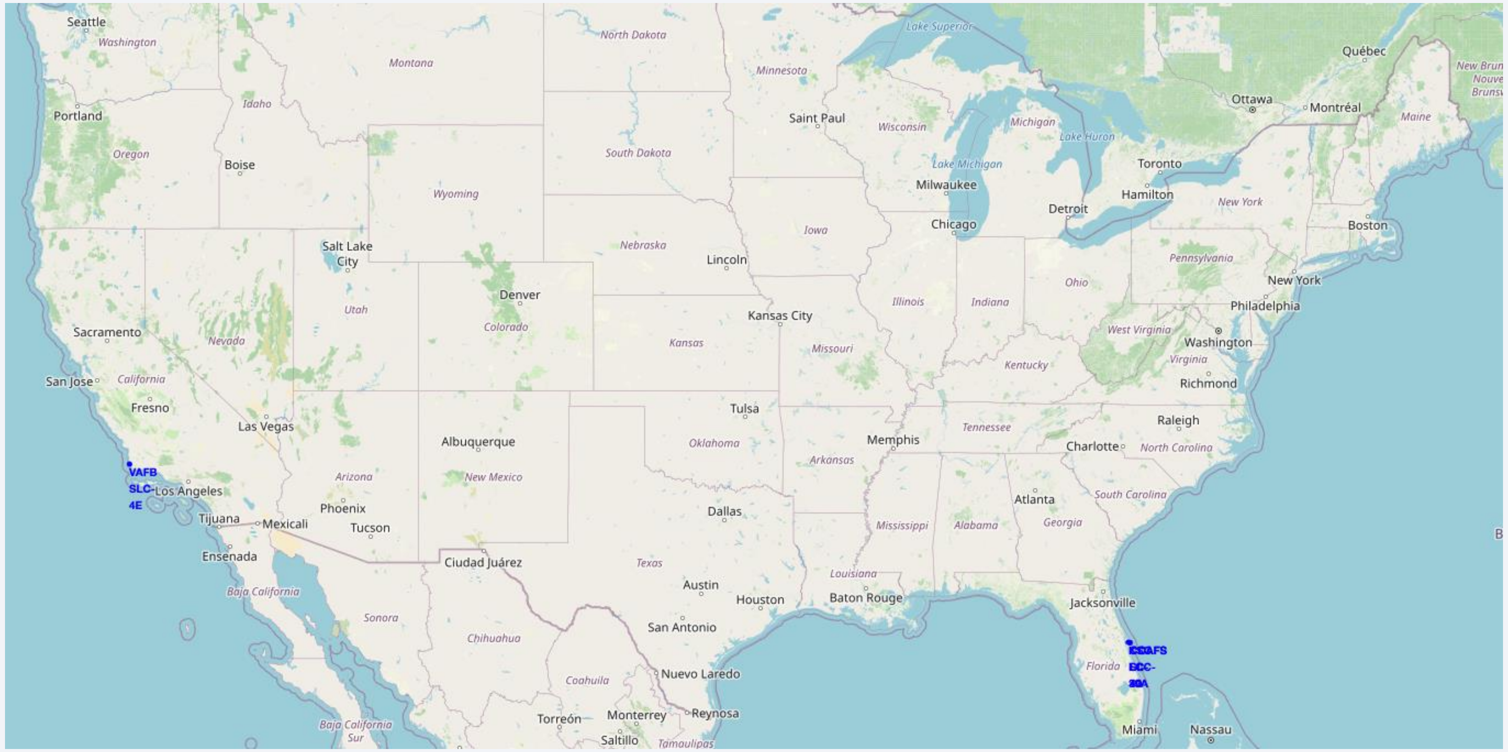
Landing _Outcome	Landing Outcome Count	Date
No attempt	10	22-05-2012
Success (drone ship)	5	08-04-2016
Failure (drone ship)	5	10-01-2015
Success (ground pad)	3	22-12-2015
Controlled (ocean)	3	18-04-2014
Uncontrolled (ocean)	2	29-09-2013
Failure (parachute)	2	04-06-2010
Precluded (drone ship)	1	28-06-2015

- This query counts and groups the landing outcomes and then ranks them in descending order for the time period June 4, 2010 to March 20, 2017.

Section 3

Launch Sites Proximities Analysis

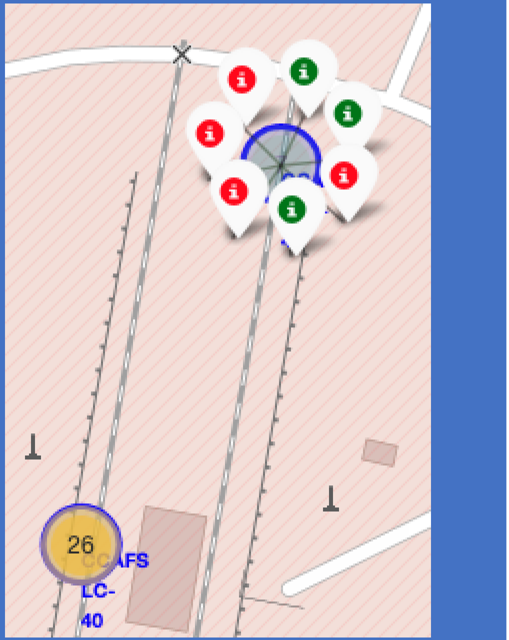
Mapped Location of all Space Mission Launch Sites



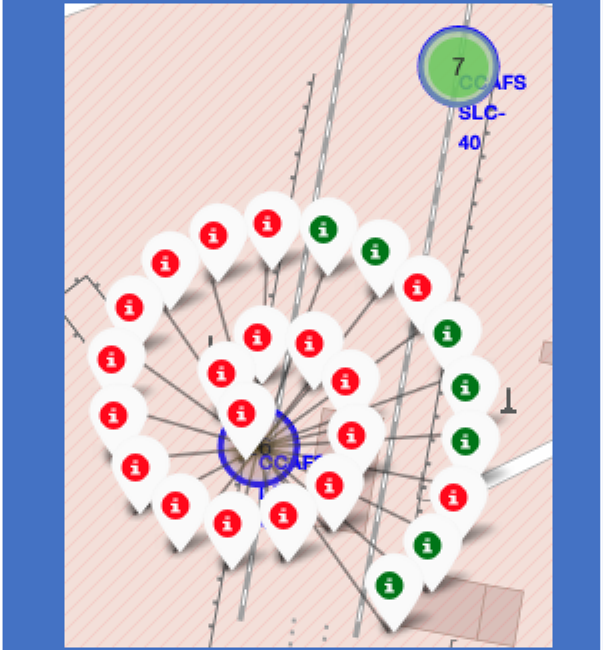
- The four mission launch sites (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A and VAFB SLC-4E) are marked in blue dots. We can see that all sites are situated very close to a coast and none of them are near the equator
- Explain the important elements and findings on the screenshot

Successful and Failed Launch Outcomes Mapped in Clusters

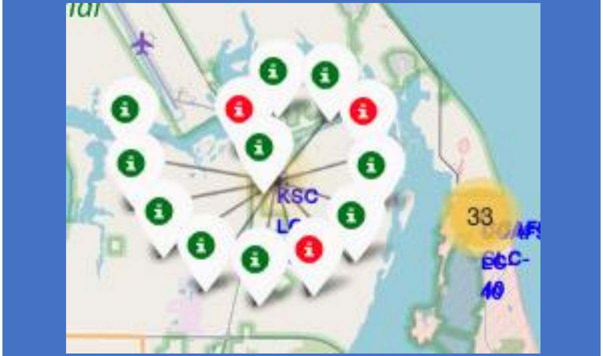
CCAFS SLC-40



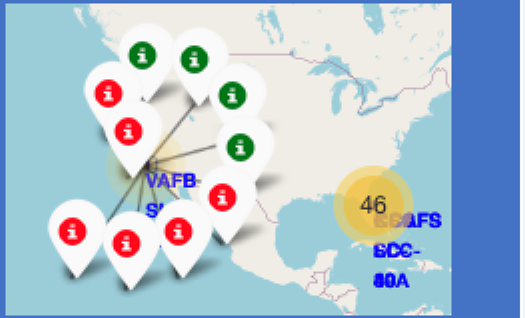
CCAFS LC-40



KSC LC-39A

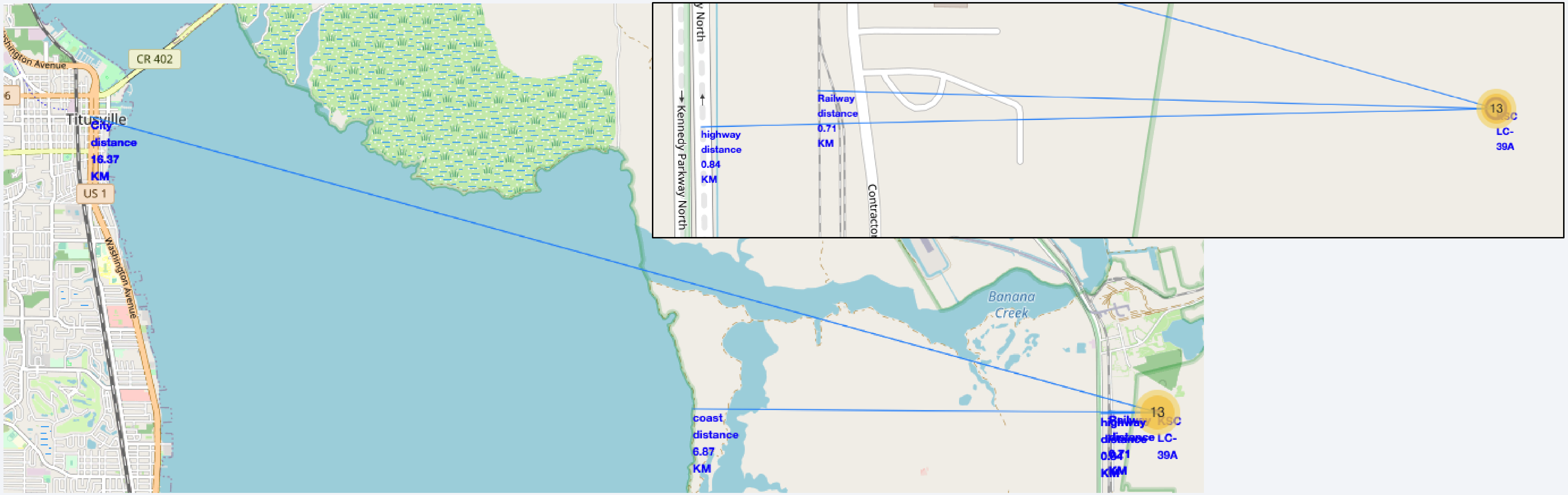


VAFB SLC-4E



- The map clustered all the launch outcomes by launch site and identified the failed outcomes in red and the successful outcomes in green.
- We can see there has been a lot of launches from site CCAFS LC-40 but launch site KSC LC-39A shows the most successful launch outcomes.

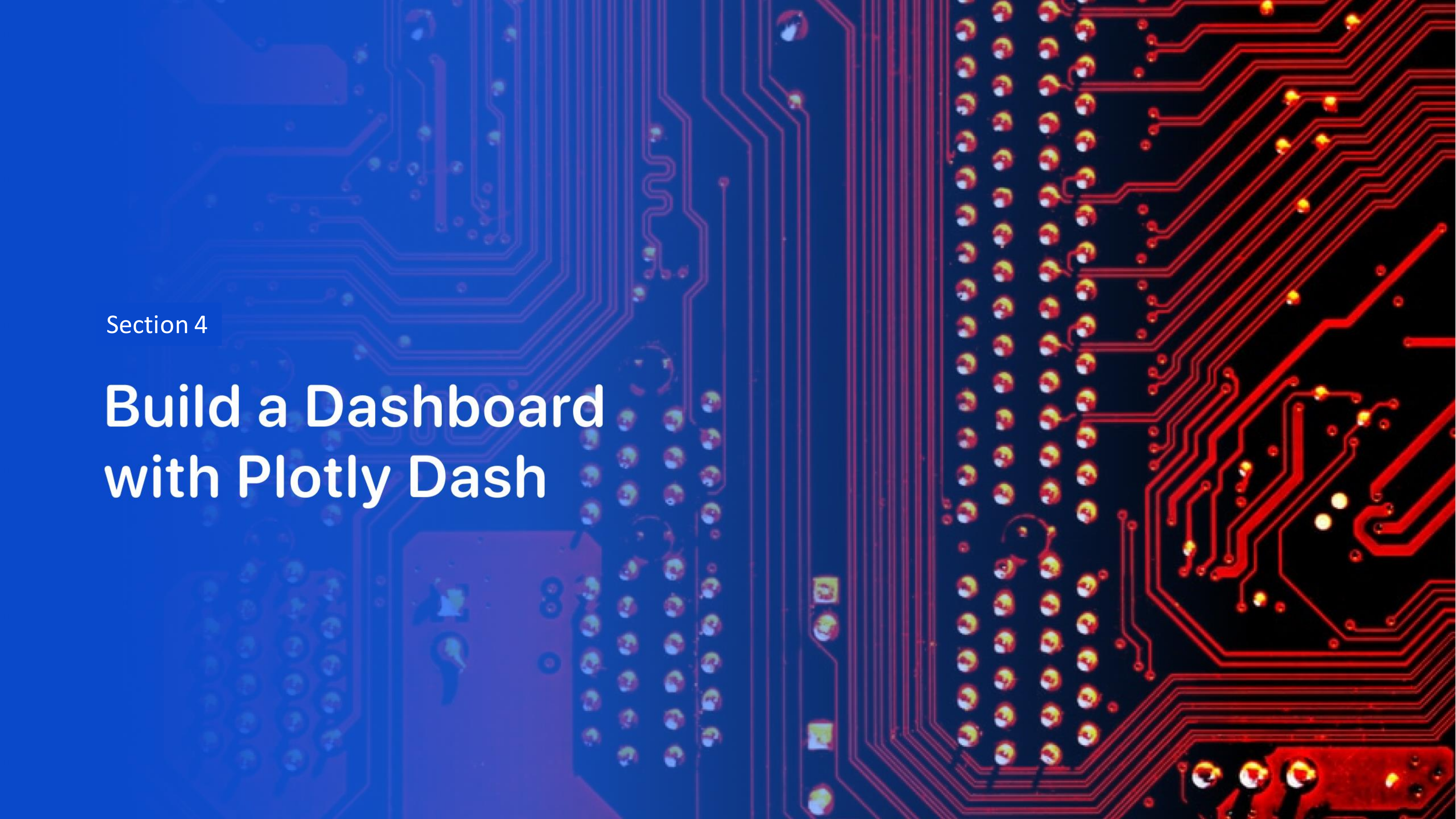
Highway, Railway and City Closest to Launch Site KSC LC-39A



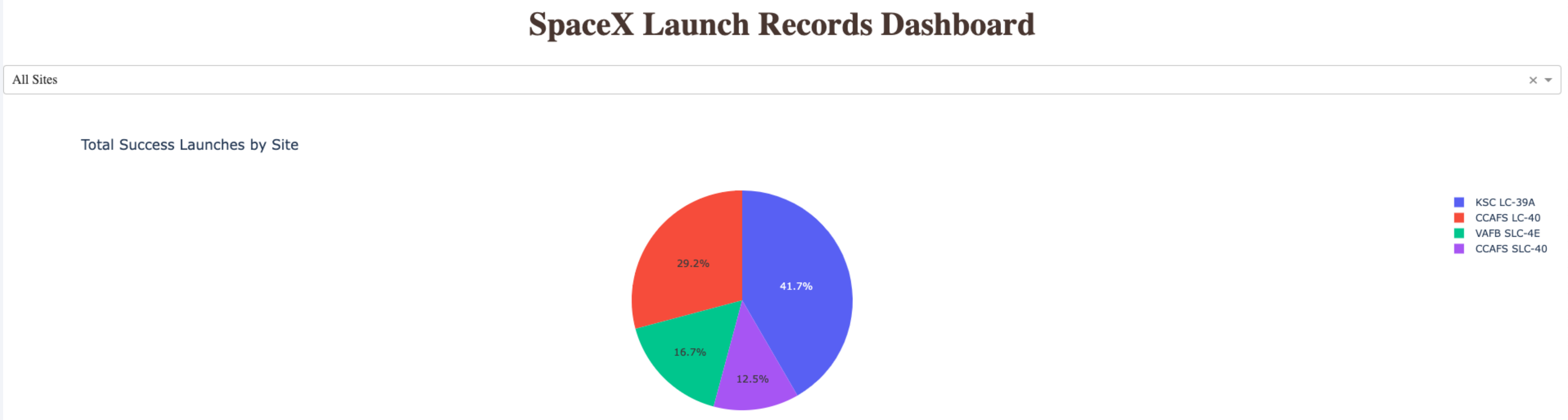
On the map we can see that generally a site is fairly further from a city but closer to a coast. There are also highways and railways nearby making transportation accessible.

Section 4

Build a Dashboard with Plotly Dash



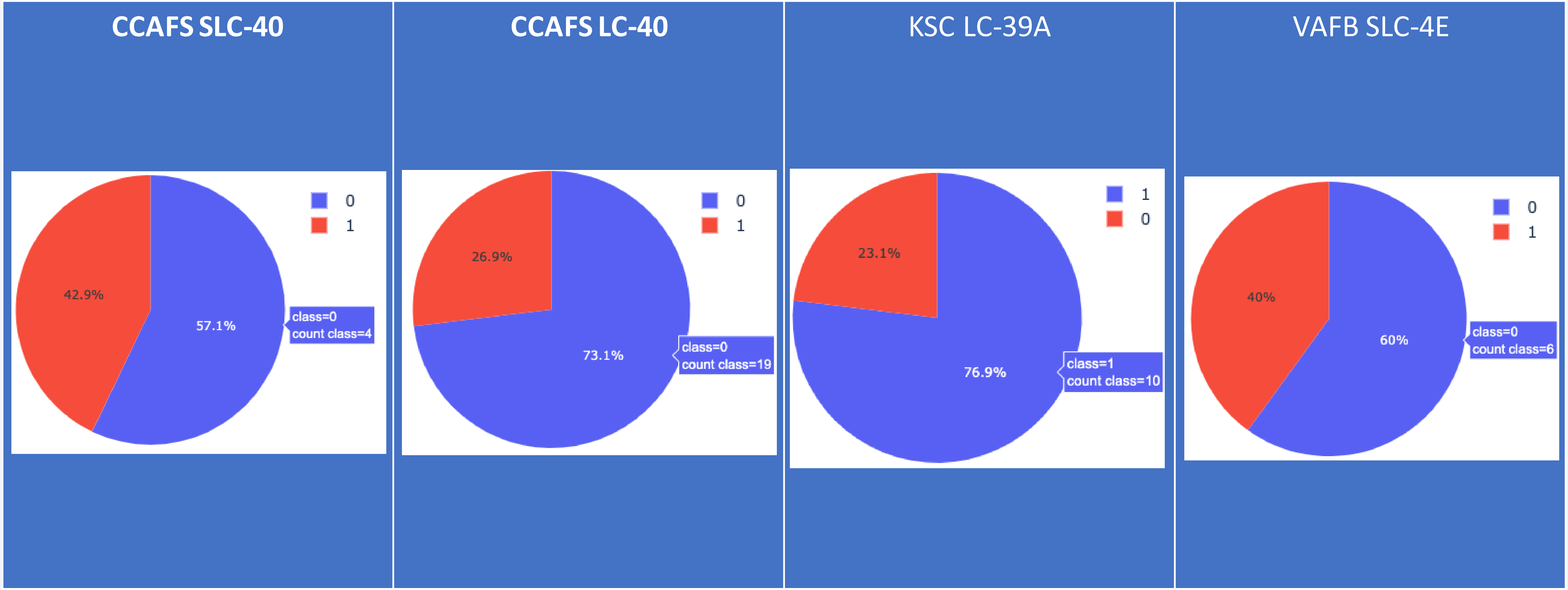
Pie Chart Showing Success Rate of Each Launch Site



SpaceX Launch Records Dashboard

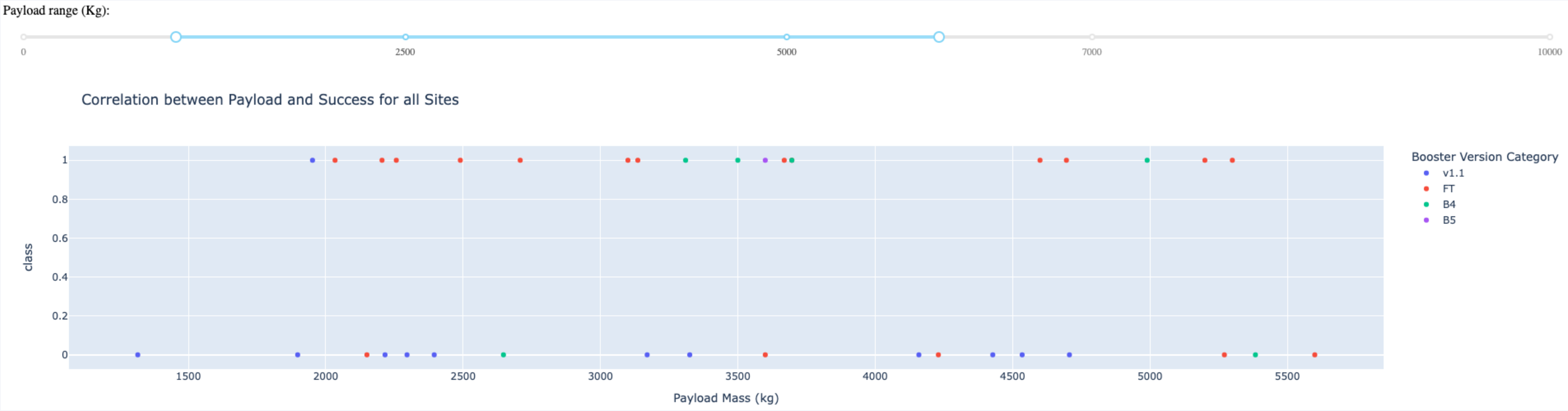
This dash shows launch site KSC LC-39A has the highest successful landing outcomes followed by launch site CCAFS LC-40, then VAFB SLC-4E and lastly CCAFS SLC-40 has the least success.

Success to Failed Launch Outcome Ratio Based by Launch Site



Here we can see launch site the highest successful to failed landing outcome ratio. With the outcome being 76.9% successful to 23.1% failed. Note: class 1 indicates successful outcome and class 0 indicates failed outcome.

Correlation between Booster Version, Payload Mass and a Successful Outcome



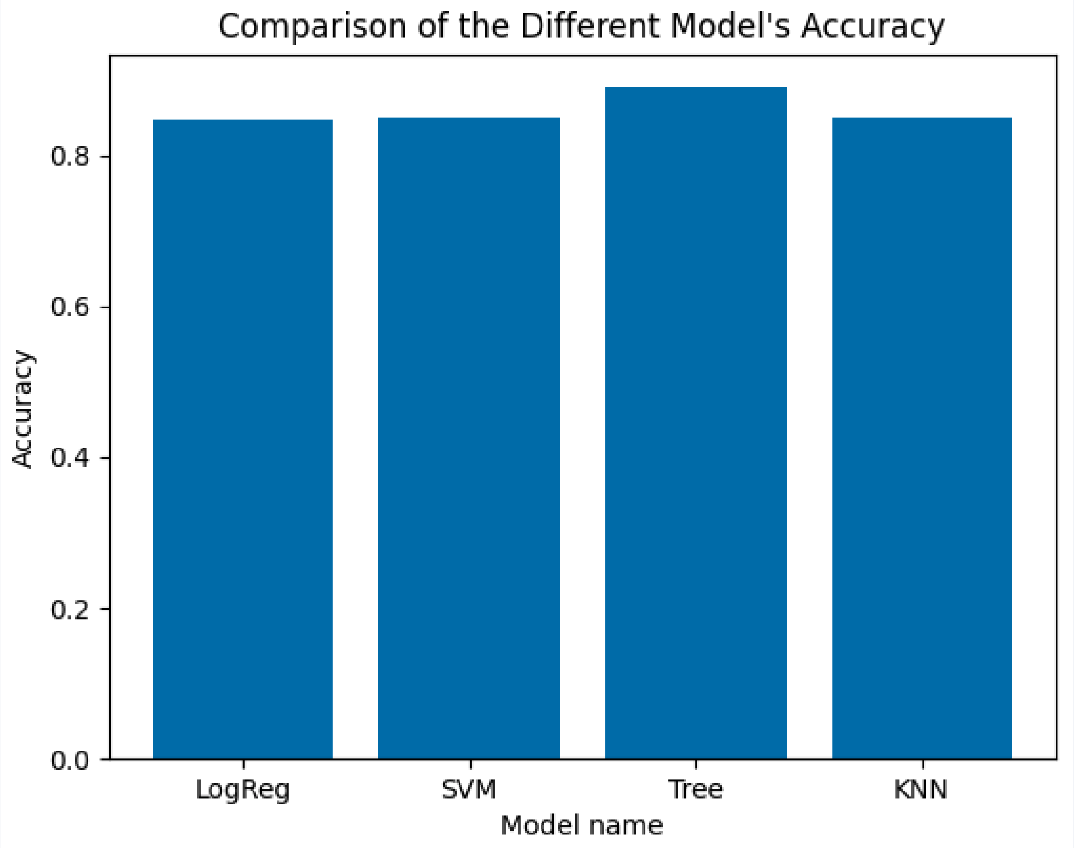
- From the first scatter plot, we can see more successful landing outcomes are scattered between payload ranges of 1900kg to 3700kg and as well as between ranges 4600 kg to 5300kg.

- We can also see that most of the successful outcomes comes from booster versions FT and in second B4. Booster B4 experiences more success when using payload mass between 3300kg to 3700kg. When using booster version FT, there is more success when using payload masses ranging between 2200kg to 3200kg.

Section 5

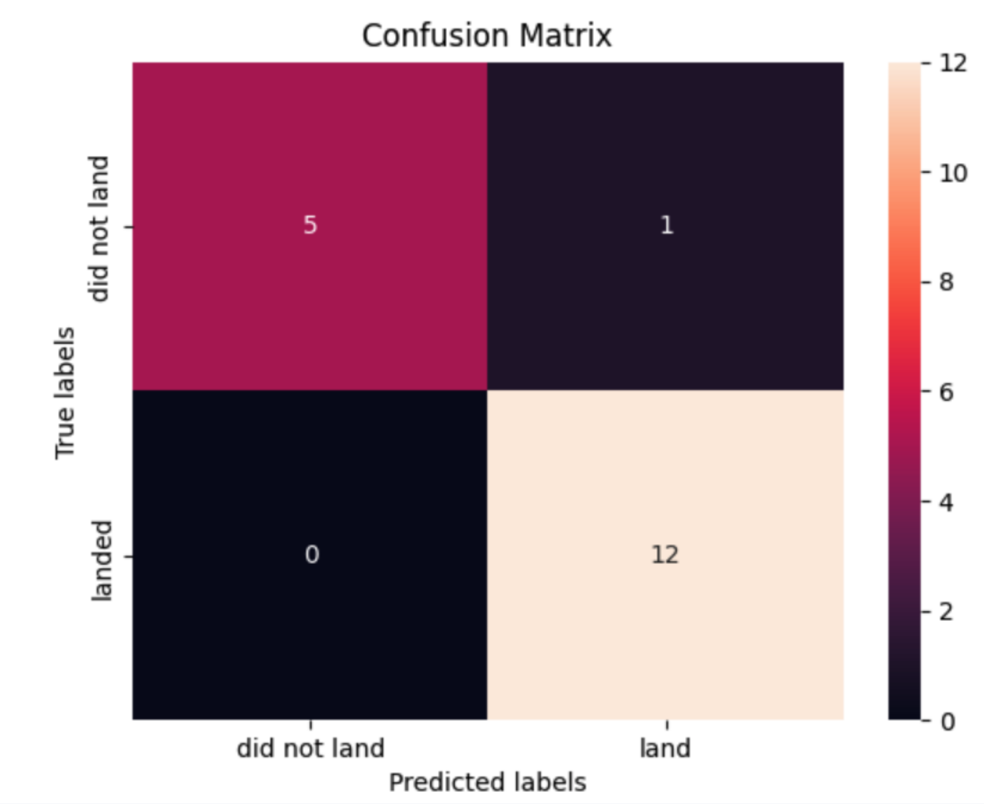
Predictive Analysis (Classification)

Classification Accuracy



From this bar chart we can see the decision tree is the model with the highest accuracy, making it the ideal model for our classification prediction model.

Confusion Matrix



Here we can see the decision tree prediction model has 5 True Negatives, 1 False Positive, 0 False Negatives and 12 True Positives results. With this info the the recall and precision is calculated to be 1.0 and 0.92 respectively. The closer to the value of 1, the better a model is. The decision tree showed higher precision compared to the other models.

Conclusions

This project analysis demonstrates we can obtain a higher odd of successfully landing if we launch from site KSC LC-39A while using a booster version FT and a payload ranging from 3600- 5300kg. Higher than 5500 kg payload with FT booster results in more failed landing.

We also saw the decision tree provided a more accurate predictive model for the classification of a landing outcome.

Thank you!