IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ogbonna Ngwu
3rd December 2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Dashboard development with Dash/plotly API
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result
  - Selection of best algorithm

Introduction

- **Project background and context**

  The aim is to build a machine learning system predicting the successful landing of SpaceX Falcon 9 first stages, crucial for cost-effective launches. This could aid competitors in bidding against SpaceX by estimating their launch costs based on the likelihood of first-stage reusability, a key factor in reducing expenses compared to other providers.

- **Problems you want to find answers**

  1. What contributes to a successful rocket landing?
  2. The interplay of different factors impacting the likelihood of a successful landing.
  3. What conditions are necessary for a successful landing program's operation?

Section 1

Methodology

Methodology

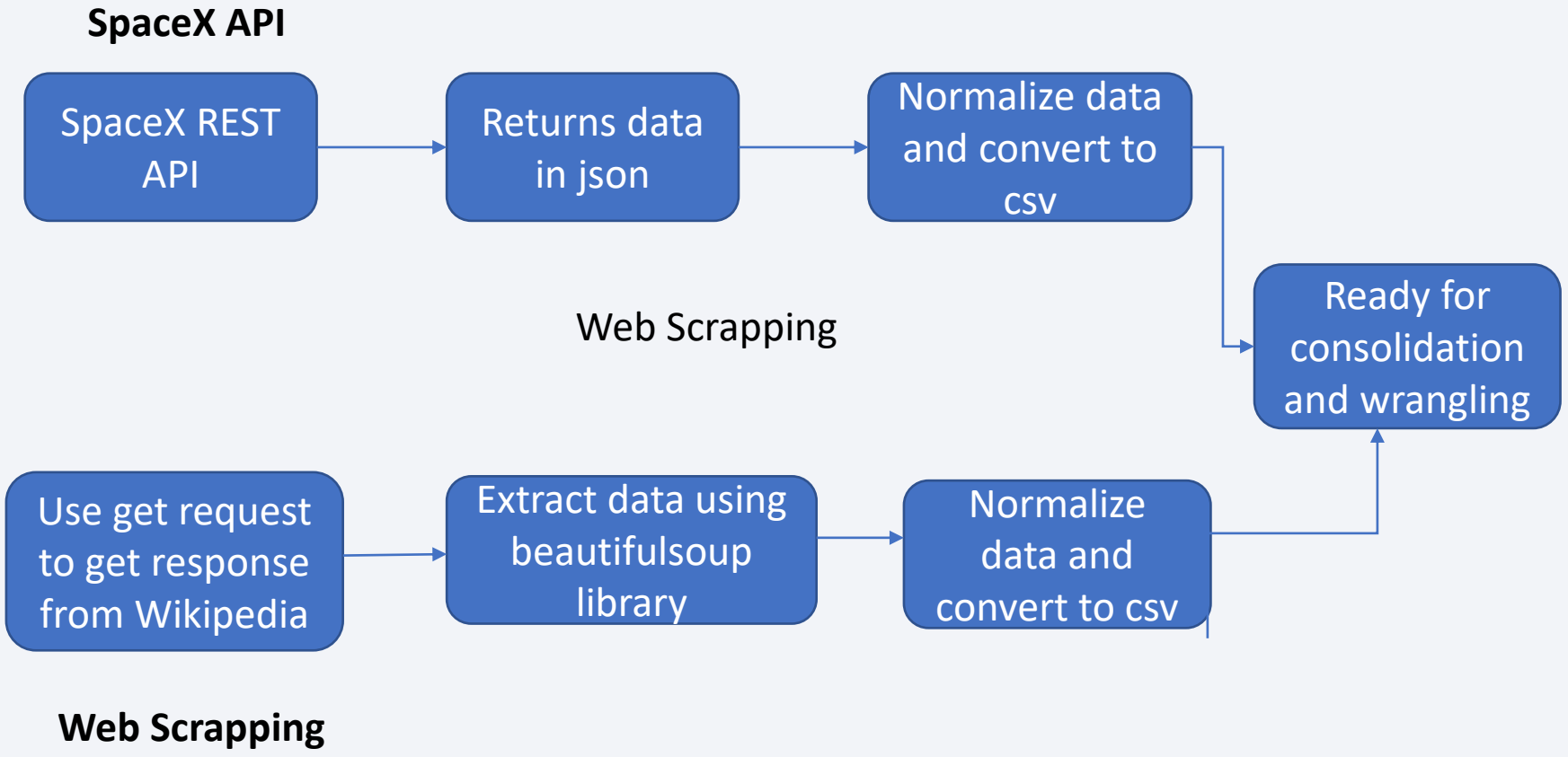
Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia
- Perform data wrangling
  - Detected and removed NULL values
  - Did value counts of categorical columns unique values to see the distribution
  - One-hot encoding was applied to categorical features
  - Created landing outcome Label column.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - SVM, DS, KNN and Logistics Regression models was built and evaluated.

Data Collection

- Describe how data sets were collected.
  - Data collection was done using get request to the SpaceX API.
  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
  - We then cleaned the data, checked for missing values and fill in missing values where necessary.
  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

Data Collection Flowchart



The following datasets was collected:

- SpaceX launch data through REST API
- This data includes info on rocket used, payload delivered, launch specifications and landing outcomes.
- The same data from SpaceX was also gathered through web scraping from Wikipedia using BeautifulSoup

Data Collection – SpaceX API

- I utilized the SpaceX API's GET request to gather data, performed data cleaning, and conducted basic formatting and manipulation.
- GitHub UR: https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/spacex-data-collection-api.ipynb

Steps taken/FlowChart:

- Request data from SpaceX API (rocket launch data)
- Decode response using .json() and convert to a dataframe using .json_normalize()
- Request information about the launches from SpaceX API using custom functions
- Create dictionary from the data
- Create dataframe from the dictionary
- Filter dataframe to contain only Falcon 9 launches
- Replace missing values of Payload Mass with calculated .mean()
- Export data to csv file

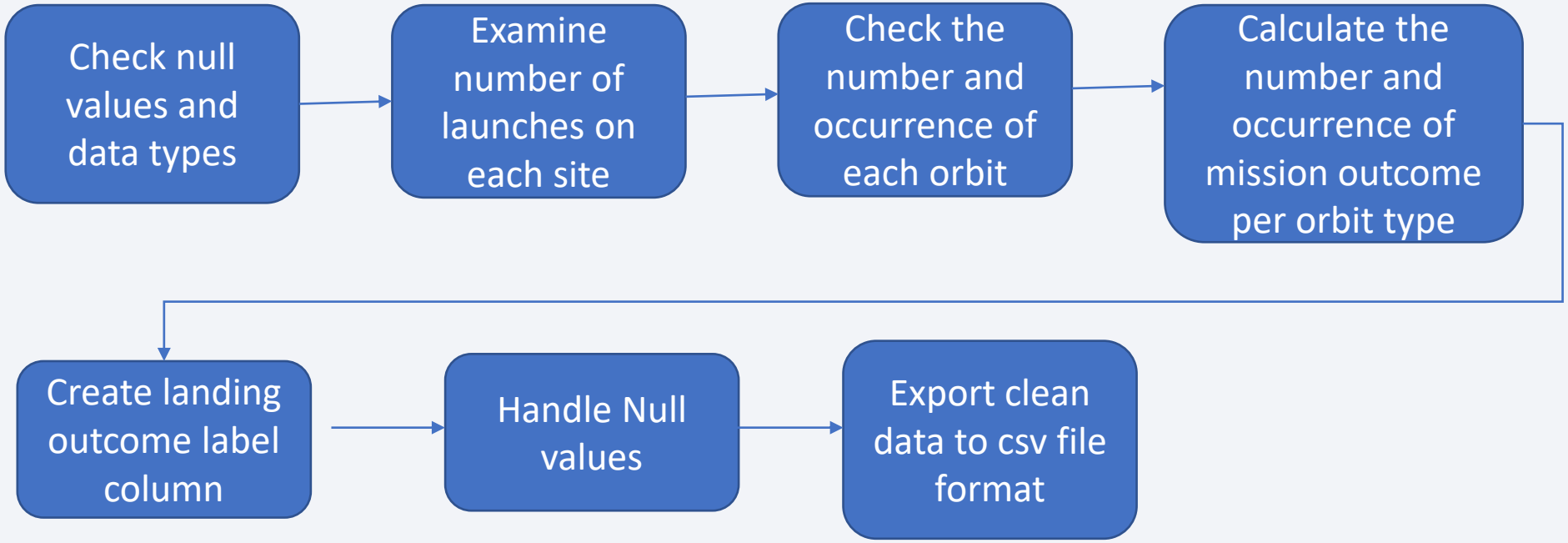
Data Collection - Scraping

- I applied web scrapping to get Falcon 9 launch records with BeautifulSoup
- Parsed the table and converted it into a pandas dataframe.
- GitHub UR: https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/webscraping_spacex_data.ipynb

Steps taken/FlowChart

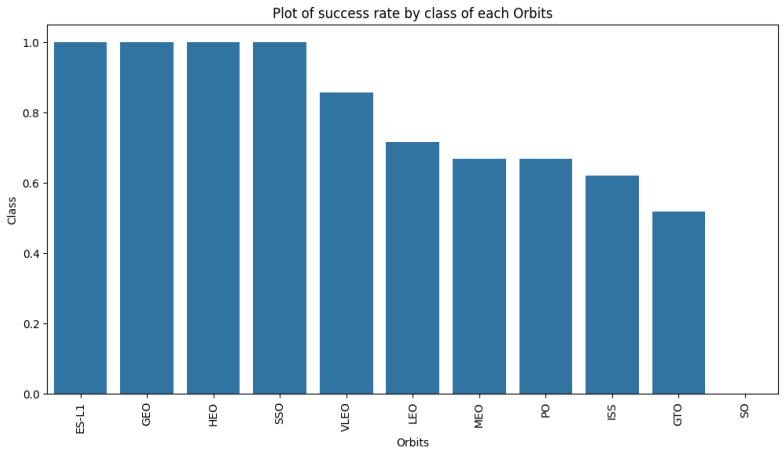
- Request data (Falcon 9 launch data) from Wikipedia
- Create BeautifulSoup object from HTML response
- Extract column names from HTML table header
- Collect data from parsing HTML tables
- Create dictionary from the data
- Create dataframe from the dictionary
- Export data to csv file

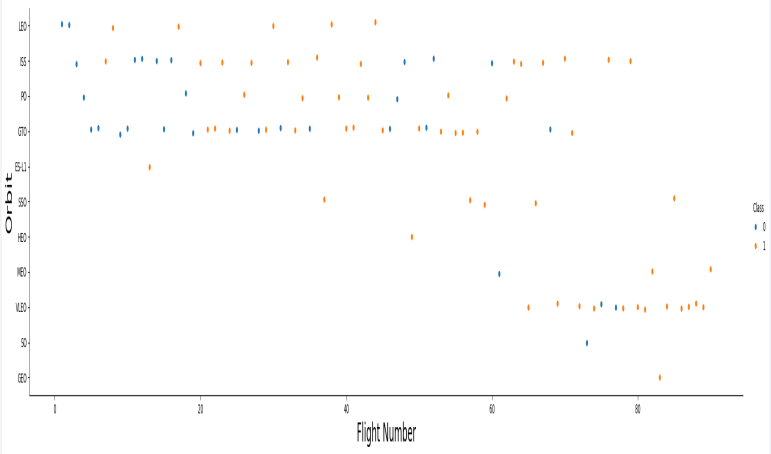
Data Wrangling

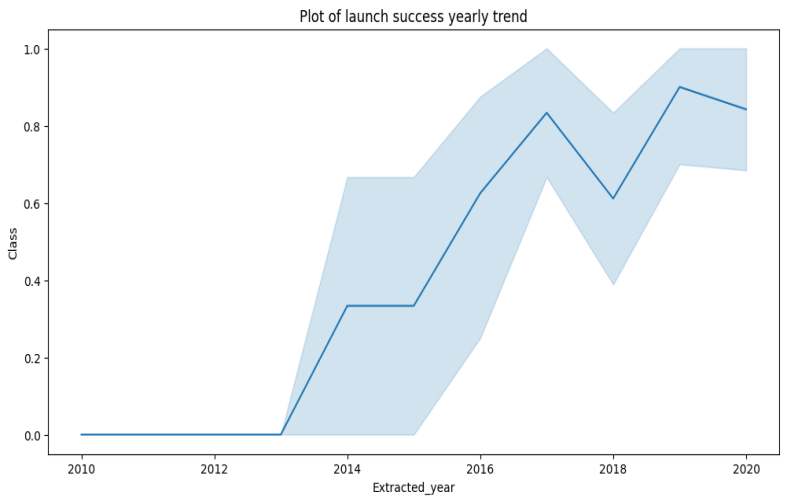


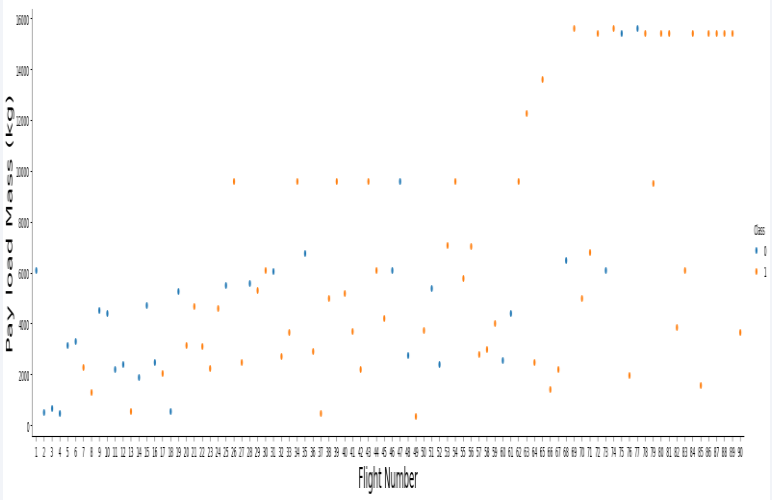
GitHub URL: https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/spacex-Data%20wrangling.ipynb

EDA with Data Visualization









- Bat Chart: This enablednme to Visualize the relationship between landing success rate of each orbit type.
- Line Chart: This helped me to track the yearly launch success rate.
- Catplot/scatter plot: I used this plot to discover relationships between various and overlayed the launch out to be able to what factors influence launch success/failure more
- GitHub URL https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/eda-dataviz.ipynb

EDA with SQL

- Tasks perfomed with SQL queries includes:
  - Displaying the names of the unique Rocket launch sites
  - Displaying 5 records where launch sites begin with “CCA”
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Displaying the total payload mass carried by booster versions F9 v1.1
  - List the date when the first successful landing outcome in ground pad was achieved.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster versions which have carried the maximum payload mass
  - 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.
  - 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.
- GitHub URL https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/eda-sql.ipynb

Build an Interactive Map with Folium

- I mapped launch sites and incorporated markers, circles, and lines to indicate launch outcomes (success/failure) on a folium map.
- Launch outcomes were categorized as 0 for failure and 1 for success.
- Through color-coded markers, I identified sites with notably high success rates.
- Distances between launch sites and their surroundings were computed, exploring proximity to railways, highways, coastlines, and distances from cities.

GitHub URL https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/launch_site_location_with_folium.ipynb

Build a Dashboard with Plotly Dash

- I created an engaging Plotly Dash dashboard.
- I added a dropdown menu so a user could filter the dashboard by a specific launch site
- I added a payload slide bar, enabling a user to filter scatter plot relationships between variables.
- I generated pie charts illustrating the total launches per specific sites. Additionally, I depicted scatter plots demonstrating the correlation between Outcome and Payload Mass (Kg) across various booster versions.

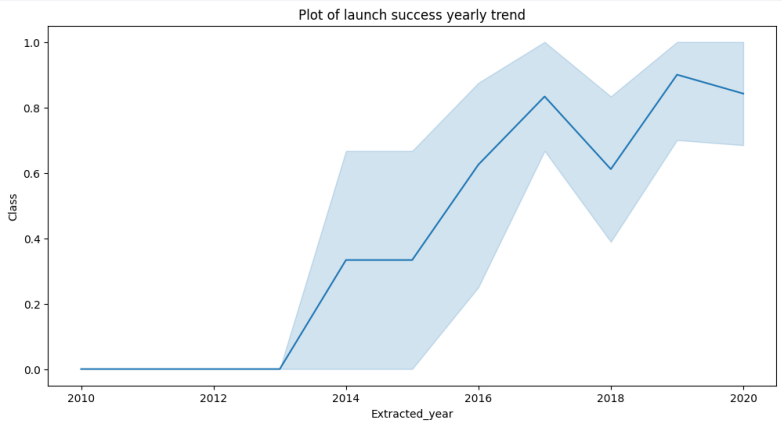
GitHub URL https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/Dash_app_dashboard.py

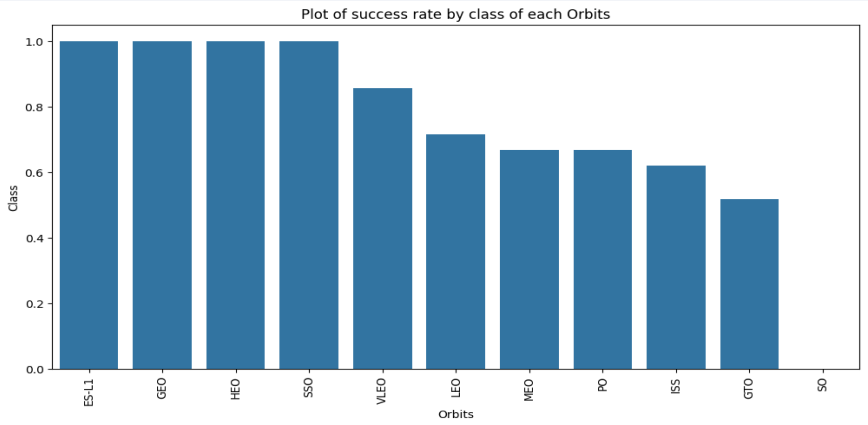
Predictive Analysis (Classification)

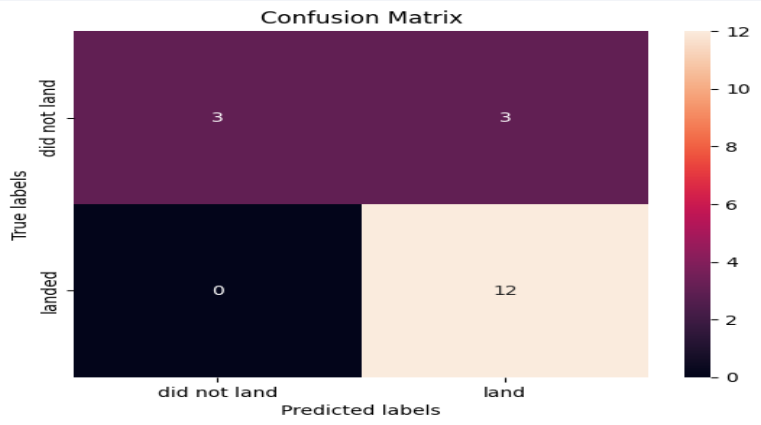
- I loaded the data using numpy and pandas, transformed the data, split our data into training and testing.
- I built different machine learning models(KNN, DS, SVM, Logistics regression) and tuned different hyperparameters using GridSearchCV.
- I used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- The SVM. KNN and Logistics Regression models achieved the highest accuracy at 83.3%, while KNN performs the best in terms of accuracy on test dataset.
- You need present your model development process using key phrases and flowchart
- GitHub URL https://github.com/ngwuprince/IBM-Coursera-Data-Science-Capstone-Project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.ipynb

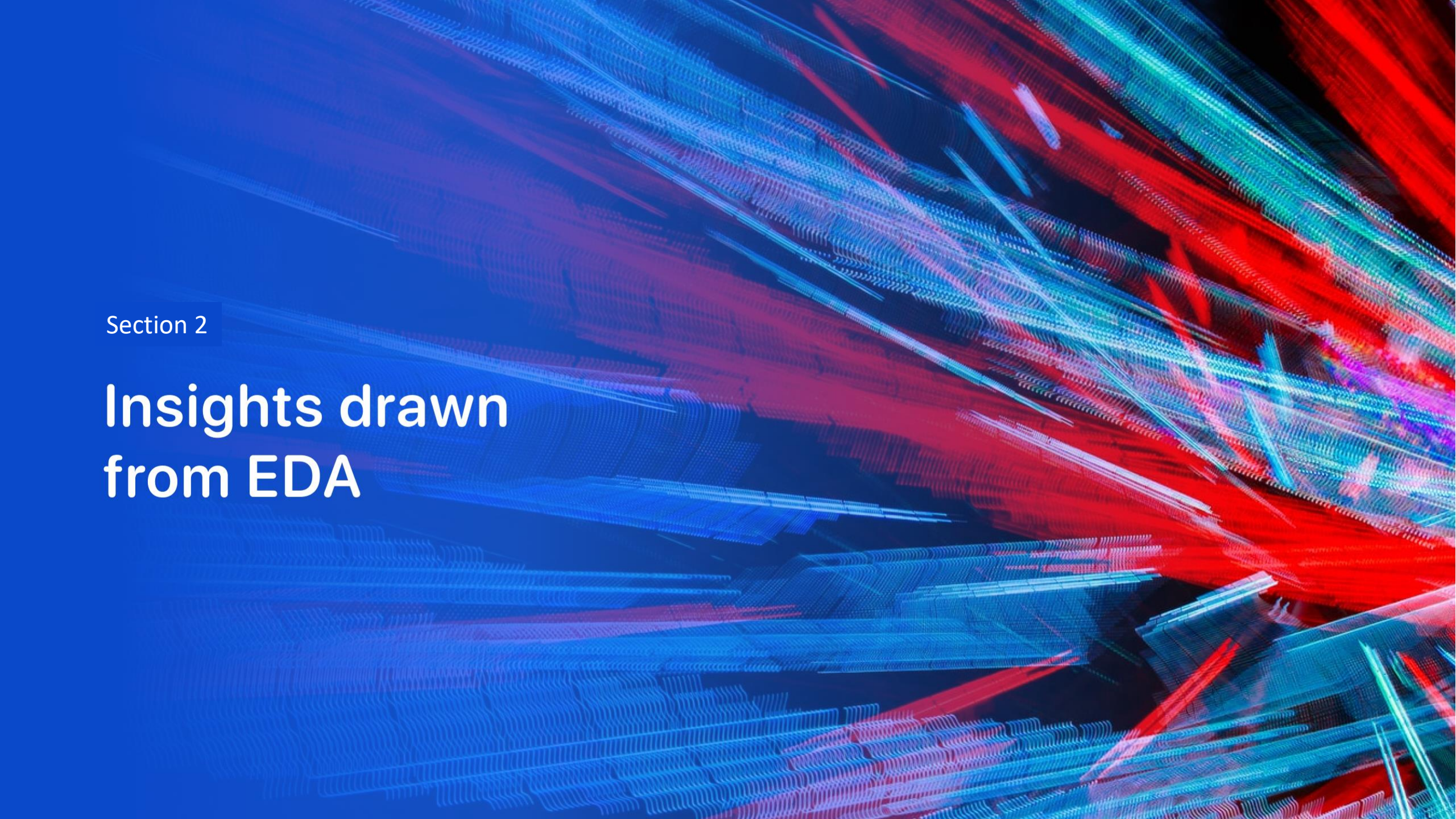
Results

- Different launch sites have different success rates. CCAFS LC-40 has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.
- 1. There are far more Launches at CCAFS SLC 40 LaunchSite than any other site
- 2. KSC LC 39A has higher successful landing outcome rate.
- The success rate for SpaceX launches increases with increase in number of years
- In the LEO orbit the Success appears related to the number of flights.
- Orbits GEO, HEO, SSO, ES L1 has the best success rates.
- Low weighted payloads perform better than the heavier payloads
- The SVM, KNN, Decision tree and Logistics Regression models all have equal accuracy of 83.3%.

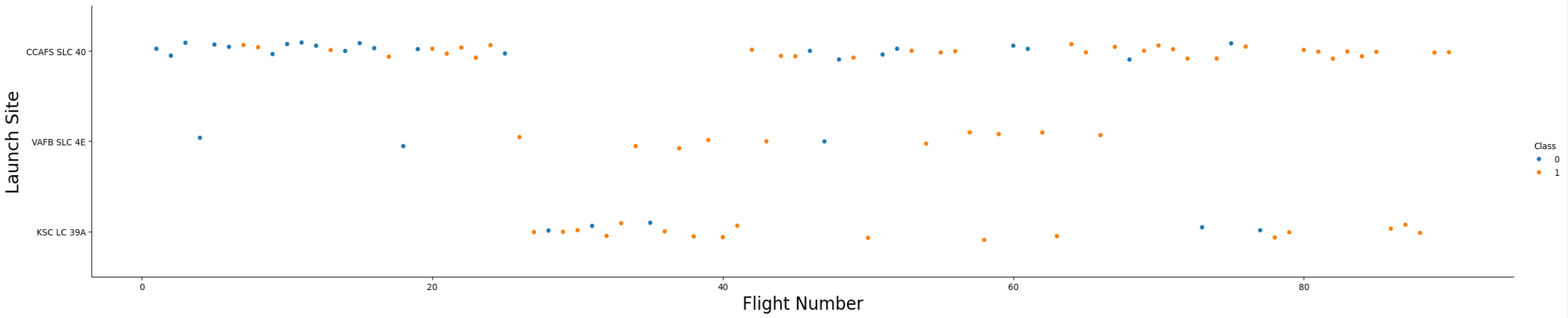






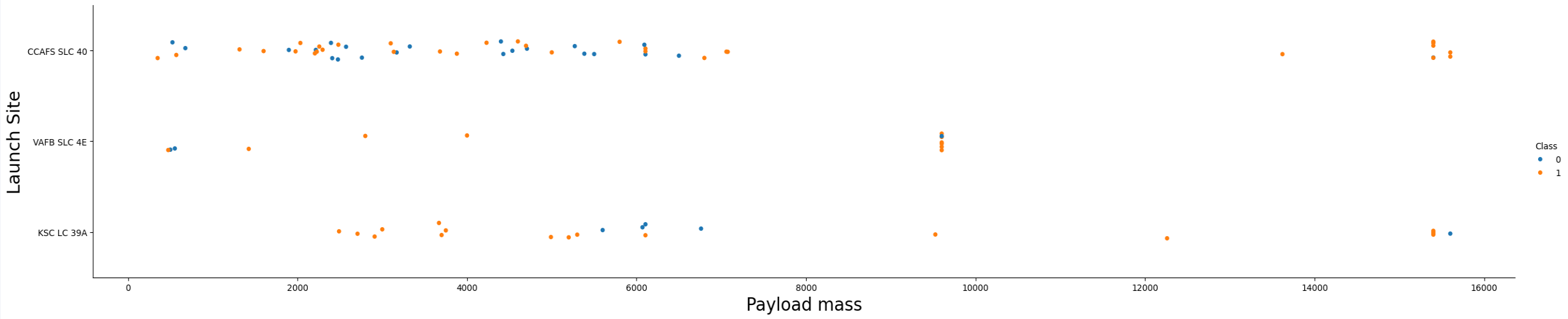

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



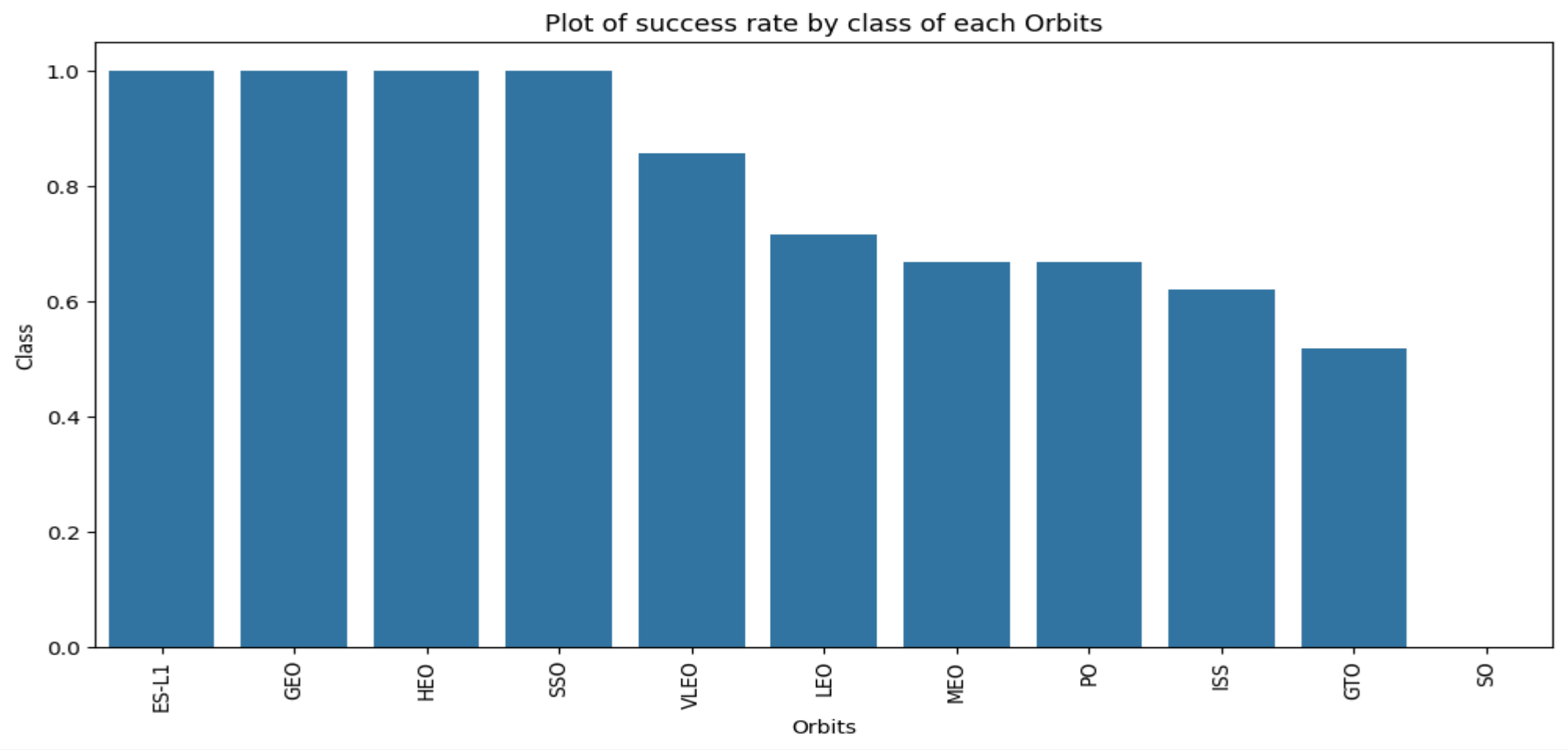
- There are far more Launches at CCAFS SLC 40 LaunchSite than any other site
- From the plot, I found that the larger the flight amount at a launch site, the greater the success rate at a launch site.

Payload vs. Launch Site



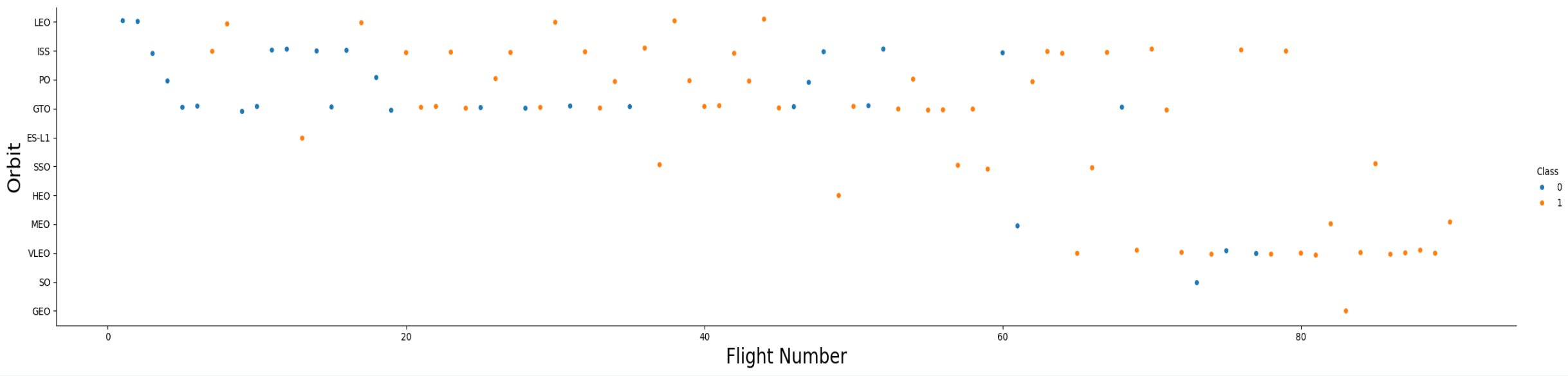
- Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavy payload mass(greater than 10000)
- The greater the payload for CCAFS SLC 40, the higher the success rate.

Success Rate vs. Orbit Type



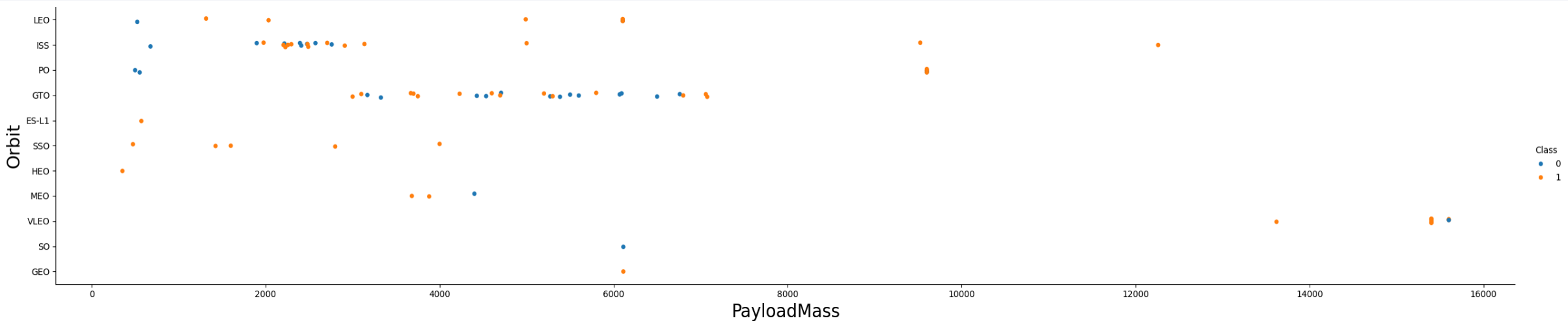
- Orbits GEO, HEO, SSO, ES-L1 has the best success rates

Flight Number vs. Orbit Type



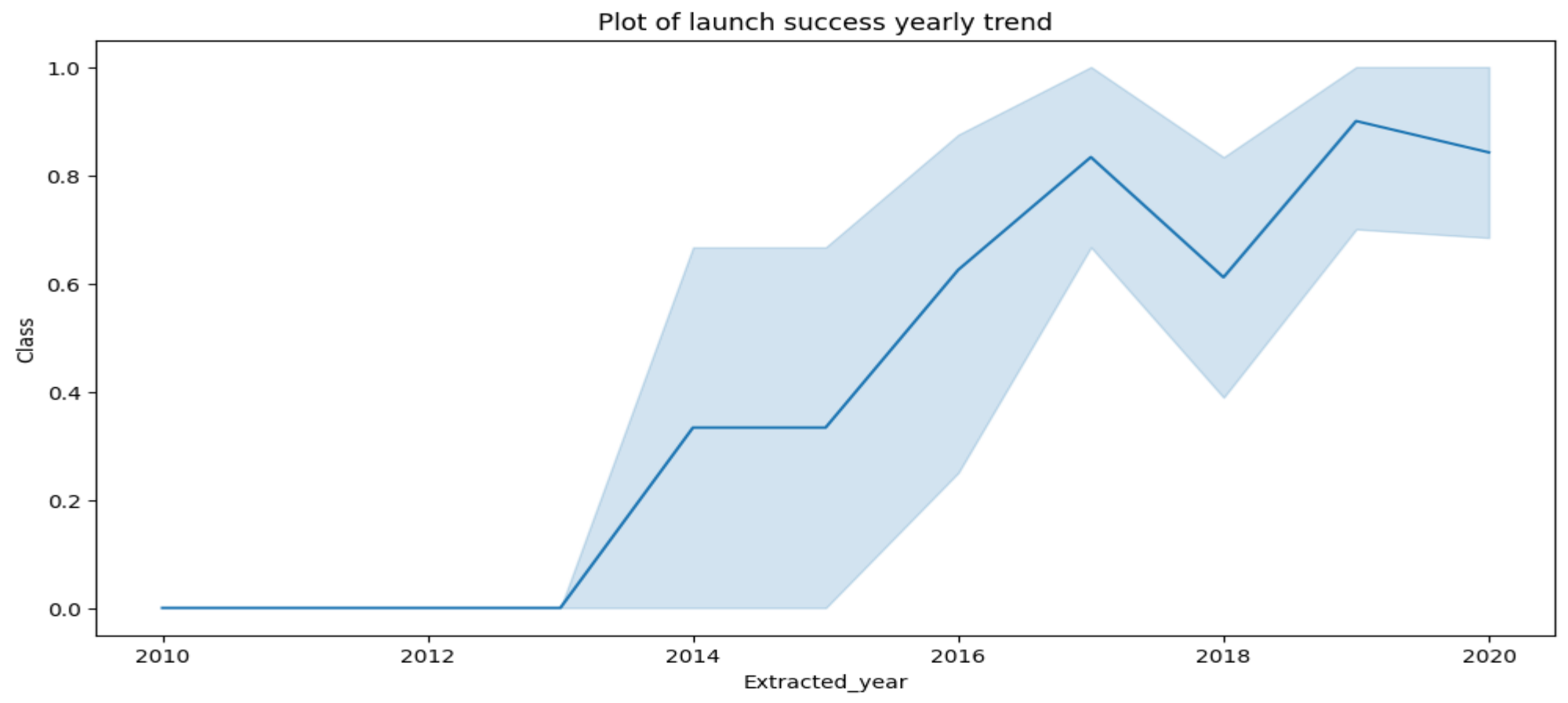
- in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



- 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.

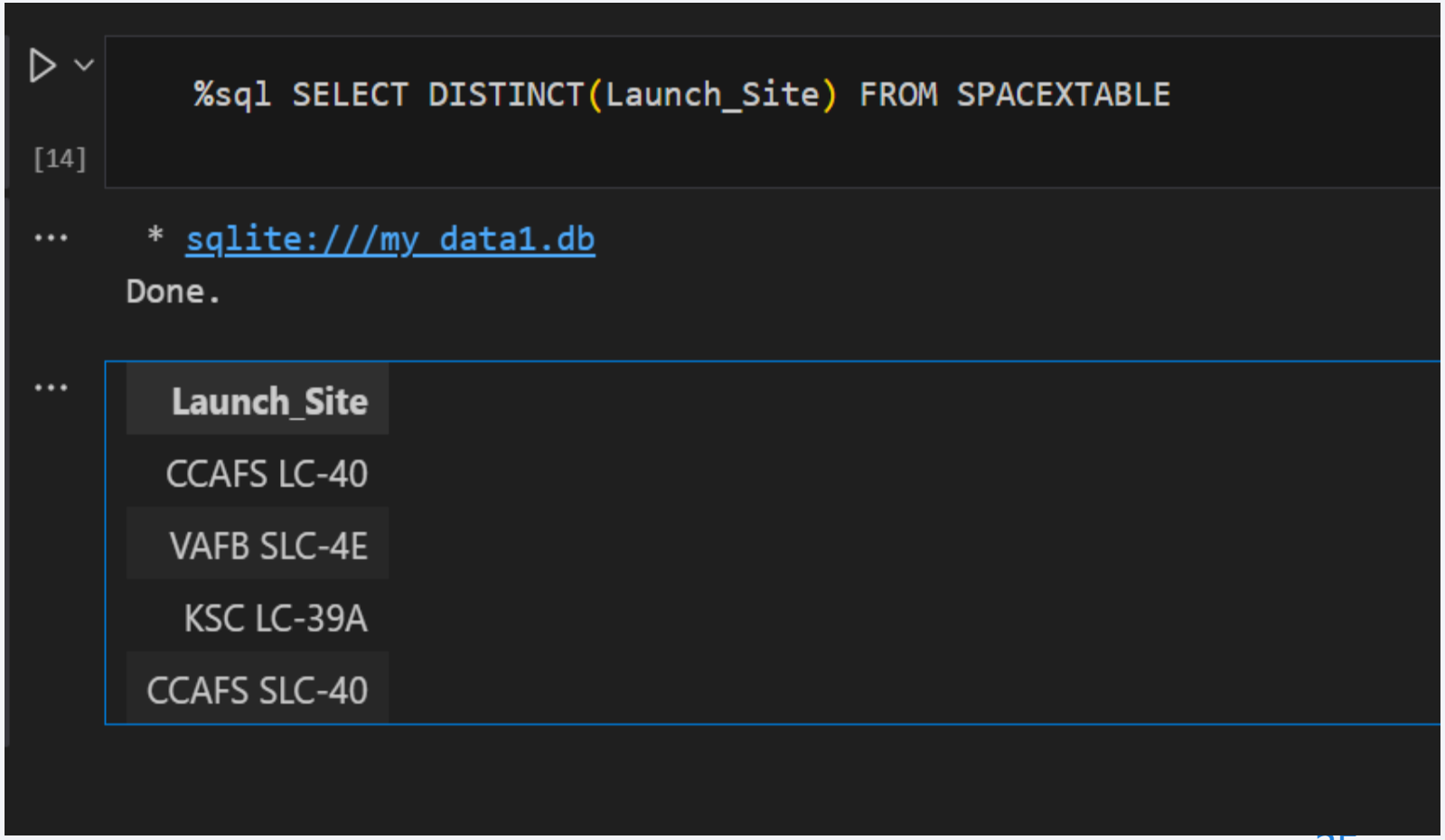
Launch Success Yearly Trend



- The success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

- I used the key word **DISTINCT** to show only unique launch sites from the SpaceX data.



Launch Site Names Begin with 'CCA'

- Used wildcard to search the launch site name with CCA.
- Used LIMIT 5 to show just five rows of data.

```
%%sql SELECT * FROM SPACEXTABLE
WHERE Launch_Site LIKE 'CCA%'
LIMIT 5
```

[16] Python

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure
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
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	

Total Payload Mass

- Used SUM function to add all payload mass

```
%%sql
select sum(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTABLE
```

[46]

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

payloadmass
619967

Average Payload Mass by F9 v1.1

- Used AVG function to compute the average value.

```
%sql select avg(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTABLE
```

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

payloadmass
6138.287128712871

First Successful Ground Landing Date

- Used Min function to filter the earliest grand landing date.

```
[57]  %sql select min(DATE) from SPACEXTBL;

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

...    min(DATE)
       2010-06-04
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- Used the **WHERE** clause to filter for boosters which have successfully landed on drone ship and applied the **AND** condition to determine successful landing with payload mass greater than 4000 but less than 6000

```
[84]  %%sql select "Booster_Version"
      from SPACEXTABLE
      where "LANDING__OUTCOME"="Success (drone ship)"
      and "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;

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

...   Booster_Version
```

Total Number of Successful and Failure Mission Outcomes

- Used Count function to count the number of occurrence of each outcome value.

```
%%sql select MISSION_OUTCOME, count(MISSION_OUTCOME) as missionoutcomeCOUNT
from SPACEXTABLE
GROUP BY MISSION_OUTCOME;
```

[71]

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

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

Boosters Carried Maximum Payload

- Determined the booster that have carried the maximum payload using a subquery in the **WHERE** clause and the **MAX()** function.

```
%%sql select BOOSTER_VERSION as boosterversion
from SPACEXTABLE where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

[74]

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

boosterversion
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

- Used a combination of the **WHERE** clause, **LIKE**, **AND**, and **BETWEEN** conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

In [18]:
```
task_9 = '''
        SELECT BoosterVersion, LaunchSite, LandingOutcome
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Failure (drone ship)'
            AND Date BETWEEN '2015-01-01' AND '2015-12-31'
        '''
create_pandas_df(task_9, database=conn)
```

Out[18]:

	boosterversion	launchsite	landingoutcome
0	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
1	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Selected Landing outcomes and the **COUNT** of landing outcomes from the data and used the **WHERE** clause to filter for landing outcomes **BETWEEN** 2010-06-04 to 2017-03-20.
- Applied the **GROUP BY** clause to group the landing outcomes and the **ORDER BY** clause to order the grouped landing outcome in descending order.

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad))

In [19]:
```
task_10 = '''
        SELECT LandingOutcome, COUNT(LandingOutcome)
        FROM SpaceX
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LandingOutcome
        ORDER BY COUNT(LandingOutcome) DESC
        '''
create_pandas_df(task_10, database=conn)
```

Out[19]:

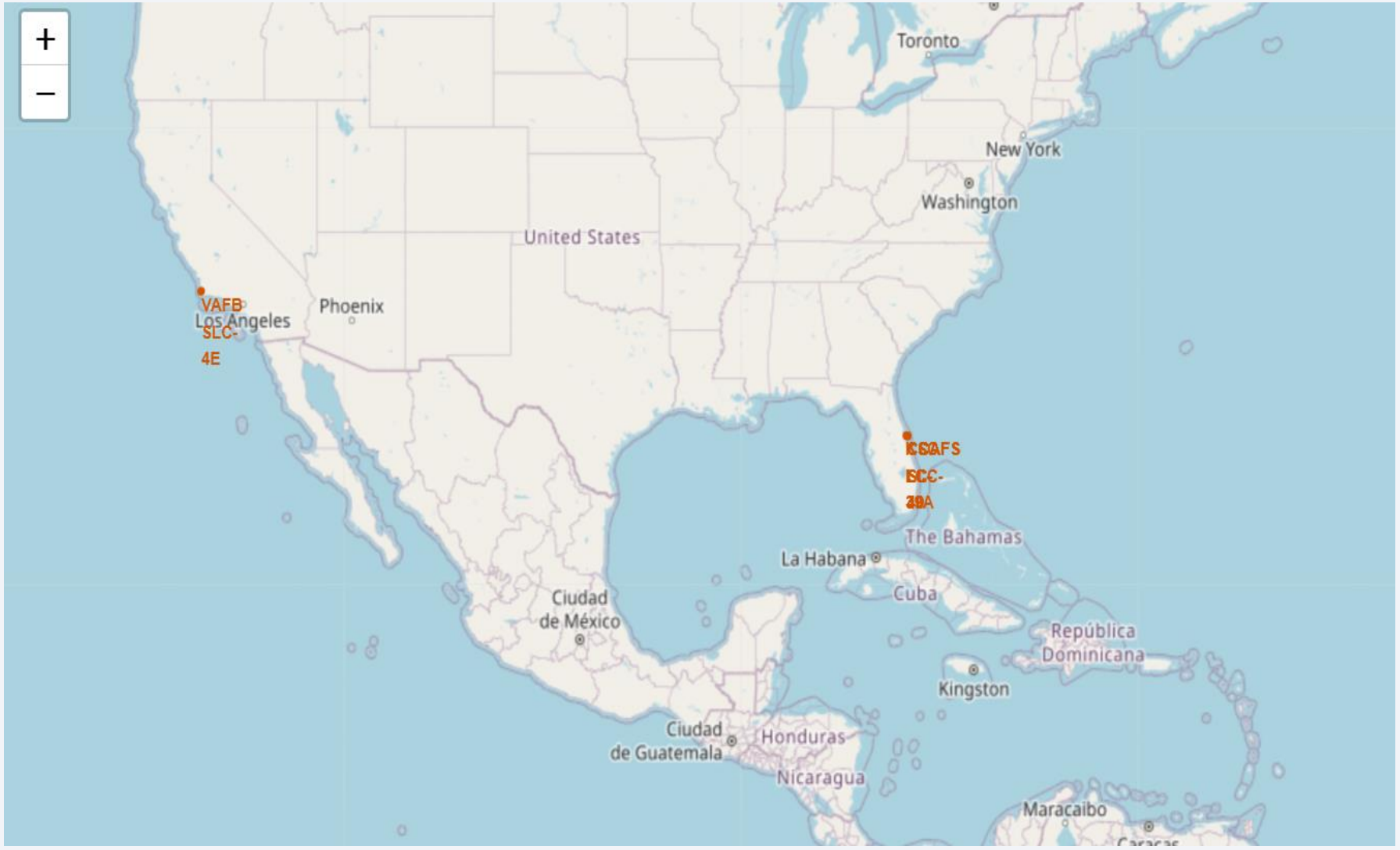
	landingoutcome	count
0	No attempt	10
1	Success (drone ship)	6
2	Failure (drone ship)	5
3	Success (ground pad)	5
4	Controlled (ocean)	3
5	Uncontrolled (ocean)	2
6	Precluded (drone ship)	1
7	Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

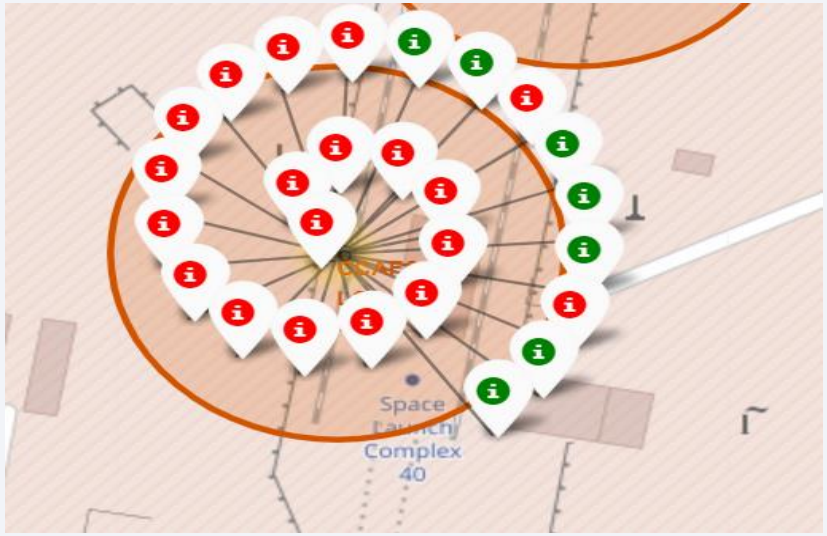
Launch Sites on a Map

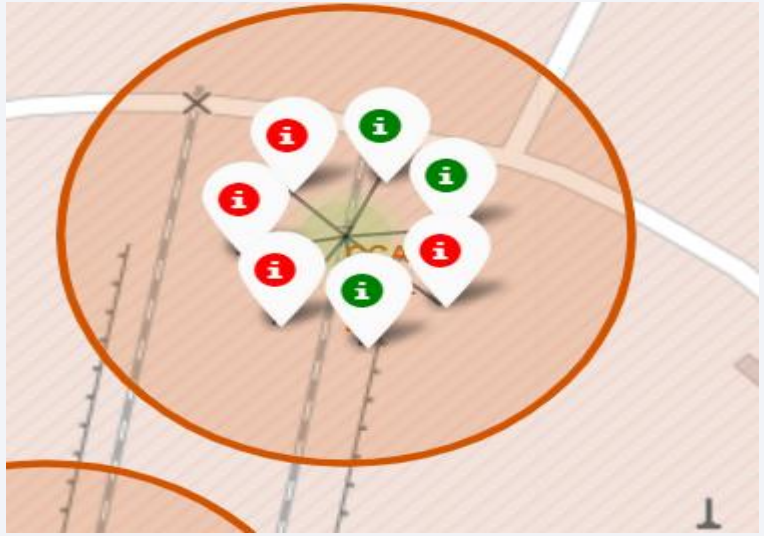
- The Launch sites are all located close to the coasts at Florida and California respectively.

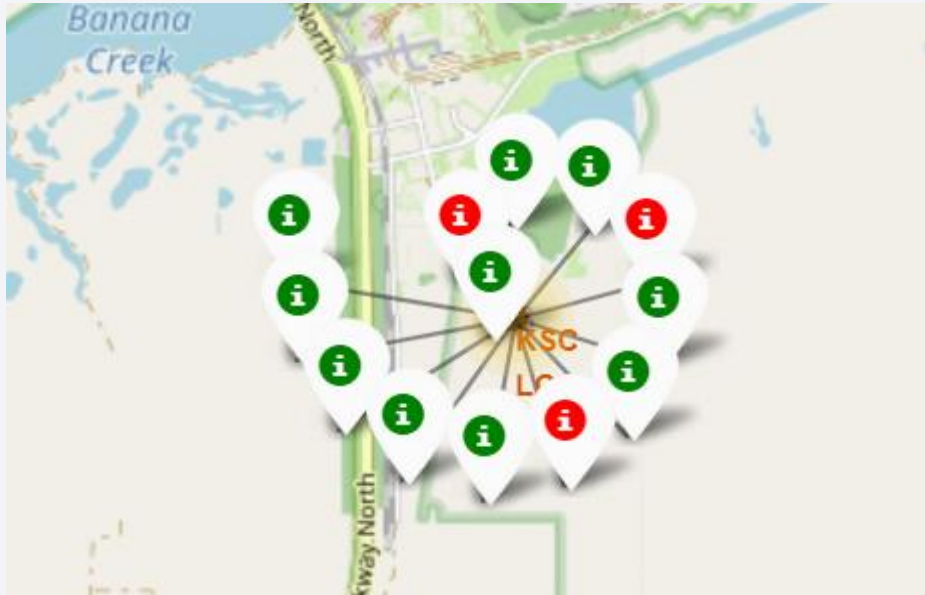


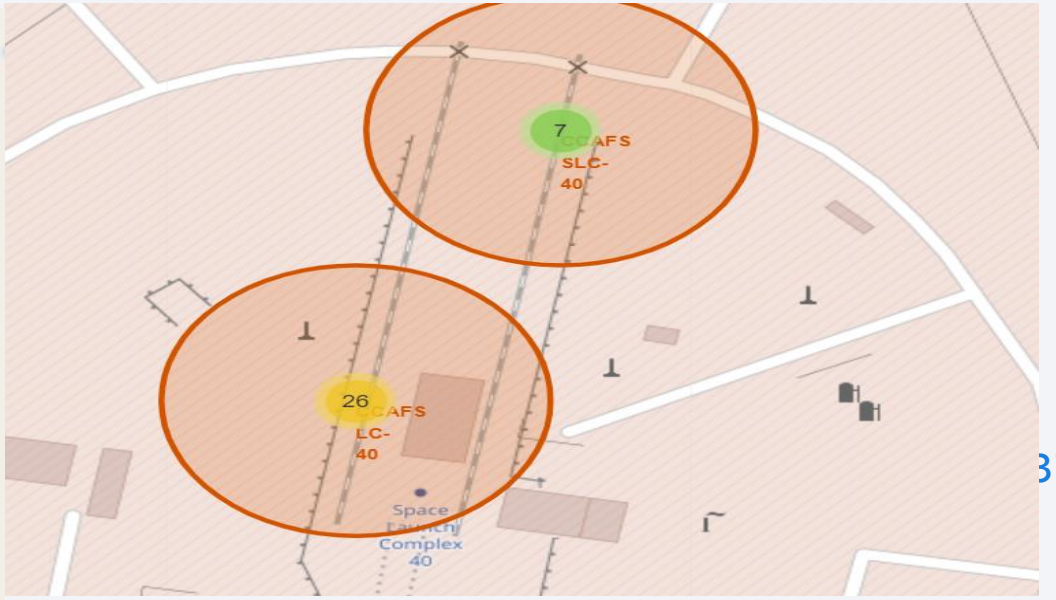
Markers showing launch sites with color labels

- The green markers shows the successful launches and Red marker shows failed launches







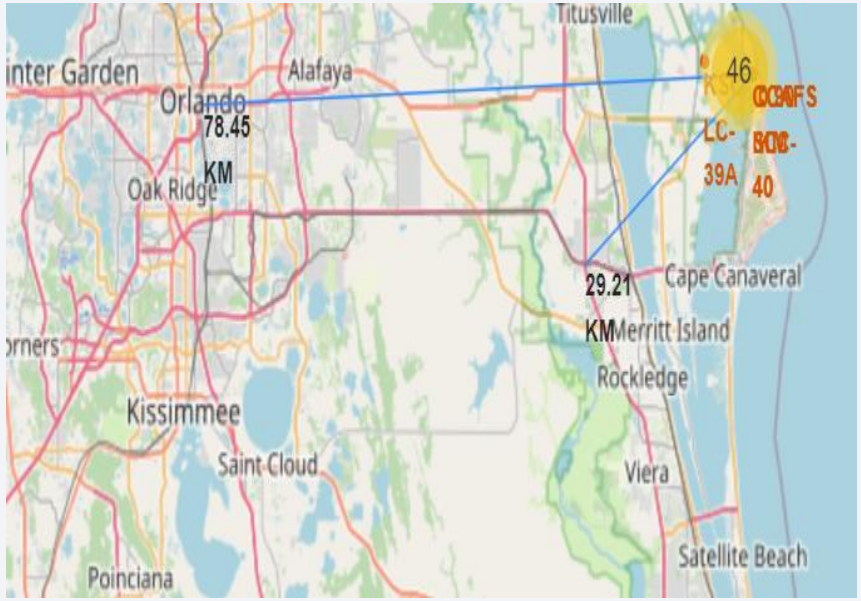


Launch Site distance to landmarks

KEY QUESTIONS AND ANWERS

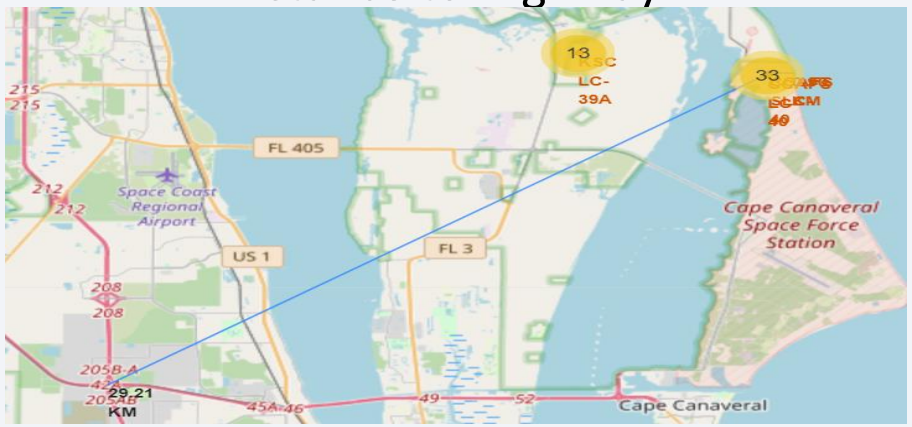
- Are launch sites in close proximity to railways? NO
- Are launch sites in close proximity to highways?NO
- Are launch sites in close proximity to coastline? YES
- Do launch sites keep certain distance away from cities? YES

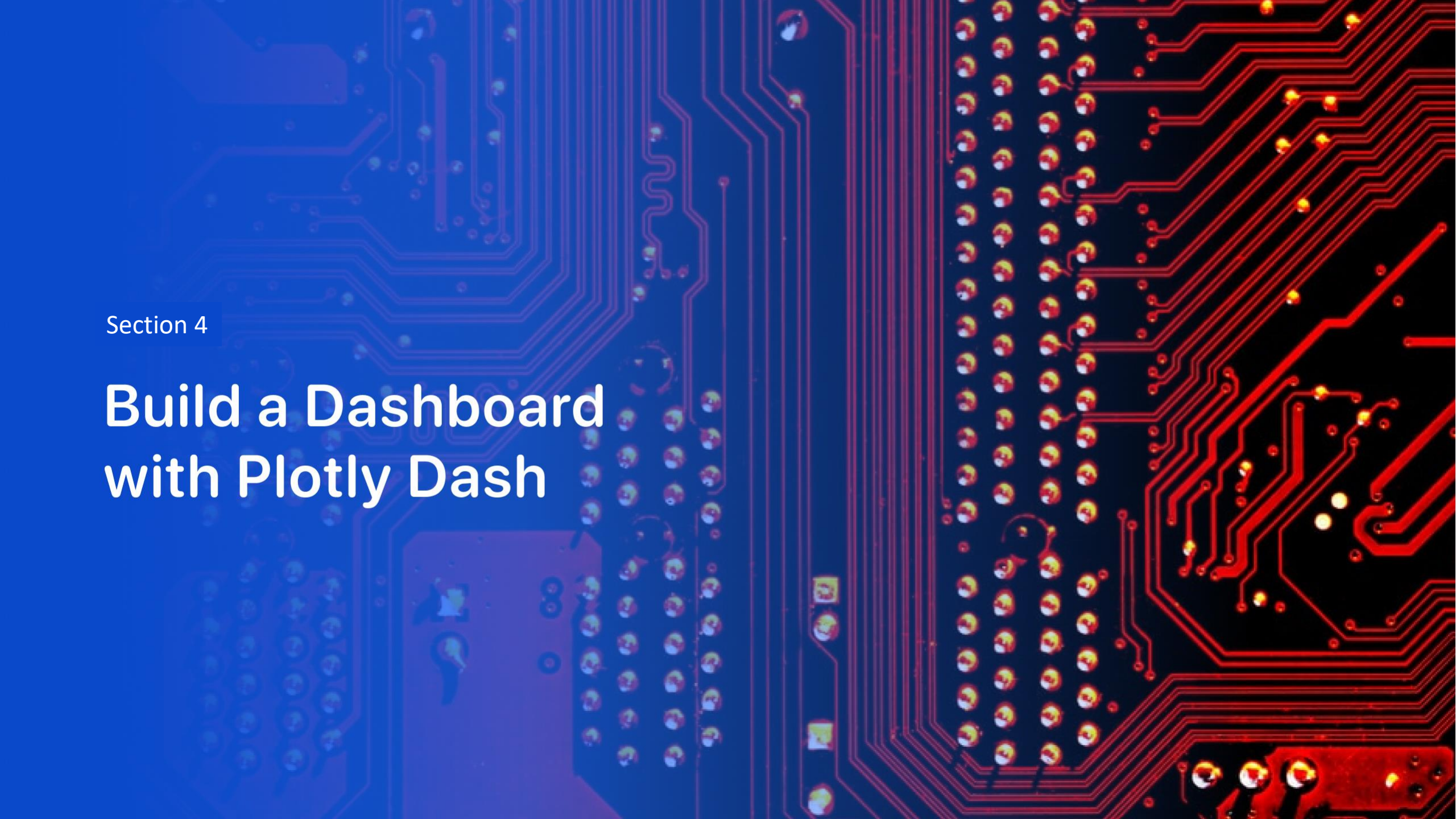
Distance to Florida City



Distance to coast line

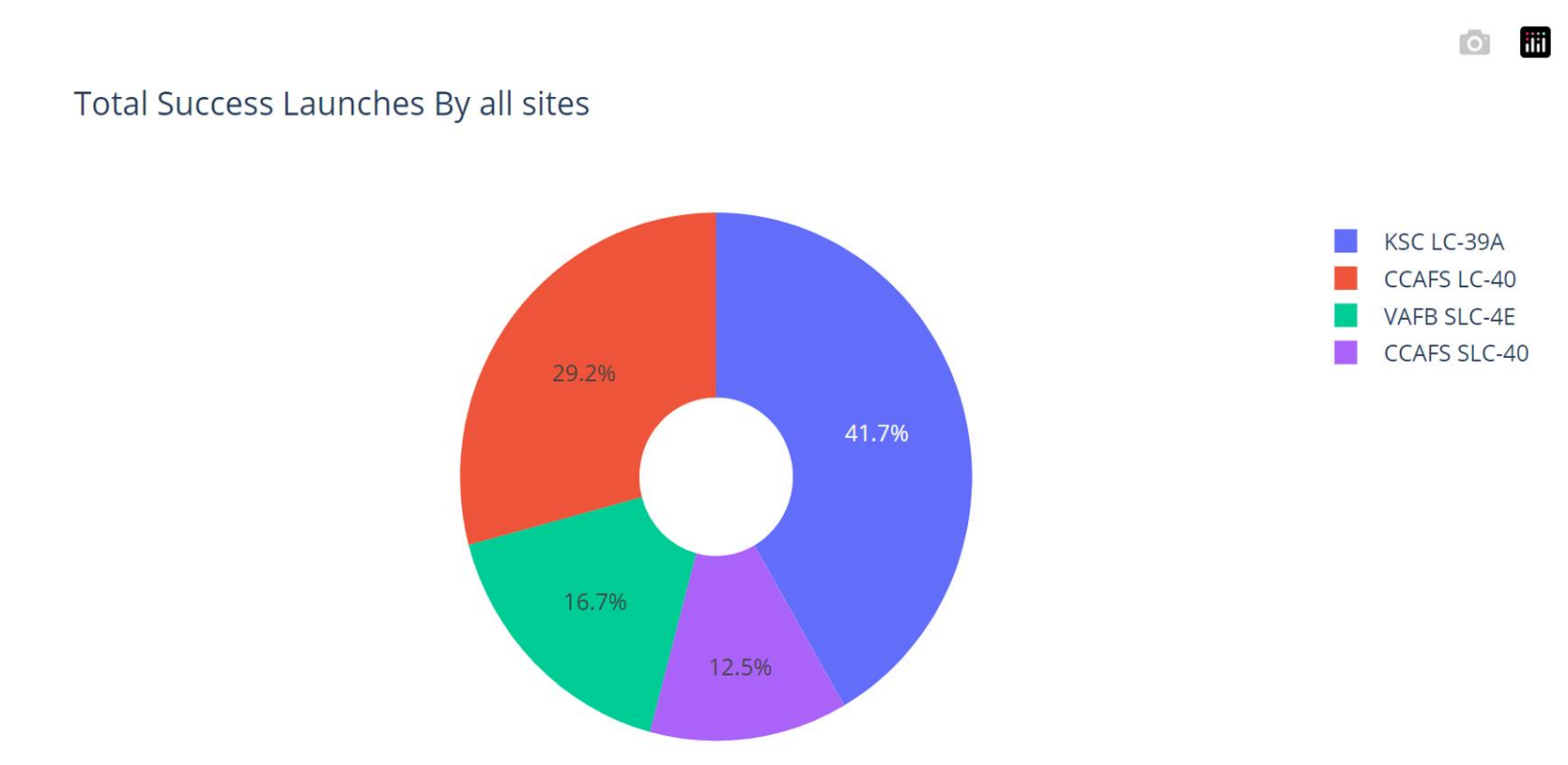
Distance to Highway




Section 4
Build a Dashboard with Plotly Dash

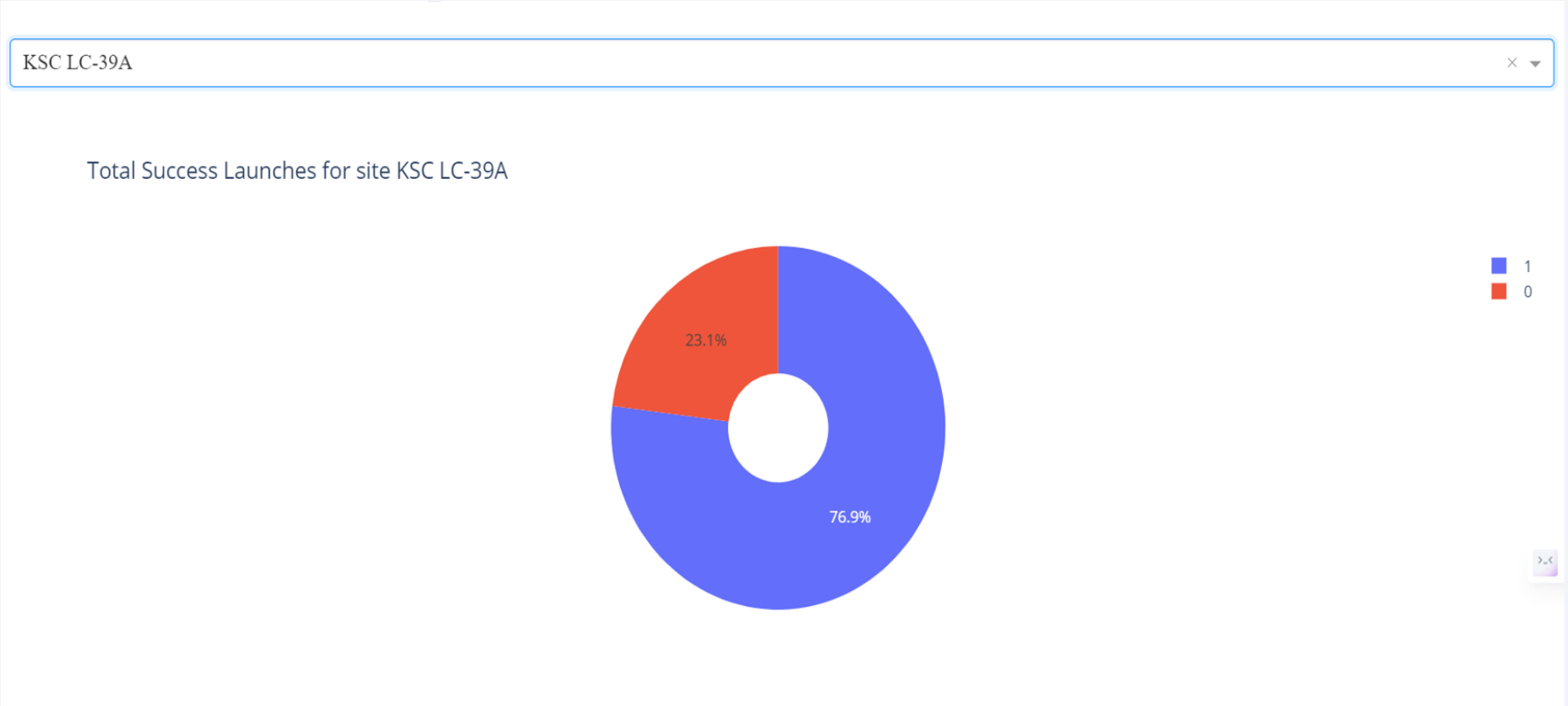
Pie chart for success percentage of all launch site

- KSC LC-39A has the most successful launches from the pie chart



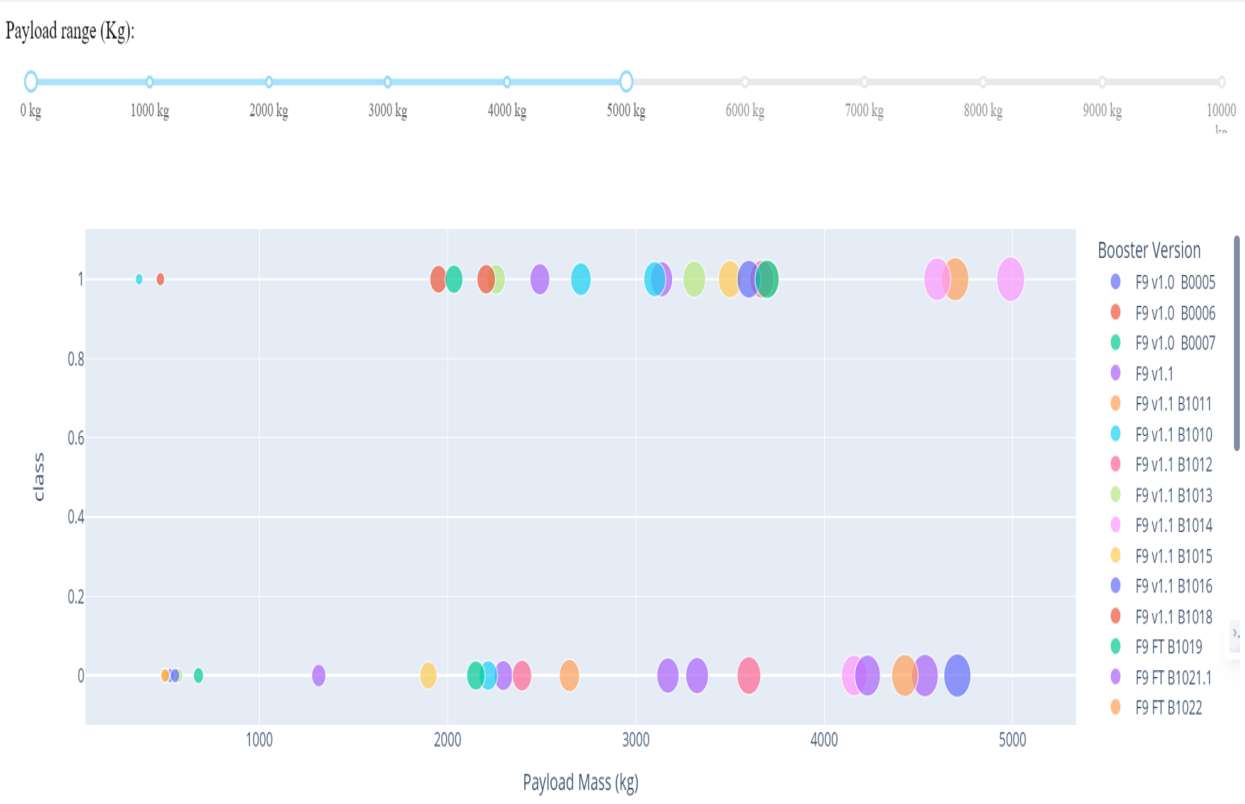
Pie chart showing the Launch site with the highest launch success ratio

- KSC LC-39A has a success rate of 76.9% and failure rate of 23.1%

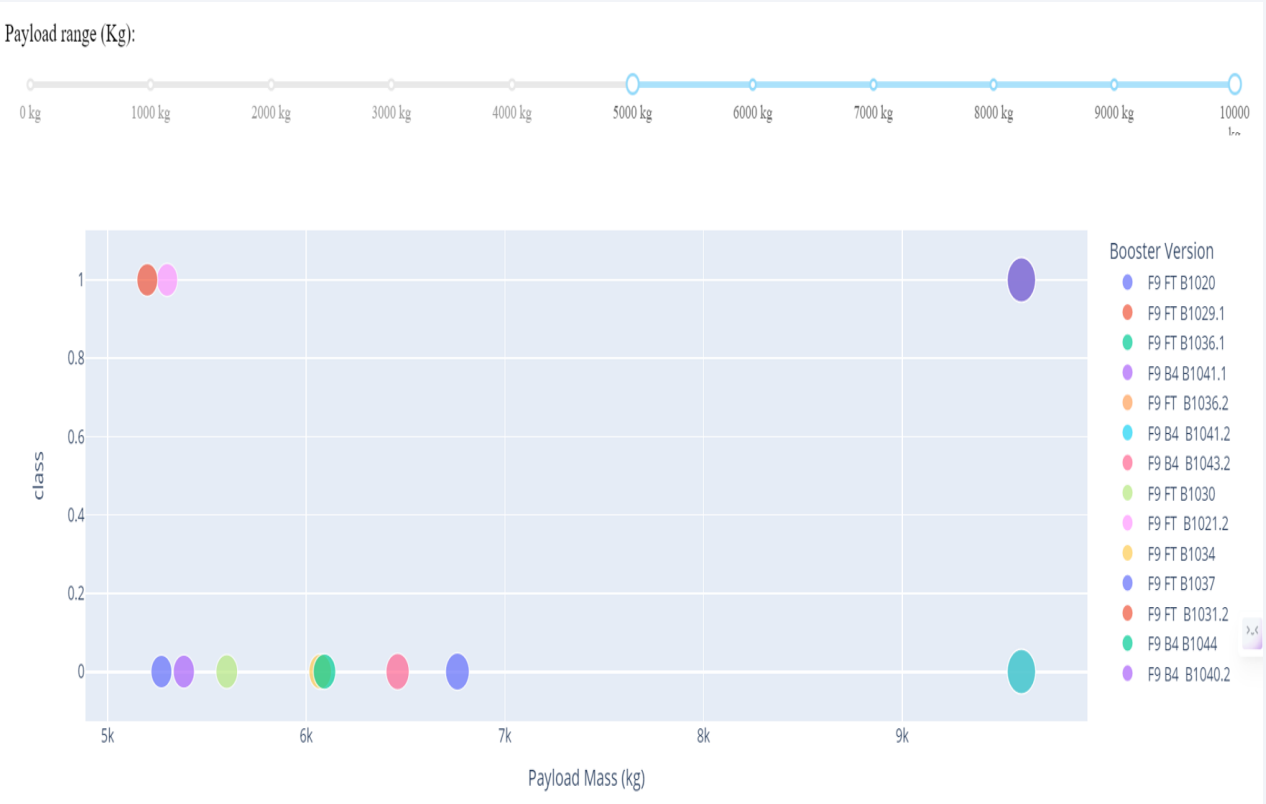


Scatter plot of Payload vs Launch Outcome for all sites, for different payload mass

0kg-5000kg payload mass



5000kg-10000kg payload mass



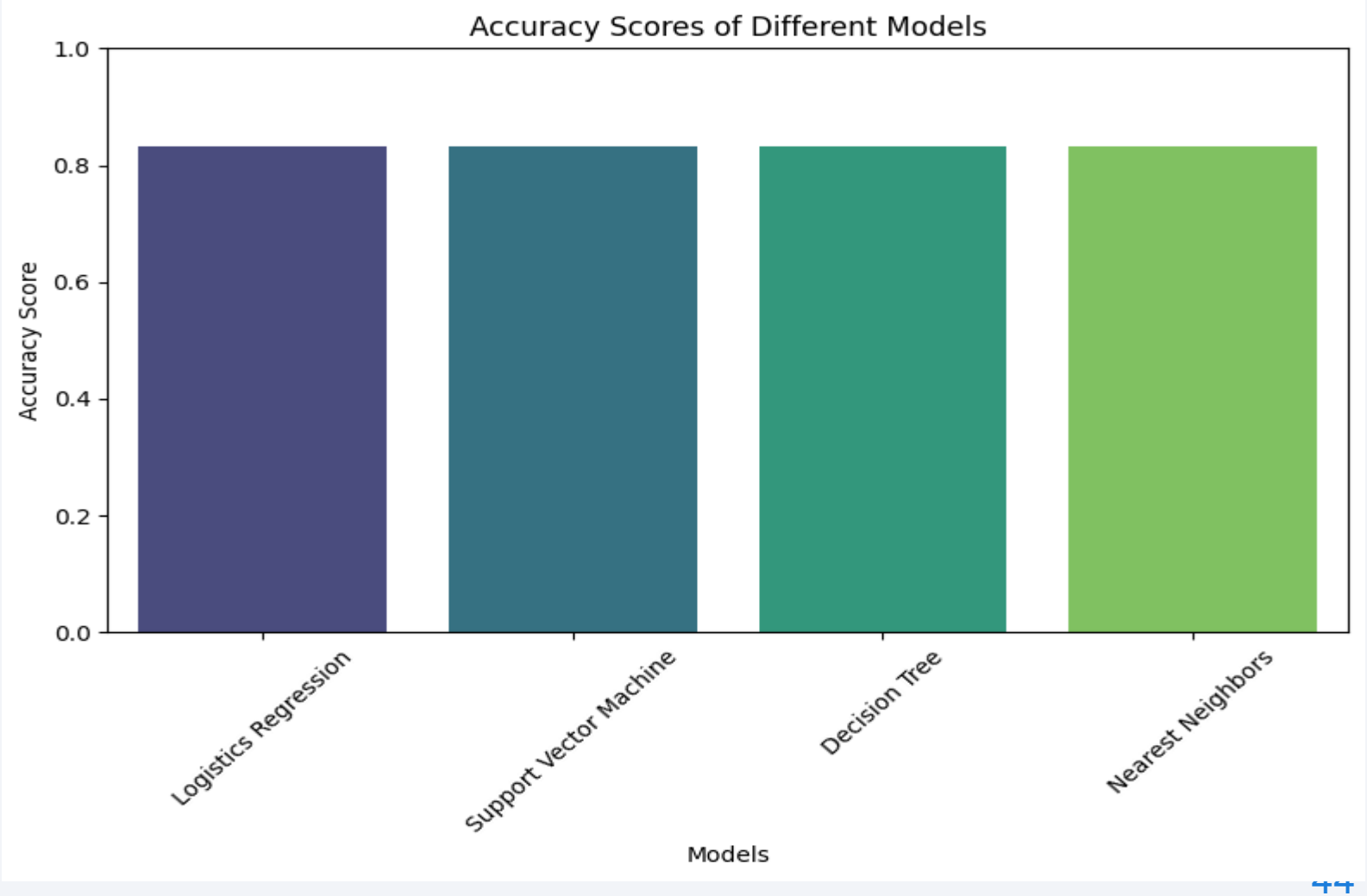
- It can be seen from the scatter plots with varying payloads that higher payload mass has significantly less success rates compared with ones with lower payloads.

Section 5

Predictive Analysis (Classification)

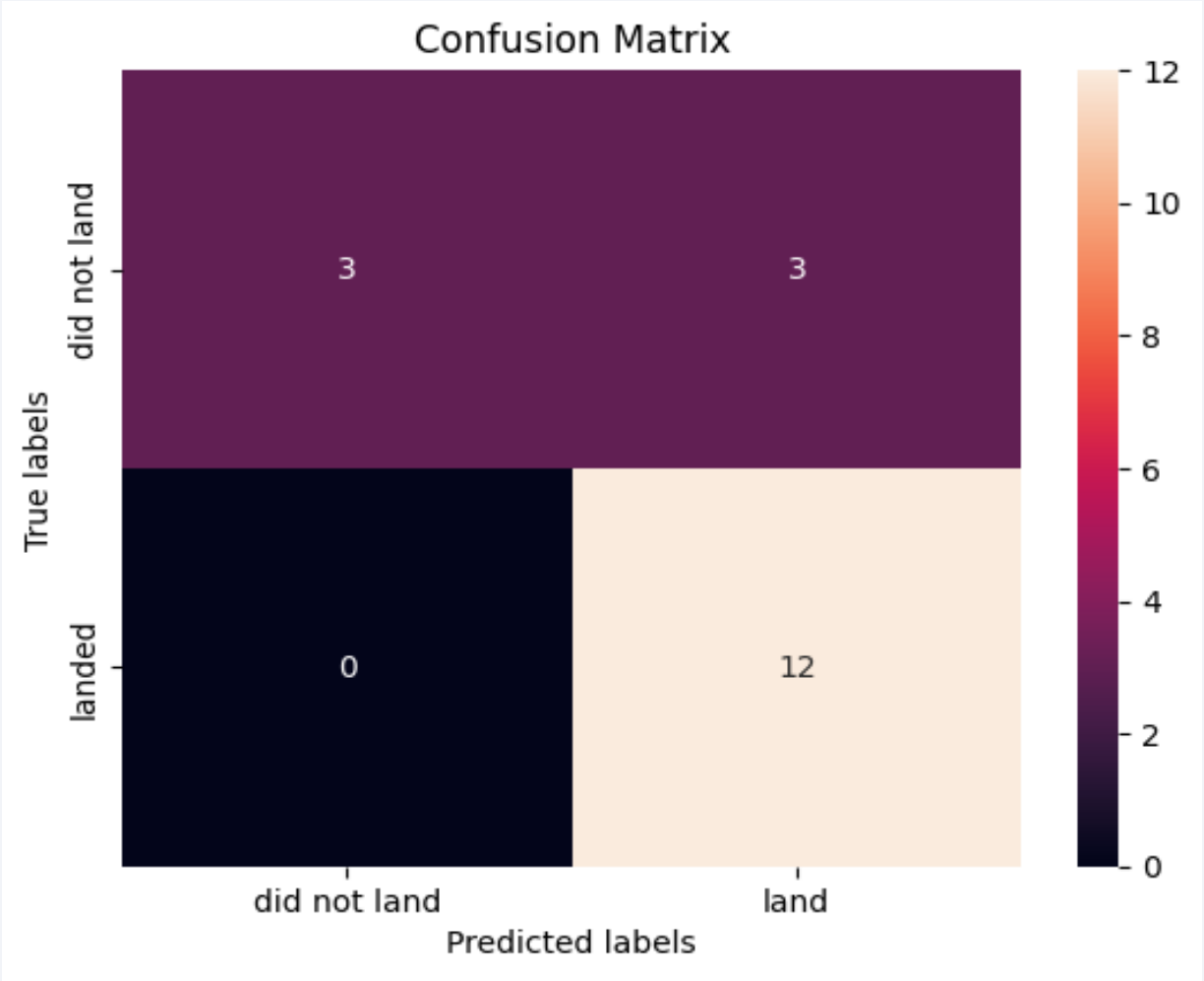
Classification Accuracy

- The SVM, KNN, Decision tree and Logistics Regression models all have equal accuracy of 83.3% on test dataset



Confusion Matrix

- The confusion matrix for the SVM classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing(3 of them) marked as successful landing by the classifier.



Conclusions

We can conclude that:

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites.
- All the classifiers have same test accuracy on the test sample, so any of them could be tagged the “Best model”

Thank you!