IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Freddy A. Saavedra H.
07/31/2024

# Outline

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

# 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
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result

# Introduction

- **Project background and context:**

Space X 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 Space X can reuse the first stage. Therefore, 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 space X for a rocket launch. This goal of the project is to create a machine learning pipeline to predict if the first stage will land successfully.

- Problems you want to find answers

- What factors determine if the rocket will land successfully?
- The interaction amongst various features that determine the success rate of a successful landing.
- What operating conditions needs to be in place to ensure a successful landing program.

Section 1

# Methodology

# Methodology

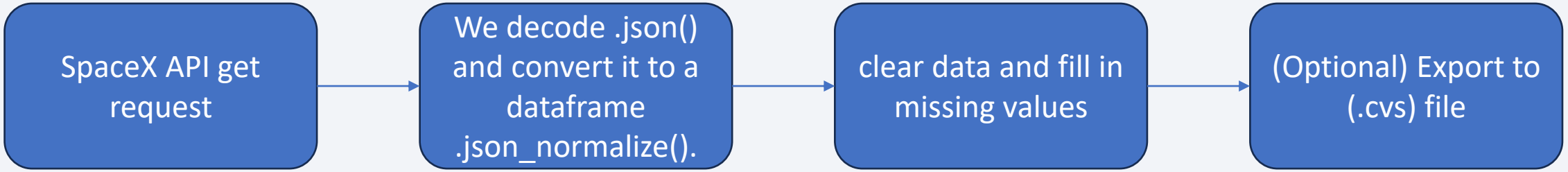
## Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling
  - One hot encoding, drop irrelevant (i.e. uncorrelated) columns
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Scatter/Bar/Pie charts for visual pattern recognition
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using Regression and Tree techniques

# Data Collection

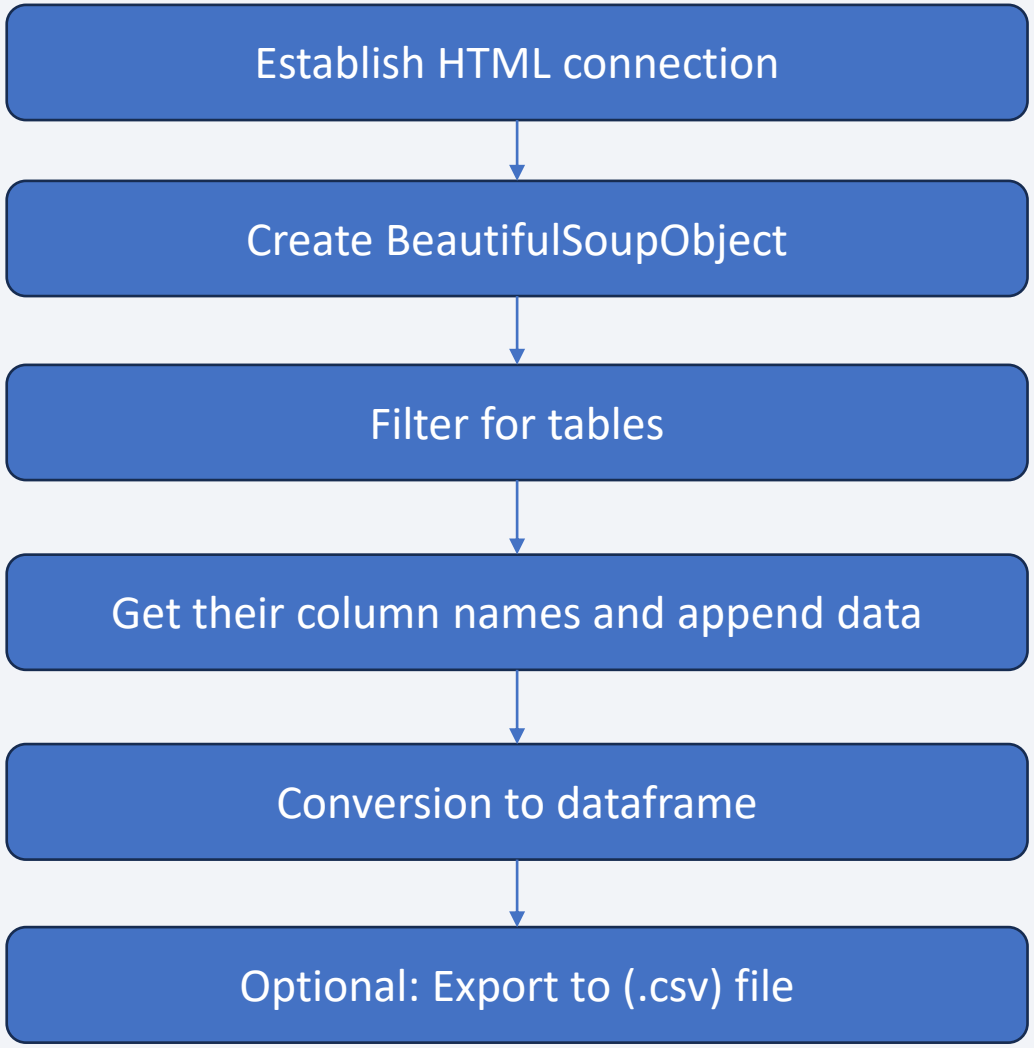
- The data was collected using various methods

# Data Collection – SpaceX API




FILE ON GITHUB

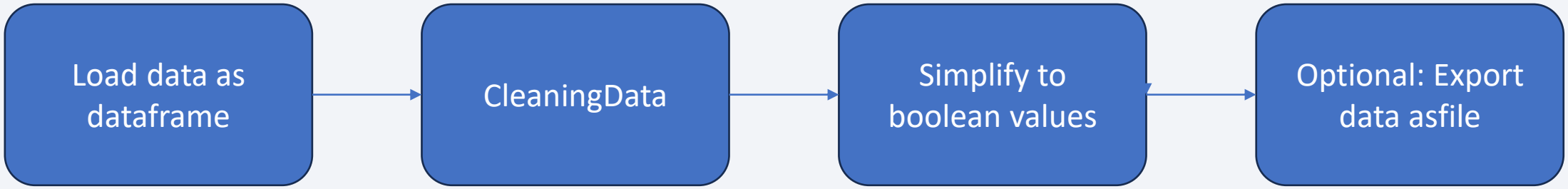
# Data Collection - Scraping




FILE ON GITHUB

# Data Wrangling

- What is to do?
  - Cleaning and polishing possible messy/complex data sets for better handling




File on GitHub

# EDA with Data Visualization

• What is to do?

Create visuals and collection optical insights

• Here:

- Payload mass/Flight number vs Launch site/Orbit type/Flight number as scatterplot for type of dependency
- Success rate vs orbit type as bar chart for impact of variable
- Launch Success vs Year for trend observation

File on GitHub

# EDA with SQL

- What is to do? •
  - Heuristically guessing/querying/questioning in the database what might have happened
- We have questioned as follows:
  - General overview over available landsides, in particular whose five entries who start with 'CCA’
  - Number of successful/failed mission outcome
    - List for failed ones in 2015
    - List first successful landing outcome in drone ship
  - Specify, count and rank outcomes between ~2010 and ~2017
  - Average Payload per booster version 'F9v1.1'/Total for boosters carried by NASA (CRS)
  - Booster version with maximal payload
  - Names of boosters with successful ground pad and certain payload

File on GitHub

# Build an Interactive Map with Folium

- We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- We assigned the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0 for failure, and 1 for success.
- Using the color-labeled marker clusters, we identified which launch sites have relatively high success rate.
- We calculated the distances between a launch site to its proximities. We answered some question for instance:
  - Are launch sites near railways, highways and coastlines.
  - Do launch sites keep certain distance away from cities.

File on GitHub

# Build a Dashboard with Plotly Dash

- What has been done?
  - Selection of Launch Site
  - Pie charts for relation percentage based on launch site
  - Scatter graphs for correlation between payload and success based on launch site
- Why? To visualize the effectiveness of launch site and payload mass

File on GitHub

# Predictive Analysis (Classification)


- Standardize/transform data
- Split into test/training sets
- Using training set initialize different ML algorithm

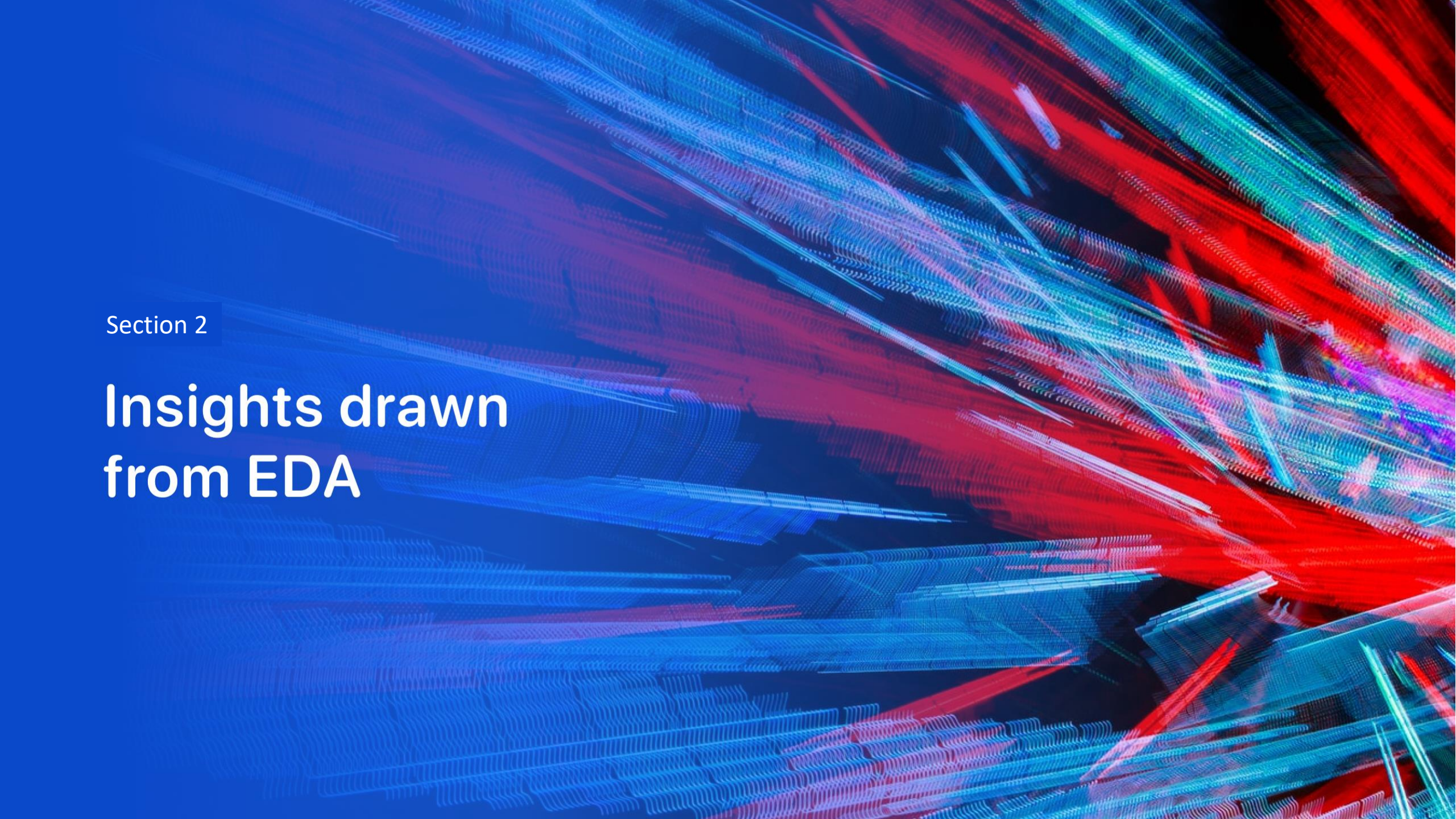
- Check for accuracy via
  - R-score
  - Confusion matrix (true/false vs land./not land.)
- Search for best
  - Score
  - Parameters for ML

- Depending on accuracy: Choose best model!

File on GitHub

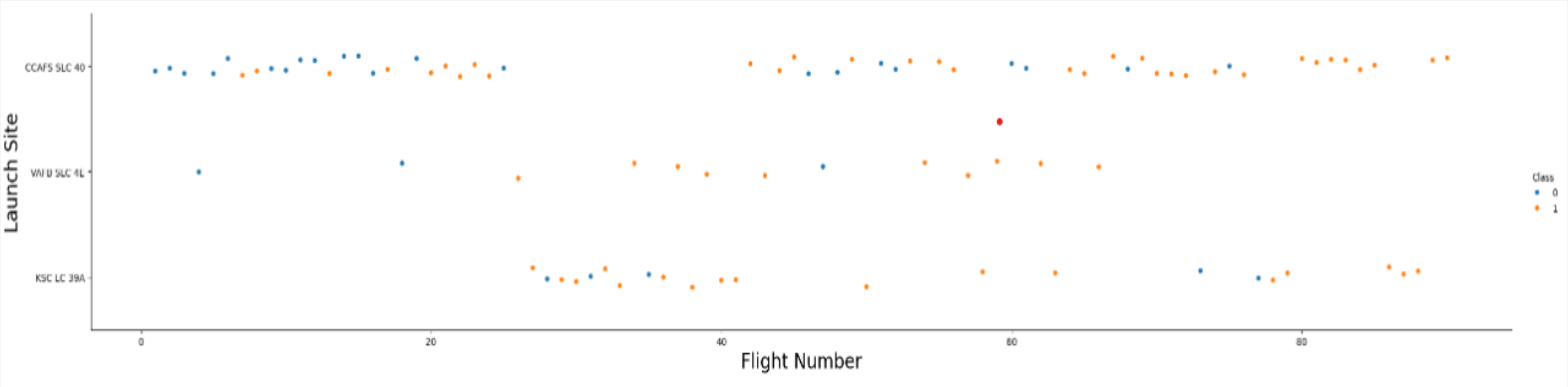
# Results

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


Section 2
Insights drawn from EDA

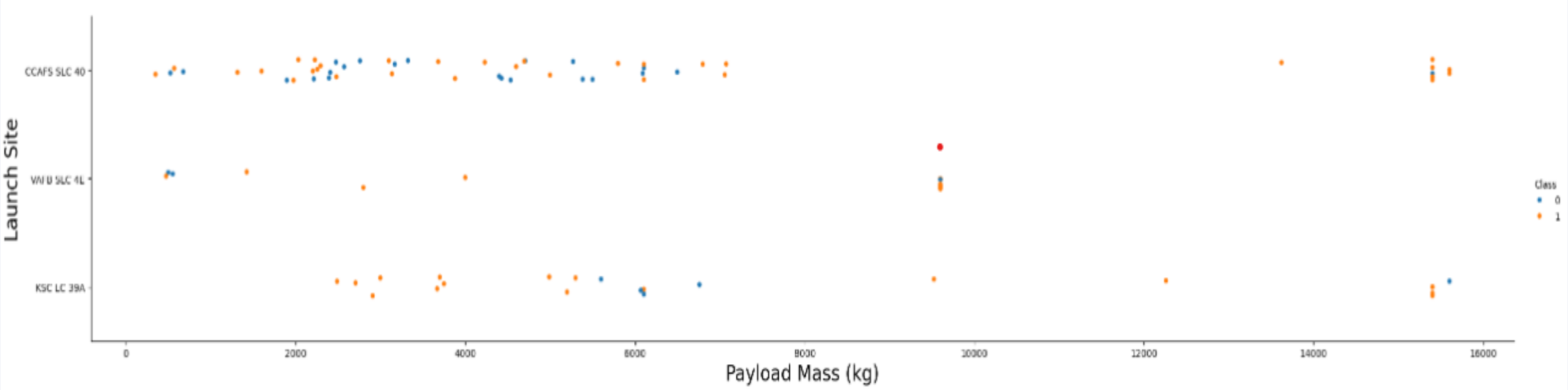
# Flight Number vs. Launch Site

- First ~25 Starts, in particular. at CCAFS SLC 40 were mostly failures
- Starts at KSC LC 39A gave them then information for success
- VAFB SLC 4E with most successful launches (relatively)



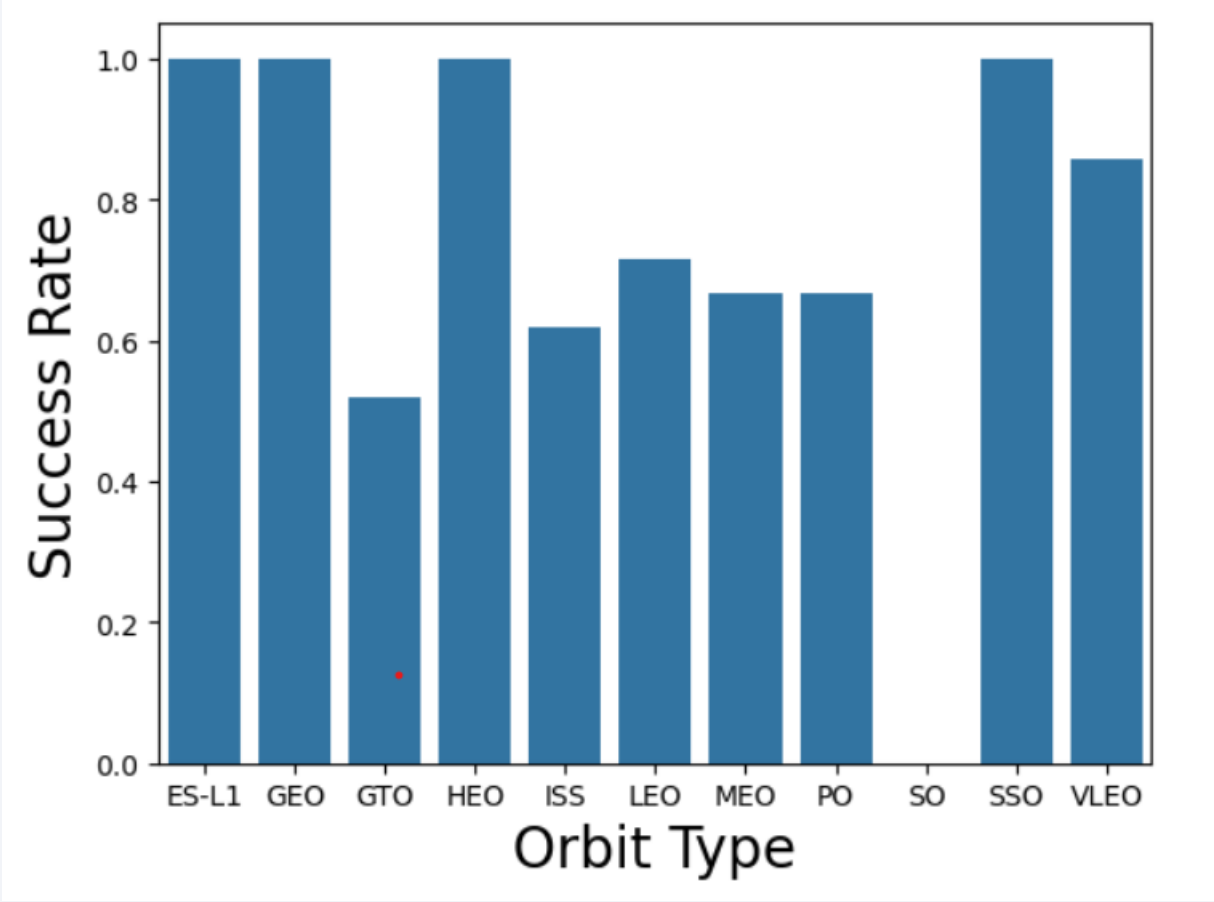
# Payload vs. Launch Site

- After successfully establish a launch with low mass ~7500kg,
- Launches with higher payload masses were almost always successful



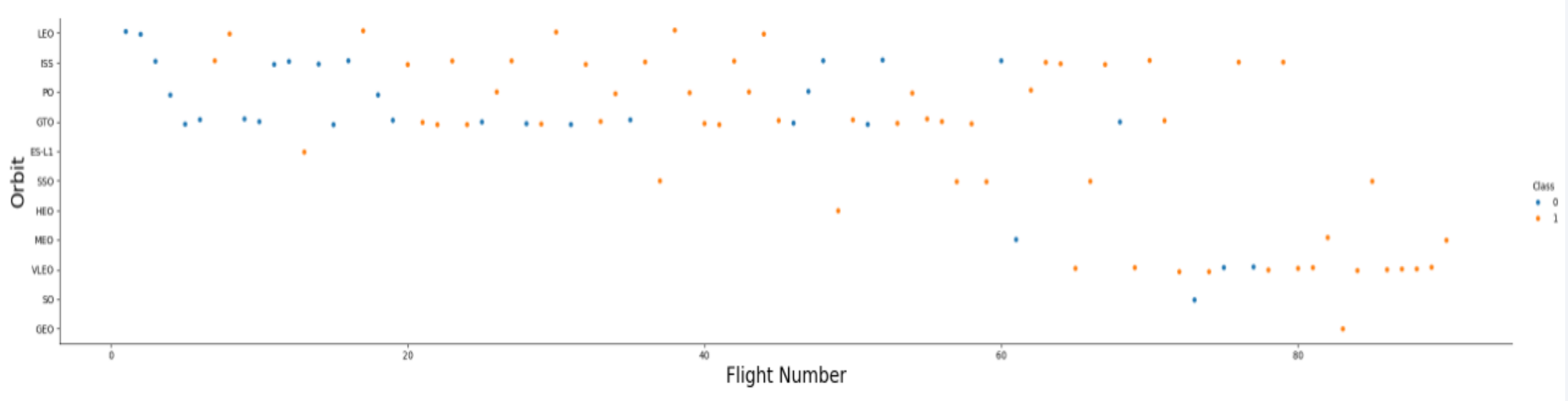
# Success Rate vs. Orbit Type

- Launches onto SO were disastrous (only one launch)
- ES -L1, GEO, HEO and SSO with success only



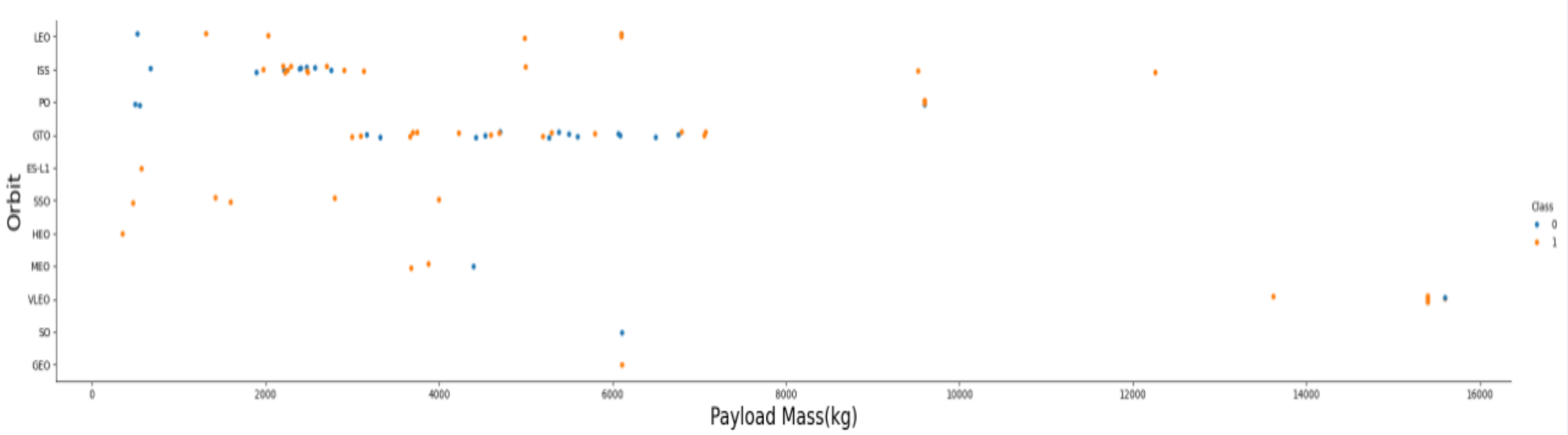
# Flight Number vs. Orbit Type

- At the beginning, testing on four selected orbits
- ISS and GTO gave them a breakthrough after flight number ~ 20
- After flight number ~ 60 also focused on other orbits, in particular on VLEO



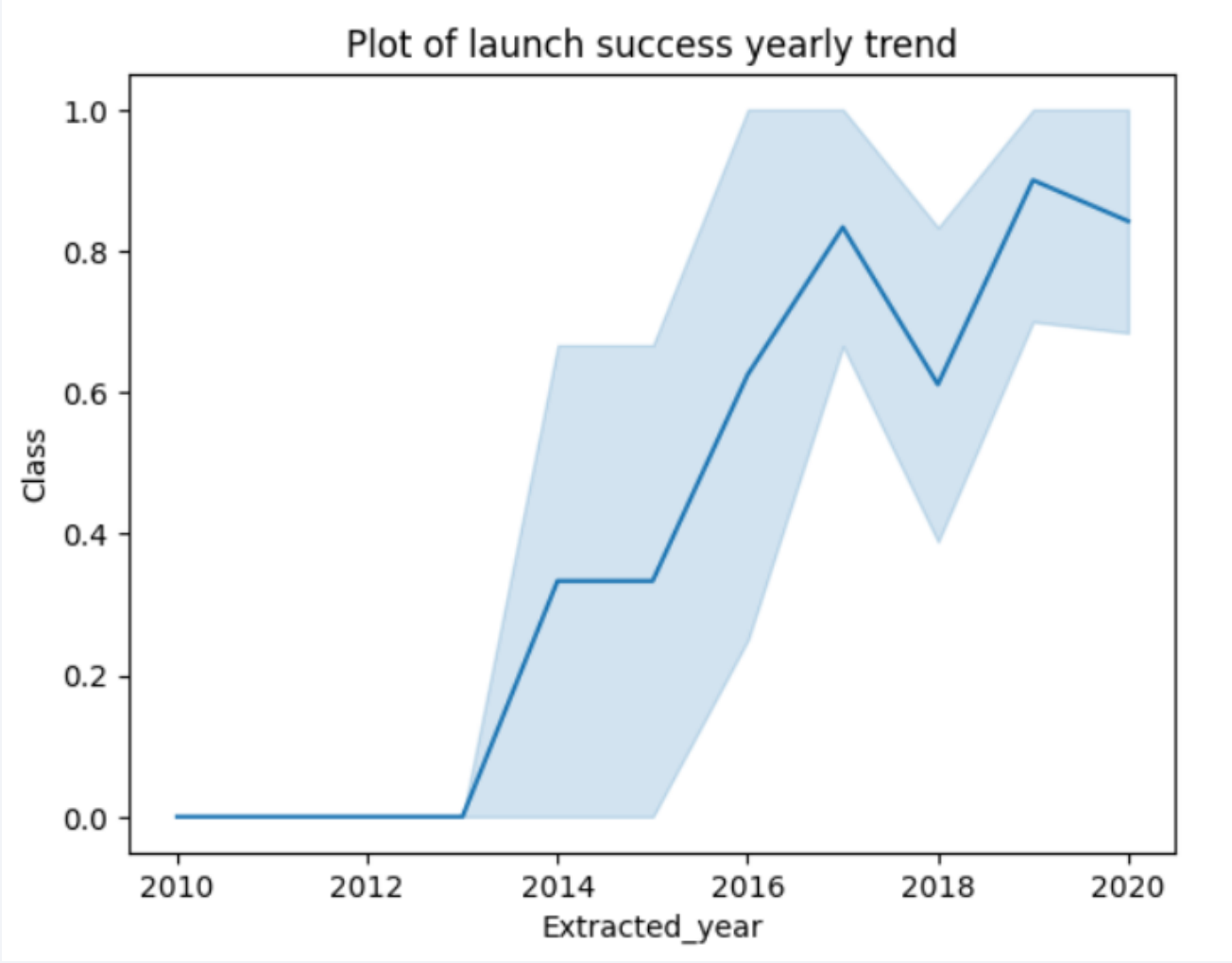
# Payload vs. Orbit Type

- We can observe that with heavy payloads, the successful landing are more for PO, LEO and ISS orbits.



# Launch Success Yearly Trend

- Since 2013 mostly improving
- Since 2017 mostly >80% except for
- Dip in Success in 2018



# All Launch Site Names

- Find the names of the unique launch sites
- Key word DISTINCT does the job

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

```
[13]: %sql select Distinct(LAUNCH_SITE) from SPACEXTBL;
```

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

[13]:

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

# Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Make use of the WHERE keyword for the condition and use % as a wildcard for an arbitrary ending

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

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

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

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|
| 2010-06-04 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success | Failure (parachute) |
| 2010-12-08 | 15:43:00 | F9 v1.0 B0004 | CCAFS LC-40 | Dragon demo flight C1, two CubeSats, barrel of Brouere cheese | 0 | LEO (ISS) | NASA (COTS) NRO | Success | Failure (parachute) |
| 2012-05-22 | 7:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 0:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500 | LEO (ISS) | NASA (CRS) | Success | No attempt |
| 2013-03-01 | 15:10:00 | F9 v1.0 B0007 | CCAFS LC-40 | SpaceX CRS-2 | 677 | LEO (ISS) | NASA (CRS) | Success | No attempt |

# Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Use the function SUM to create a new column and thus, the total payload

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEXTBL WHERE Customer  LIKE 'NASA (CRS)';
```

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

| TOTAL_PAYLOAD |
|---|
| 45596 |

# Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Use the function AVG to create a new column and thus, the desired average

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

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

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

| AVG_PAYLOAD |
| --- |
| 2928.4 |

# First Successful Ground Landing Date

- We observed that the dates of the first successful landing outcome on ground pad was 22nd December 2015

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

*Hint:Use min function*

```
%sql SELECT MIN(DATE) AS FIRST_SUCCESS_GP FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (ground pad)';
```

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

| FIRST_SUCCESS_GP |
| --- |
| 2015-12-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

We 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

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 DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND LANDING_OUTCOME = 'Success (drone ship)'
```

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

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

# Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Using the function COUNT one can count the amount of entries But, we would like to do so with a grouping of result which is done via the GROUP BY keyword

List the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL GROUP BY MISSION_OUTCOME ORDER BY MISSION_OUTCOME;
```

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

| Mission_Outcome | QTY |
|---|---|
| Failure (in flight) | 1 |
| Success | 98 |
| Success | 1 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

Used a second query to find the maximal payload in which we used the MAX function. With this information, we can employ it in the condition part.

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

```
%sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL) ORDER BY BOOSTER_
```

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

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

# 2015 Launch Records

Here we use a specialty of SQLite, the substr function, i.e. it gives a substring based on an index and some length. We apply it on the date variable to extract year and month

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

**Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.**

```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Failure (drone ship)' AND substr(Date,0,5)='2015';
```

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

| Booster_Version | Launch_Site |
|---|---|
| F9 v1.1 B1012 | CCAFS LC-40 |
| F9 v1.1 B1015 | CCAFS LC-40 |

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

Used the DESC keyword to get a descending list which is ordered by the amount of landing outcomes while satisfying all other conditions

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order. ¶

```
%sql SELECT LANDING_OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING_OUTCOME ORDER BY
```

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

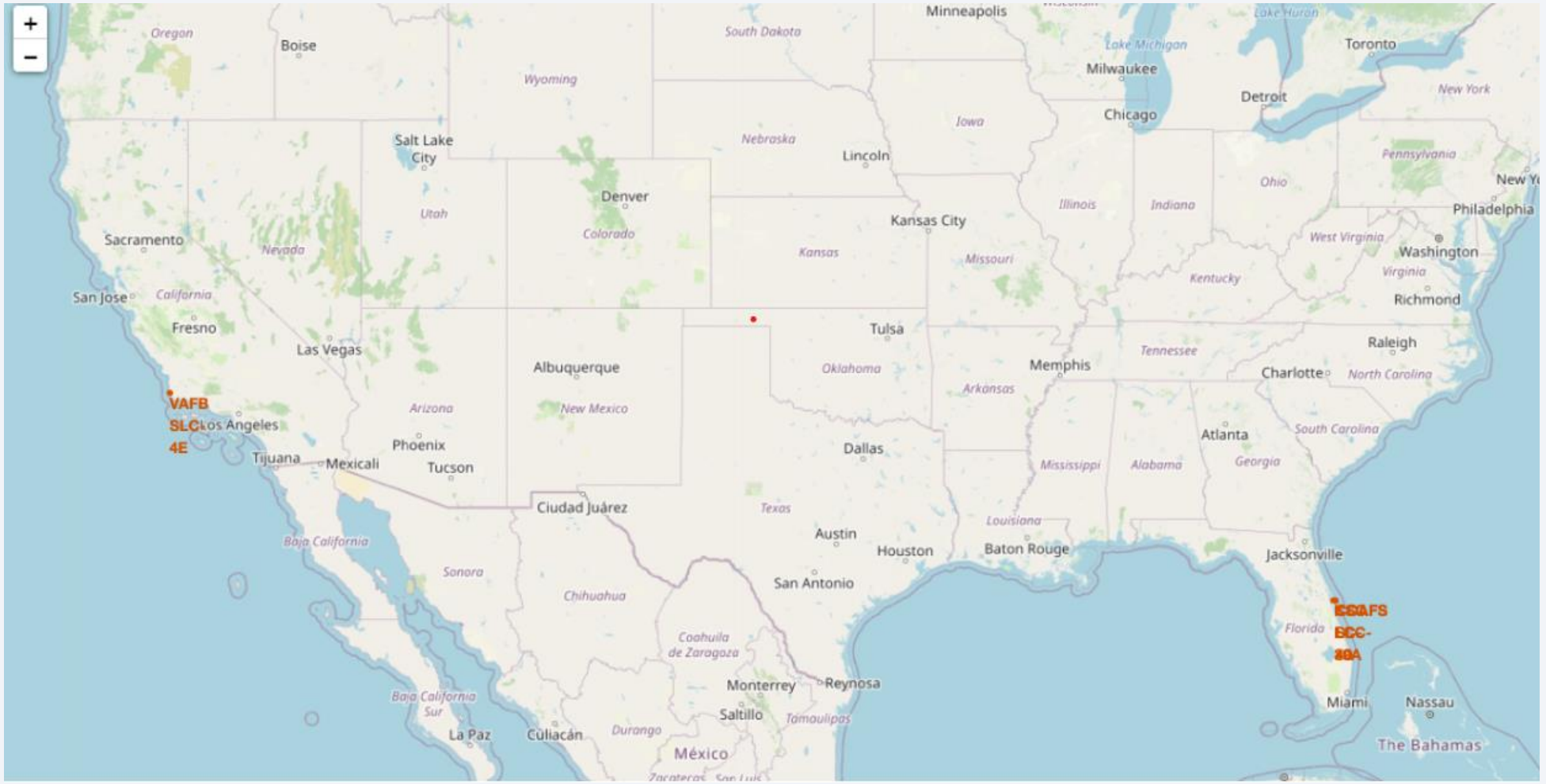
| Landing_Outcome | QTY |
|---|---|
| No attempt | 10 |
| Success (drone ship) | 5 |
| Failure (drone ship) | 5 |
| Success (ground pad) | 3 |
| Controlled (ocean) | 3 |
| Uncontrolled (ocean) | 2 |
| Failure (parachute) | 2 |
| Precluded (drone ship) | 1 |

Section 3

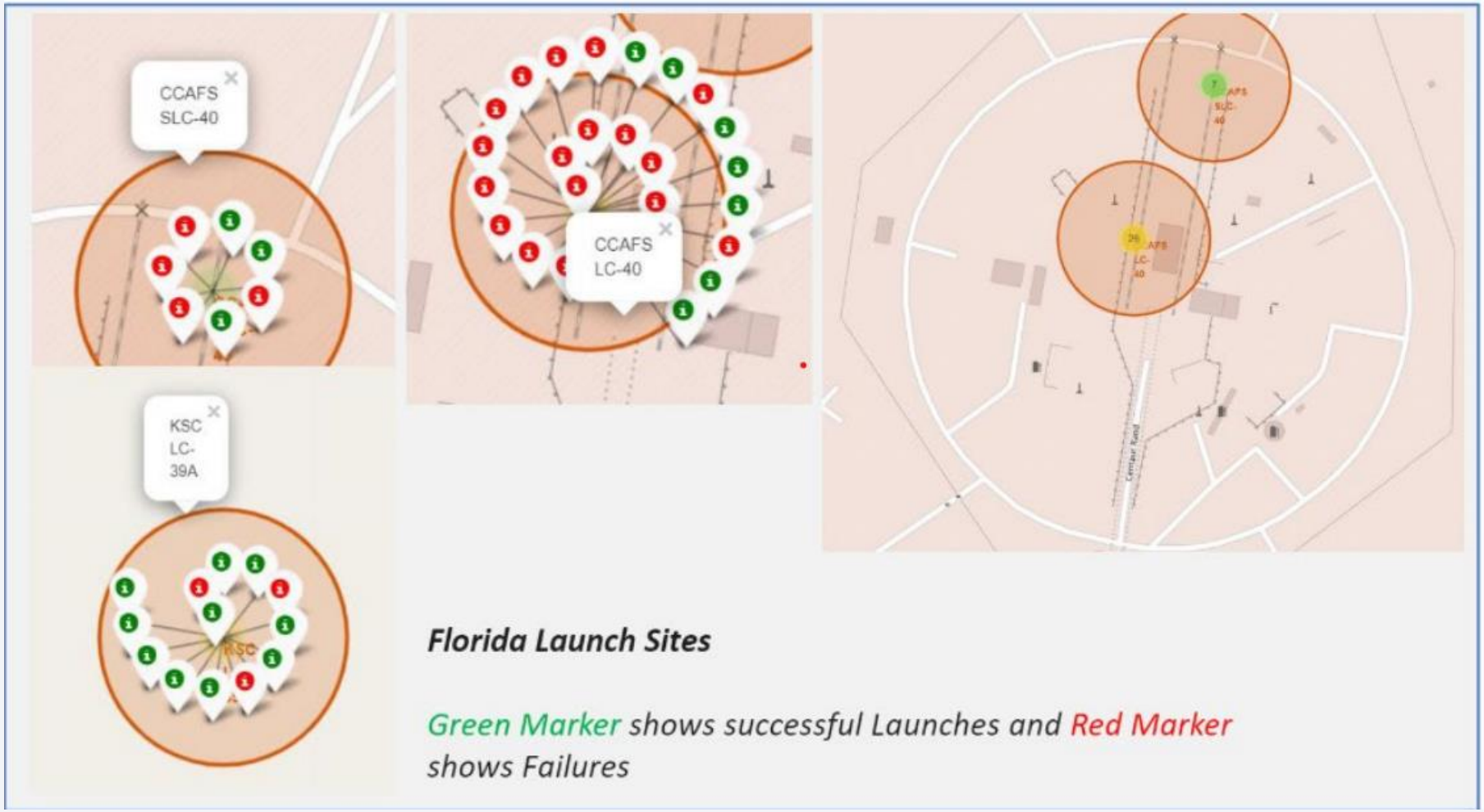
# Launch Sites Proximities Analysis

# Launch Sites on Map

- We can see that the spaceX launch sites are in the USA of American coasts. Florida and California.



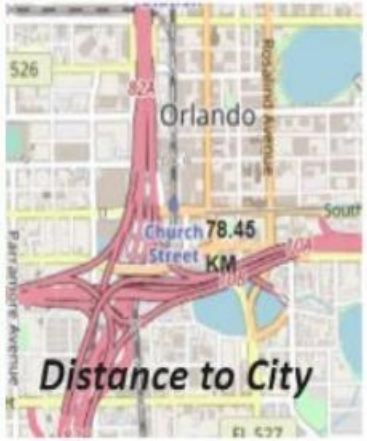
# Markers showing launch sites with color labels



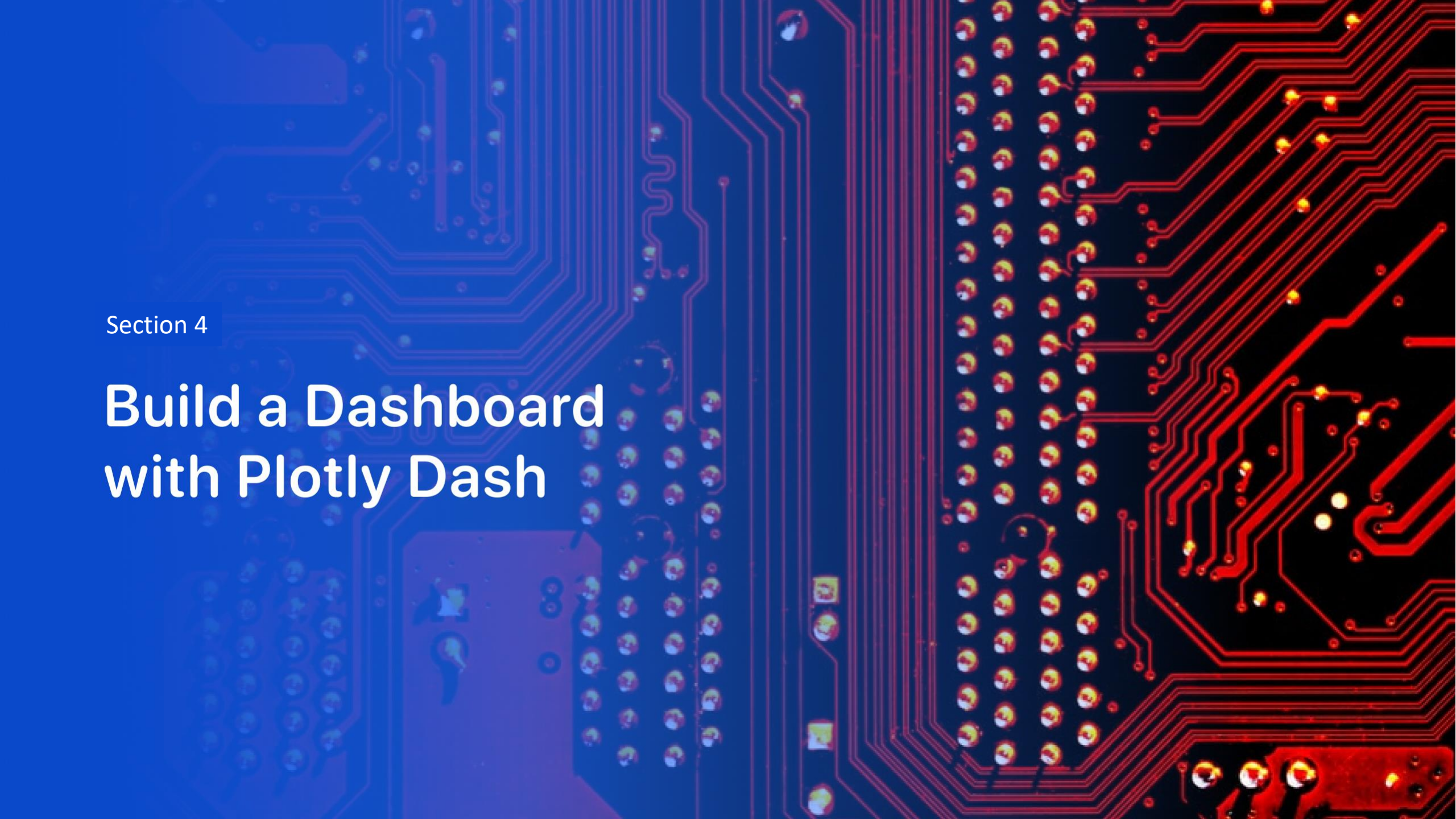

# Launch Site distance to landmarks





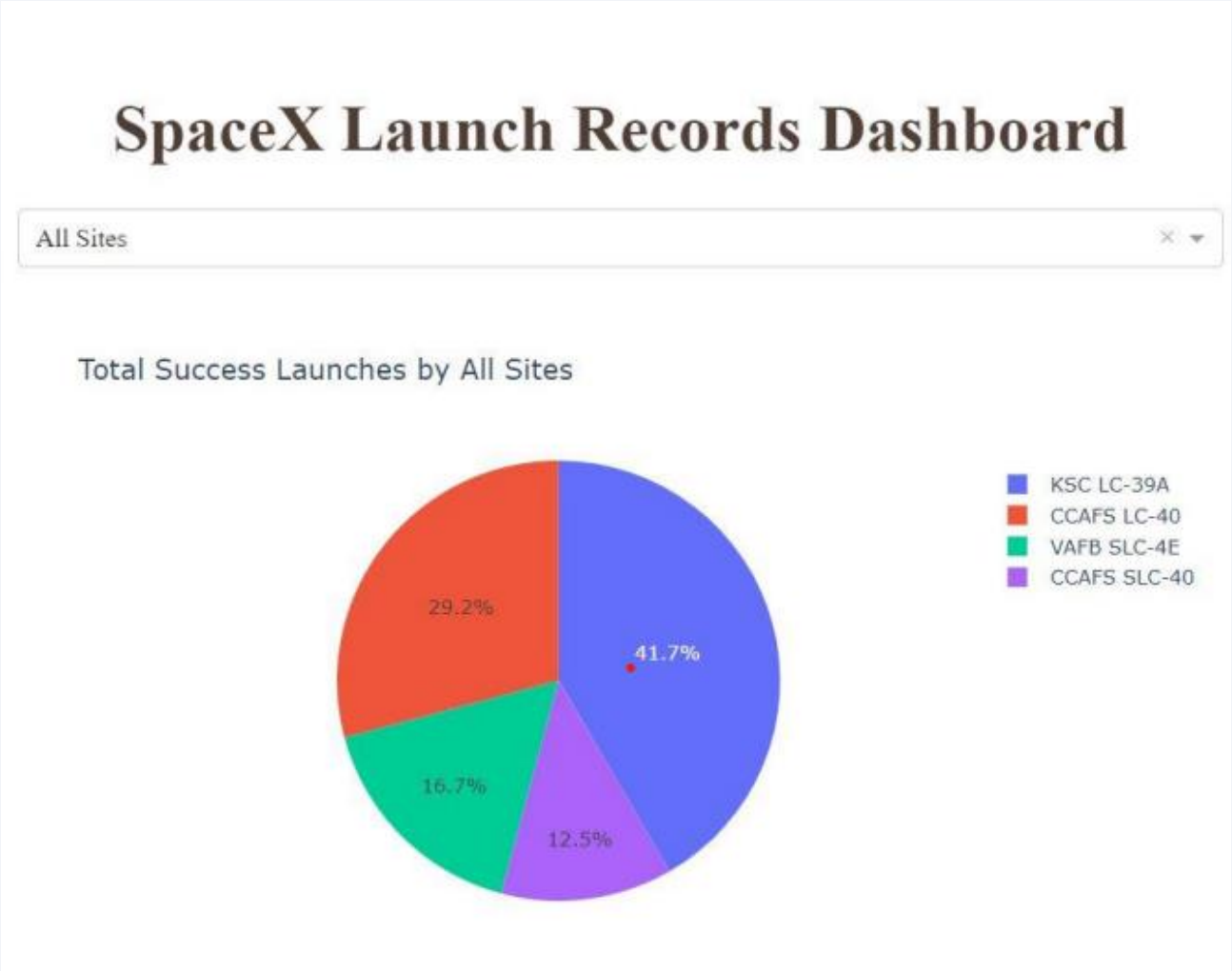



- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes


Section 4
Build a Dashboard with Plotly Dash

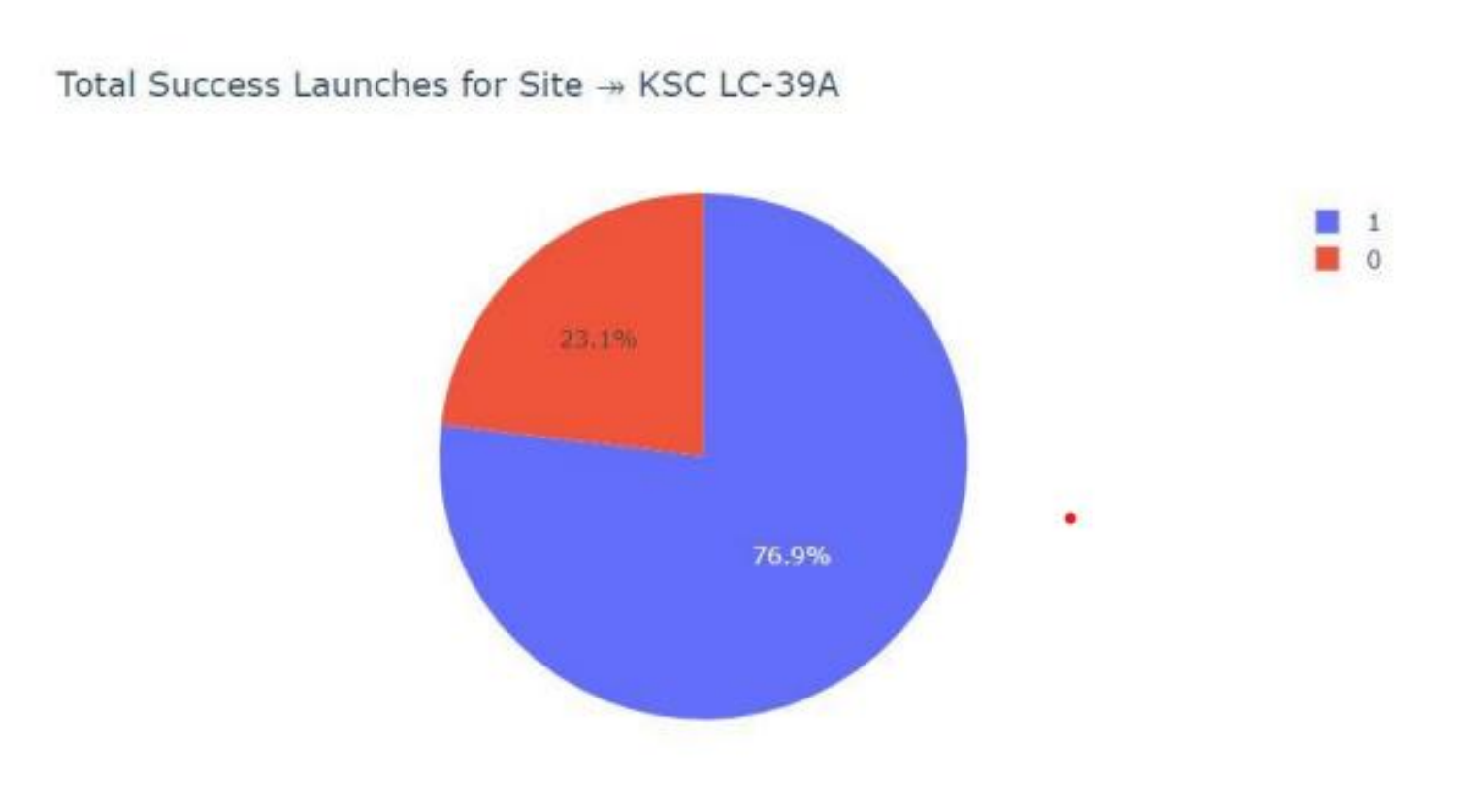
# Launch Success: All Sites

- KSC-LC-39A has highest share of success
- CCAFS.SLC-40 has lowest share of success



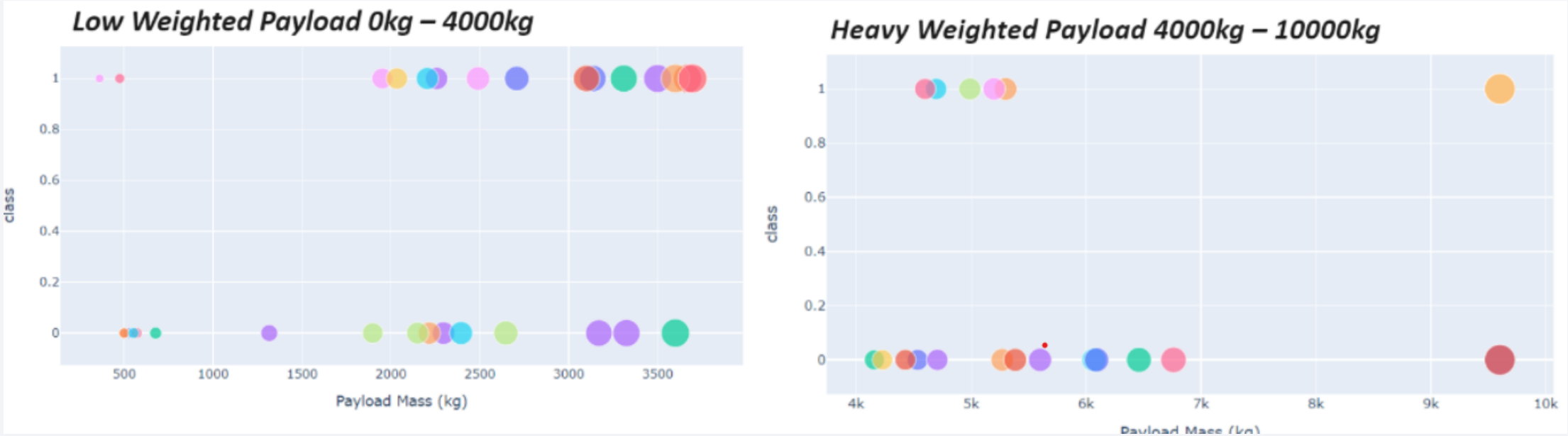
## Pie chart showing the Launch site with the highest launch success ratio

- KSL LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate.



# Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider

- We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

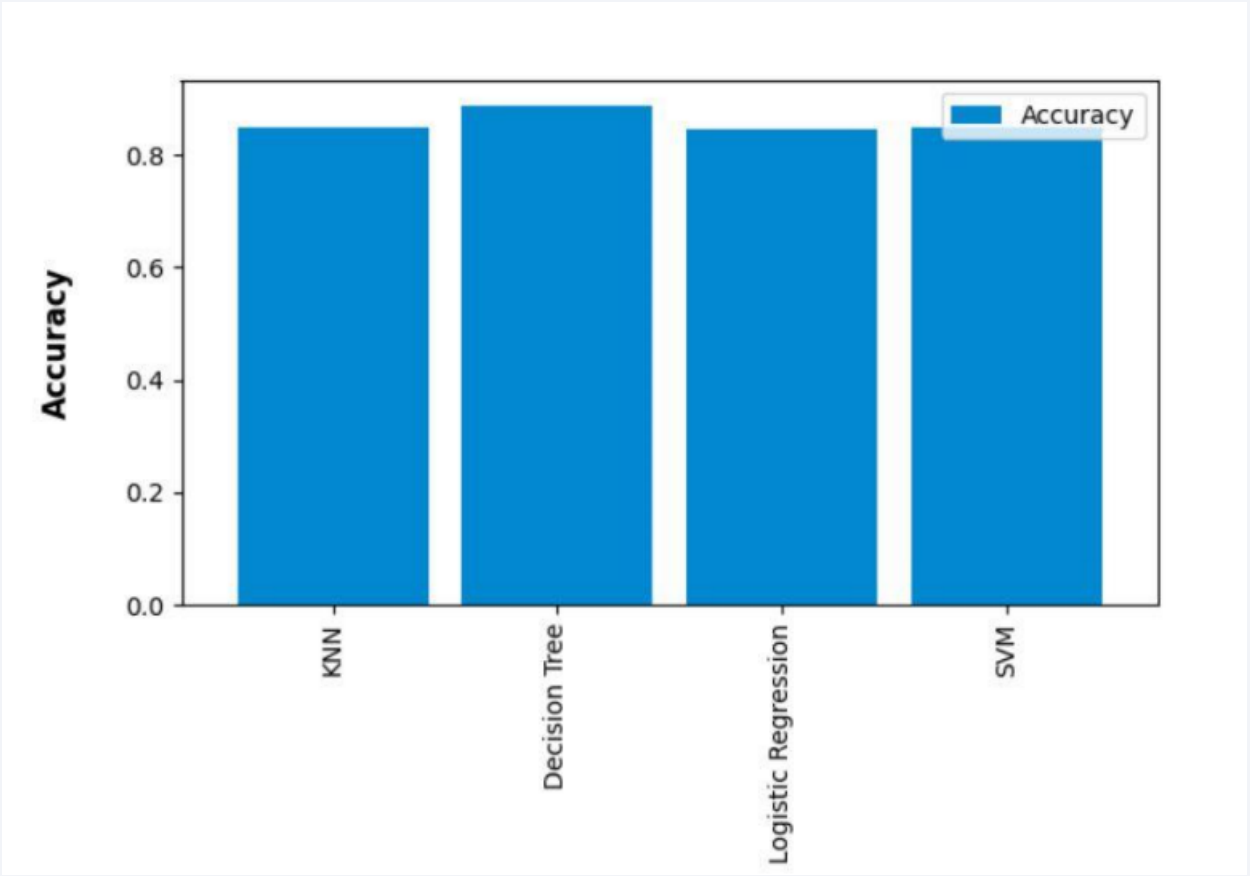


Section 5

# Predictive Analysis (Classification)

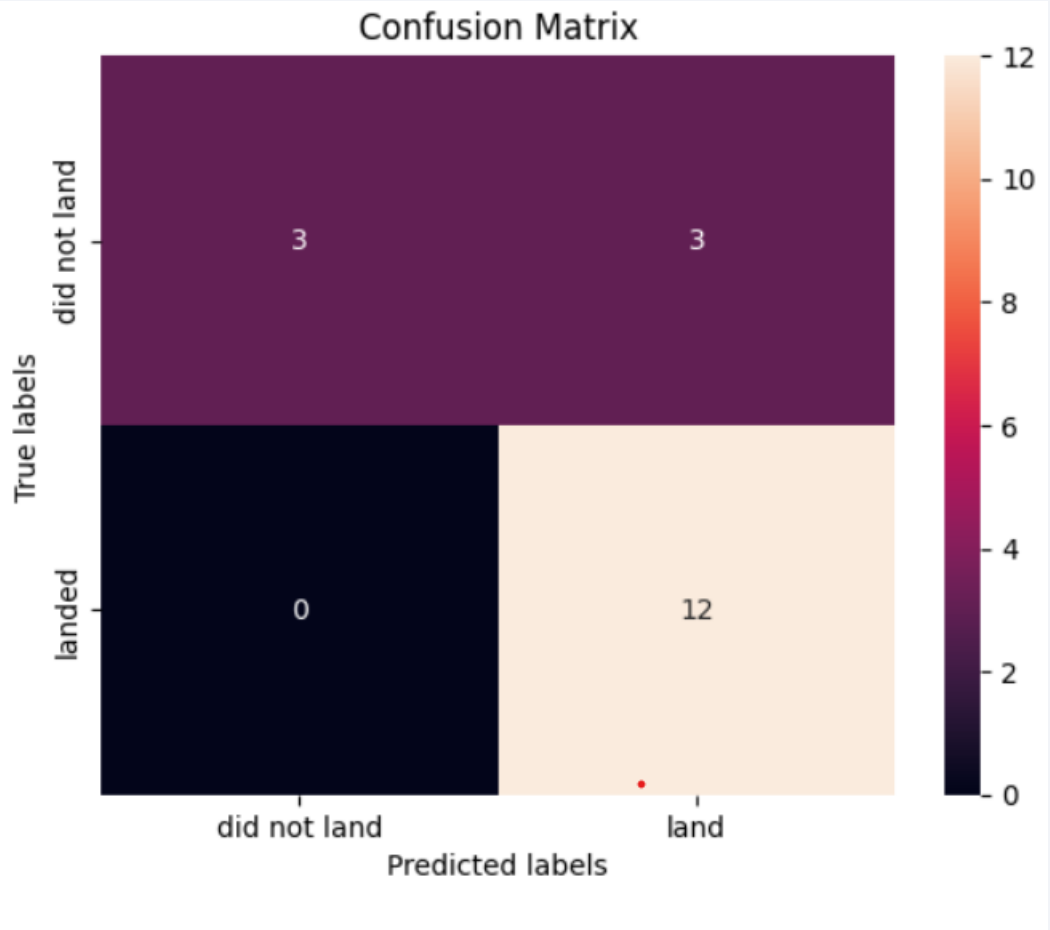
# Classification Accuracy

- Decision Tree Model is the best with an accuracy with arround 88%
- Others perform only minorly worse with accuracy rates > 84%



# Confusion Matrix for Decision Tree

- True/false vs Land/Not Land Matrix
- Predicted landing outcomes for the test data=subset of original data
- Unfortunately, we have True/Not -Land outcomes
- But, overall 15/18 correct predictions



# 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.
- The Decision tree classifier is the best machine learning algorithm for this task.

# Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

# Thank you!