Winning Space Race with Data Science

Axel Mukwena
13 September 2022

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

The Falcon 9 is a reusable two-stage rocket designed and manufactured by SpaceX to reliably and safely transport people and payloads to Earth orbit and beyond. The Falcon 9 is the world's first reusable orbital-class rocket. Reusability allows SpaceX to remanufacture the most expensive parts of the rocket, which in turn reduces the cost of access to space. Therefore, if we can determine if the first stage will land, we can determine the cost of a 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

The effect each relationshop with certain rocket variables will impact determining the success rate of a successful landing, and what conditions does SpaceX have to achieve to get the best results

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceXAPI(https://api.spacexdata.com/v4/rockets/)
  - WebScraping
- Perform data wrangling
  - Identified missing values
  - Replace missing values with mean
  - Identifies Null Values
  - Created a training label class.

Methodology

Executive Summary

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
- Data that was collected until this step were normalized, divided in training and test data sets and evaluated by four different classification models, being the accuracy of each model evaluated using different combinations of parameters.

Data Collection

- Datasets collected from:
  - SpaceX API (https://api.spacexdata.com/v4/rockets/)
  - Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches), using web scraping.

Data Collection – SpaceX API

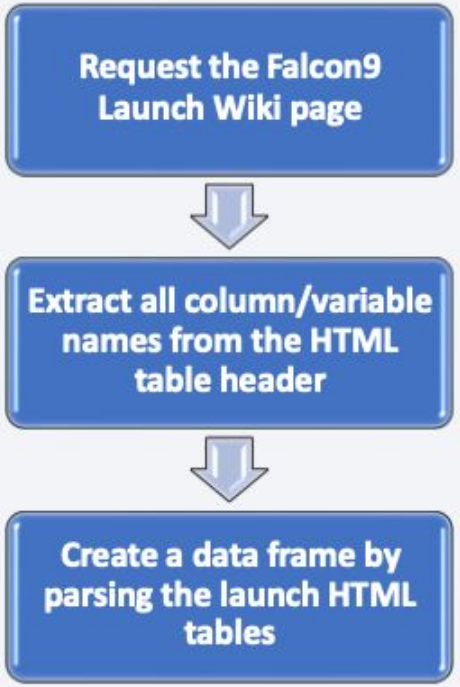
- Using a SpaceX public API, like flowchart:
- Source code:

  https://github.com/heliospjunior/IBM-Data-Science-Capstone-SpaceX/blob/main/spacex-data-collection-api.ipynb

Data Collection - Scraping

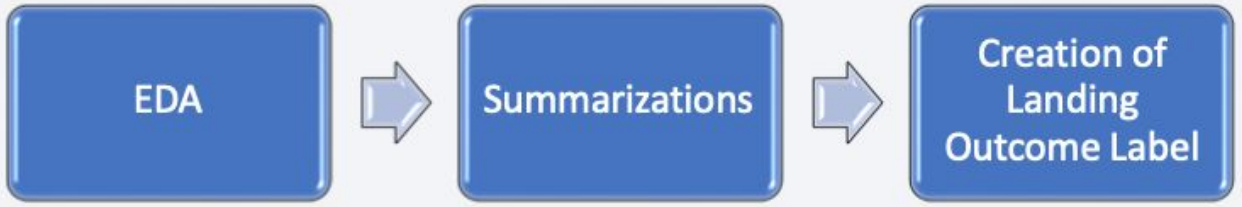
- Data from SpaceX launches obtained from Wikipedia, used like flowchart:.
- Source code:

https://github.com/heliospjunior/IBM-Data-Science-Capstone-SpaceX/blob/main/webscraping.ipynb



Data Wrangling

- Collect data, analysis and check missing data and data types,clean and format data types, change data types if necessary, summary launches per site, and outcome label created from outcome product.



- Source code:

  https://github.com/heliospjunior/IBM-Data-Science-Capstone-SpaceX/blob/main/spacex-Data%20wrangling.ipynb

EDA with Data Visualization

Scatterplots and bar plots were used to visualize the relationship between pair of features:

- Payload Mass X Flight Number,
- Launch Site X Flight Number
- Launch Site X Payload Mass
- Orbit and Flight Number
- Payload and Orbit

Bar plots were used to visualize the relationship between pair of features:

- Mean X Orbit

Line Graphs were used to visualize the relationship between pair of features:

- Mean X Orbit

EDA with SQL

- SQL queries performed:
  - Display the names of the unique launch sites in the space mission;
  - Display the top 5 launch sites whose name begins with the string 'CCA';
  - Display total pay load mass carried by boosters launched by NASA (CRS);
  - Average payload mass carried by booster version F9 v1.1;
  - Date of the first successful landing outcome in ground pad;
  - Listing names of the boosters which have success in drone ship with payload mass between 4000-6000 kg;
  - Total number of successful and failure mission outcomes;
  - Listing names of the booster versions which have carried the maximum payload mass;
  - Failed landing out comes in droneship, their booster versions, and launch site names for in year 2015; and
  - Rank of the count of landing outcomes (such as Failure (droneship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20.
- Source code: https://github.com/heliospjunior/IBM-Data-Science-Capstone-SpaceX/blob/main/eda-sql.ipynb

Build an Interactive Map with Folium

- Launch success rate may depend on the location and proximity of a launch site
- Markers, circles, lines and marker clusters were used with Folium Interative Maps
- Using Folium library, identified all SpaceX launch sites on a map
- Marker clusters indicates groups of events in each coordinate, like launches in a launch site; and lines are used to indicate distances between two coordinates.

- Source code: https://github.com/heliospjunior/IBM-Data-Science-Capstone-SpaceX/blob/main/launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Build an interactive dashboard with Plotly Dash including:

- Dropdown menu for select sites
- Pie charts displaying success rate.
- Scatter graph with the relationship about Outcome and Payload Mass (Kg) for the different booster version.
- Range Slider for select range of payload mass (Kg)

Predictive Analysis (Classification)

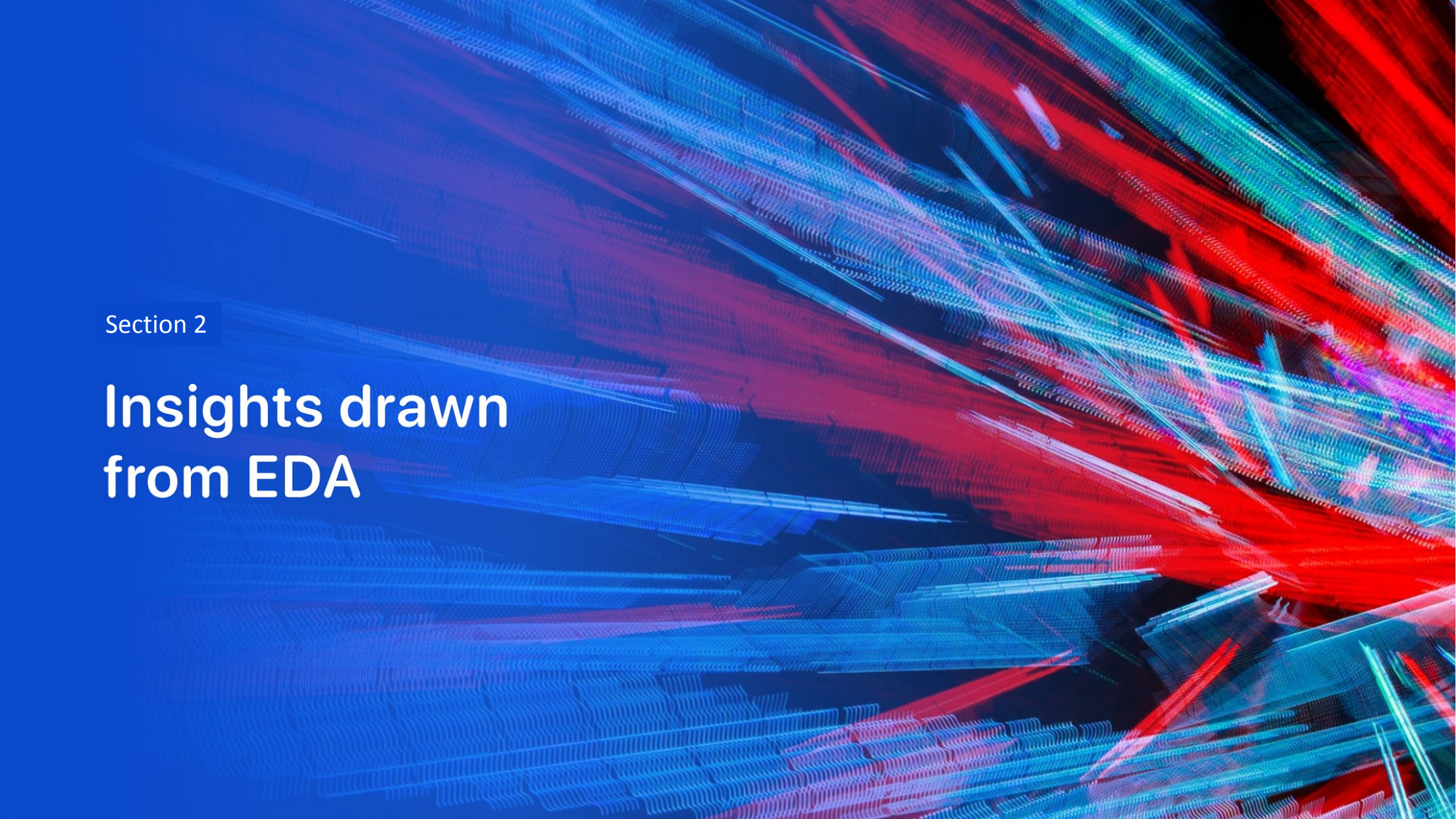
- EDA using numpy and pandas, transforme data and split our data into training and testing.
- Built some machine learning models and tune different hyperparameters using GridSearchCV.
- Used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- Found the best performing classification model.

Results

- Exploratory data analysis results:
  - Space X uses 4 different launch sites;
  - The first launches were done to Space X itself and NASA;
  - The average payload of F9 v1.1 booster is 2,928 kg;
  - The first success landing outcome happened in 2015 fiver year after the first launch;
  - Many Falcon 9 booster versions were successful at landing in drone ships having payload above the average;
  - Almost 100% of mission outcomes were successful;
  - The number of landing outcomes became as better as years passed.
  - Using interactive analytics was possible to identify that launch sites use to be in safety places, near sea, for example and have a good logistic infrastructure around.
  - Most launches happens at east cost launch sites.

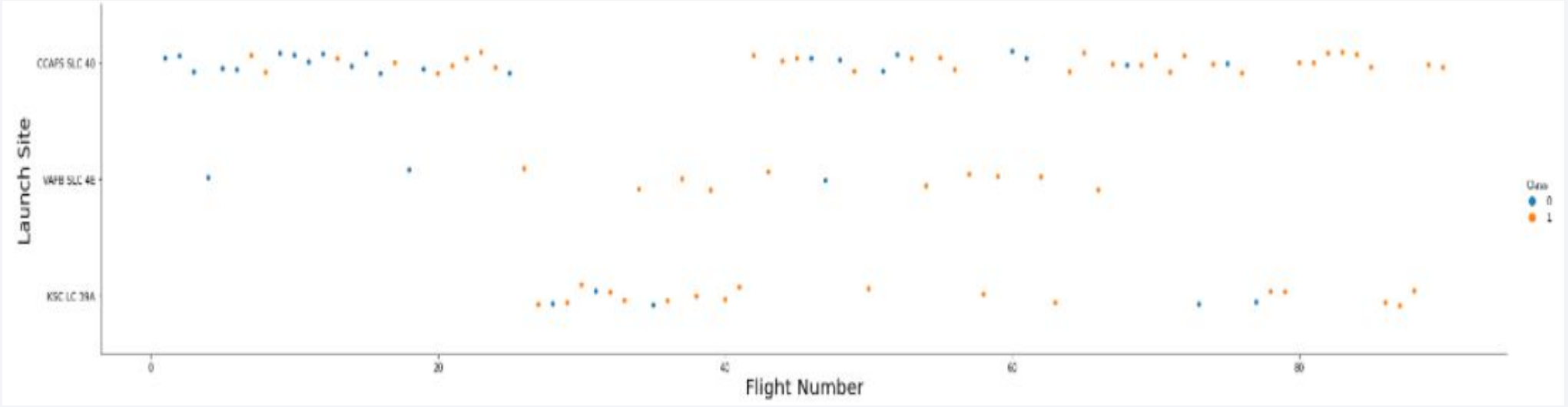
Section 2

Insights drawn from EDA



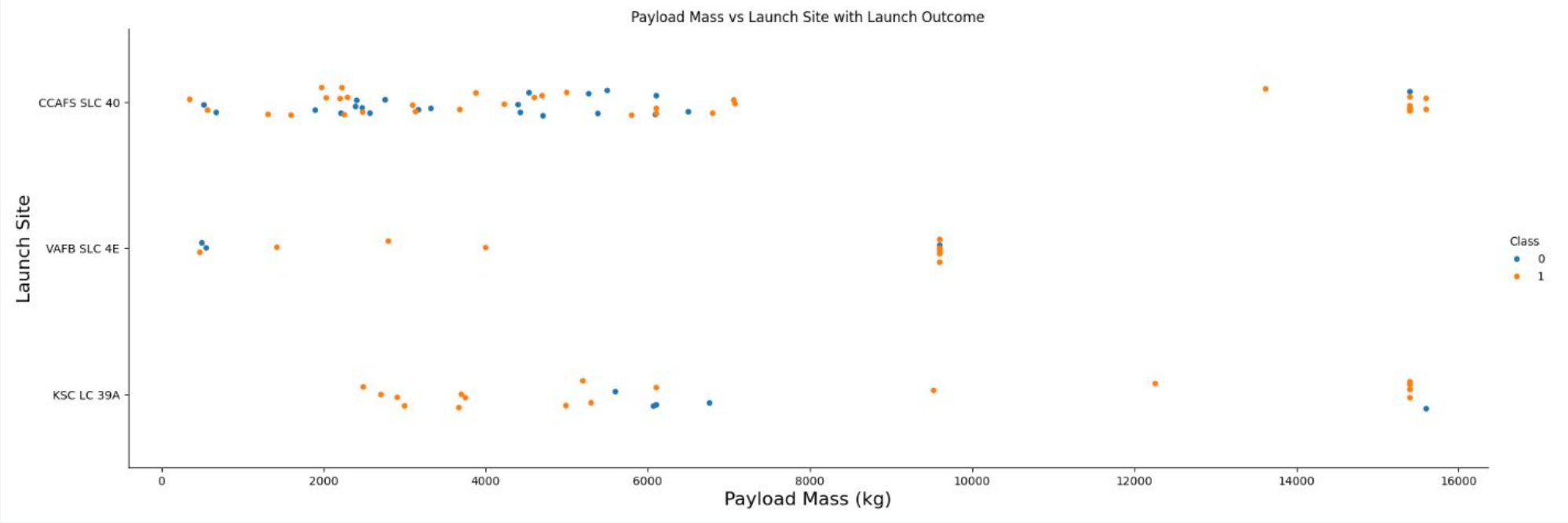
Flight Number vs. Launch Site

- CCAFS-SL 40 is the most used launch site.
- CCAFS-SLC 40 has most of failures in the early stage of Falcon9



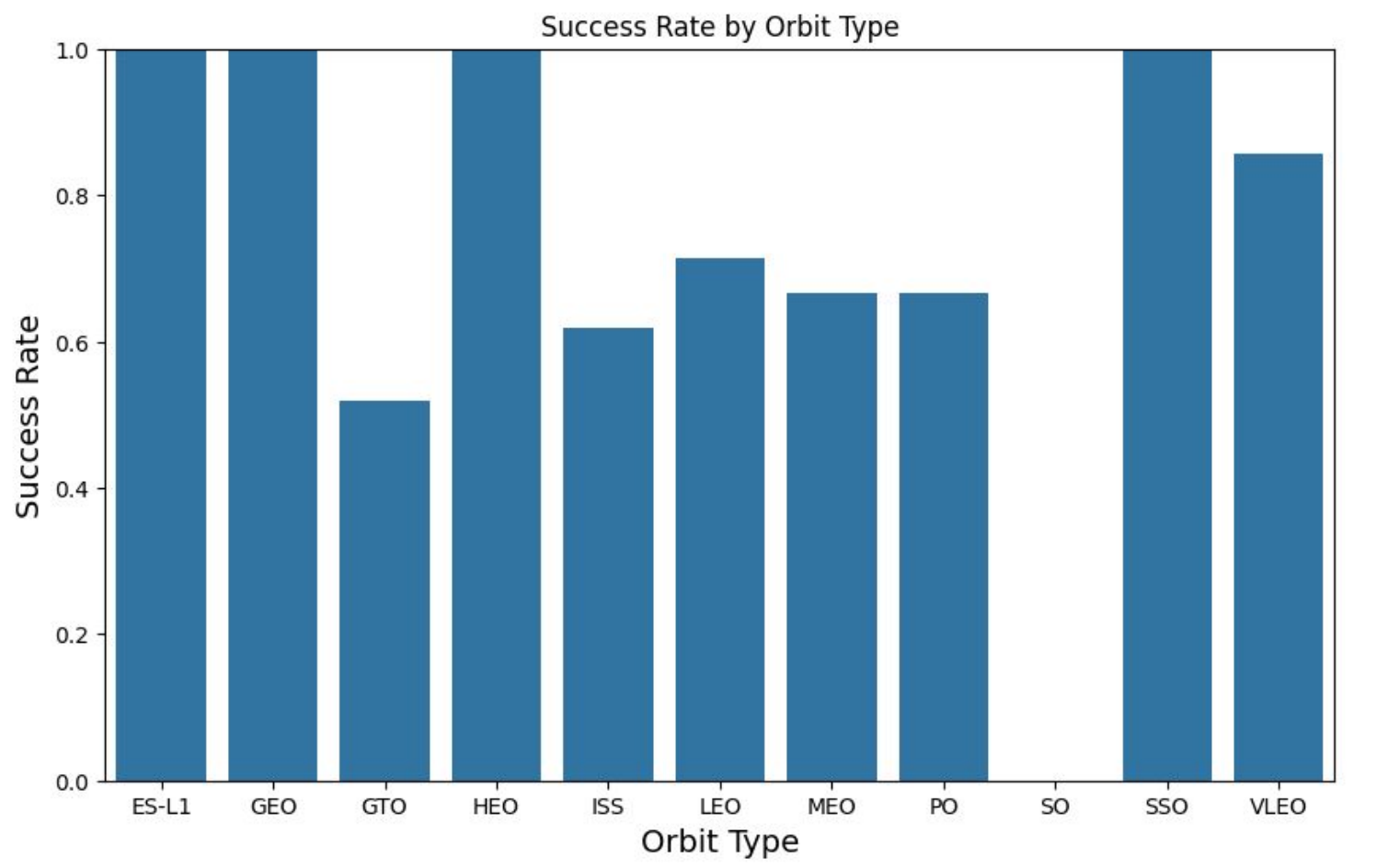
Payload vs. Launch Site

- Falcon9 specifications, heavy payloads sent to low/medium orbits.
- The percentage of failures is lower for heavy payloads.
- Success Rate X Payload Orbit needed some more information.



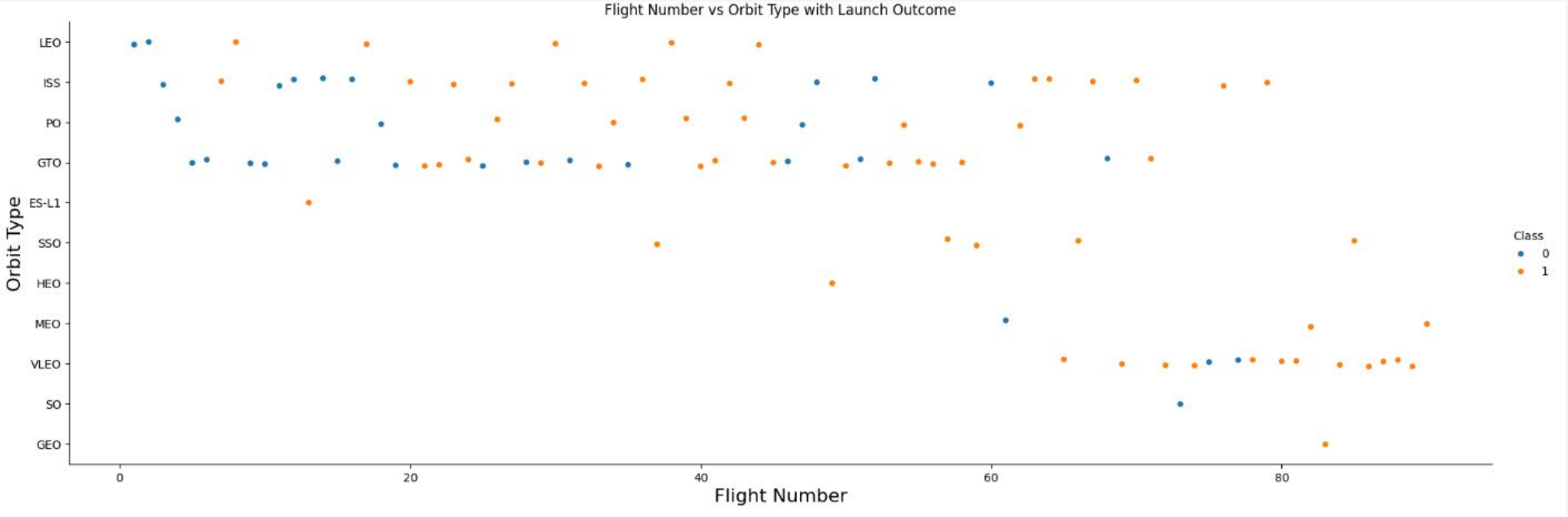
Success Rate vs. Orbit Type

- From the plot, we can see that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- GTO sees the lowesr success rate



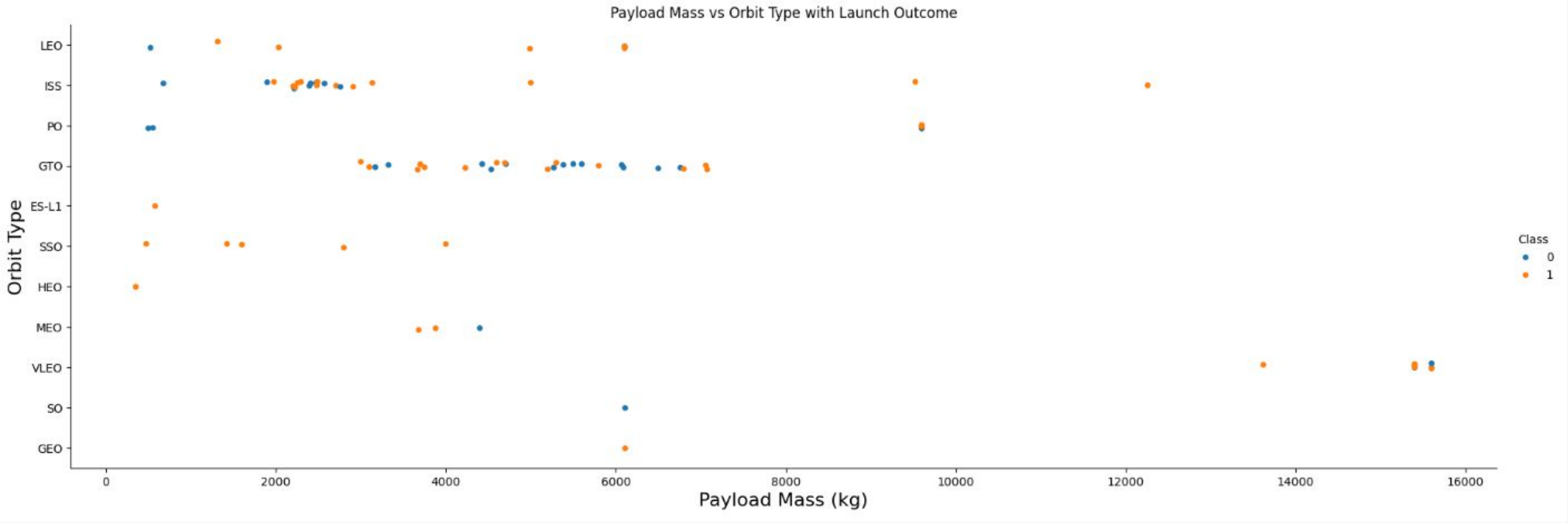
Flight Number vs. Orbit Type

- In the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.



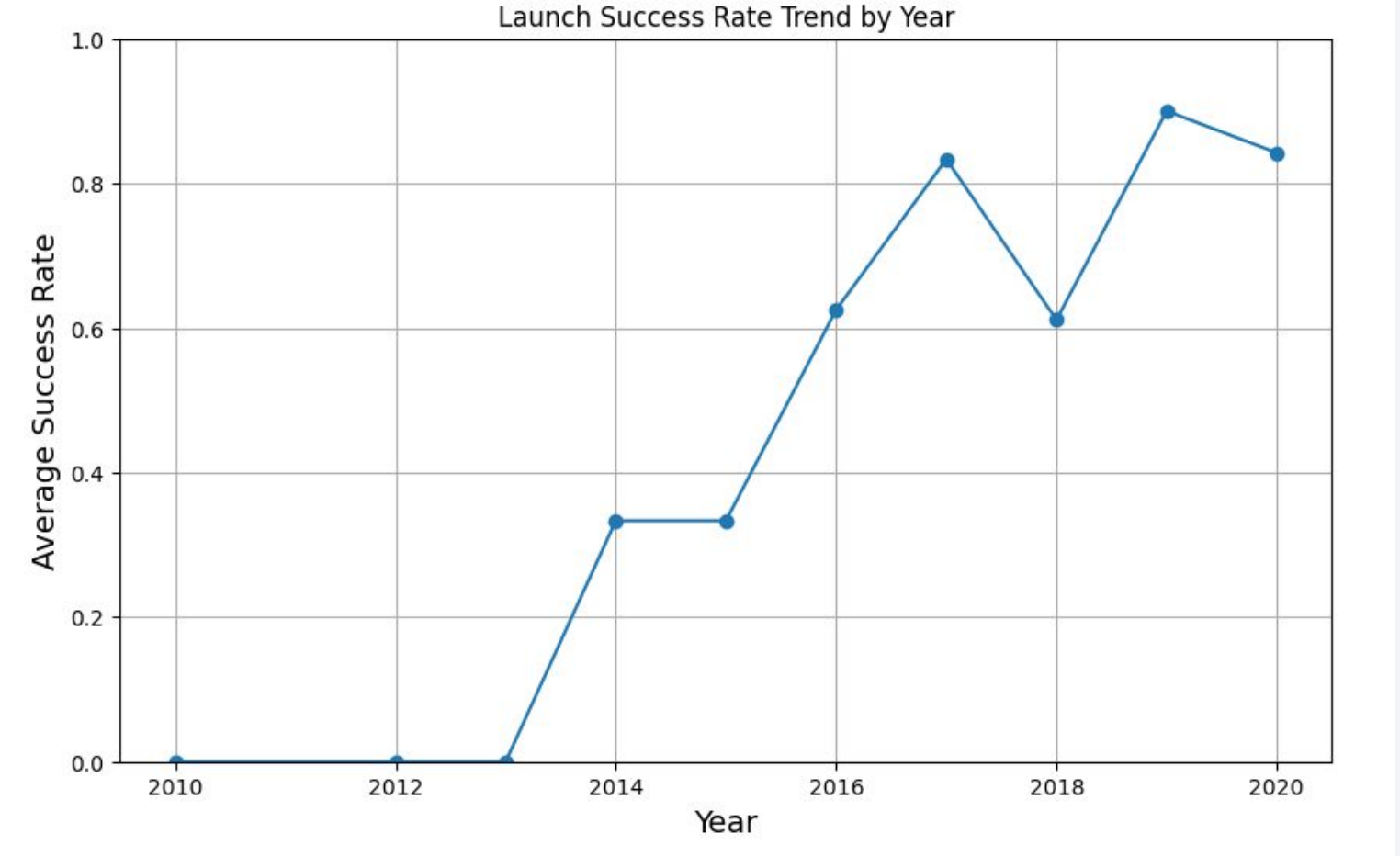
Payload vs. Orbit Type

- Max Success rate with low orbit, except ISS and low payload mass
- Between 2000-7500 ks, success rate seems to be distributed for GTO
-



Launch Success Yearly Trend

- The success rate since 2013 kept on increasing till 2020.
- Falcon9 average booster recovery success rate 66%



All Launch Site Names

- With **DISTINCT** we show only unique launch sites from the SpaceX data.

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'

Task 2

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

```
4]: %sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

4]:

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

- Query to display 5 records where launch sites begin with `CCA`

Total Payload Mass

- Total payload carried by boosters from NASA as 45596 using the query below

Task 3

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

```
%sql SELECT SUM("Payload_Mass__kg_") AS Total_Payload_Mass FROM SPACEXTBL WHERE "Customer" = 'NASA (CRS)';
```

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

Total_Payload_Mass
45596

Average Payload Mass by F9 v1.1

- Calculated the average payload mass carried by booster version F9 v1.1 as 2928.4

Task 4

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

```
[16]: %sql SELECT AVG("Payload_Mass__kg_") AS Average_Payload_Mass FROM SPACEXTBL WHERE "Booster_Version" = 'F9 v1.1';
```

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

[16]:

Average_Payload_Mass
2928.4

First Successful Ground Landing Date

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

Task 5 ¶

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

Hint:Use min function

```
[17]: %sql SELECT MIN("Date") AS First_Successful_Ground_Landing FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

[17]:

First_Successful_Ground_Landing
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

[18]:
```
%sql SELECT "Booster_Version" FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (drone ship)' AND "Payload_Mass__kg_" BETWEEN 4000 AND 6000;
```

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

[18]:

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

Total Number of Successful and Failure Mission Outcomes

- Wildcard like '%' to filter for **WHERE** MissionOutcome was a success or a failure.

List the total number of successful and failure mission outcomes

In [16]:
```
task_7a = '''
        SELECT COUNT(MissionOutcome) AS SuccessOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Success%'
        '''

task_7b = '''
        SELECT COUNT(MissionOutcome) AS FailureOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Failure%'
        '''
print('The total number of successful mission outcome is:')
display(create_pandas_df(task_7a, database=conn))
print()
print('The total number of failed mission outcome is:')
create_pandas_df(task_7b, database=conn)
```

Out[16]:
```
The total number of successful mission outcome is:
```

	successoutcome
0	100

```
The total number of failed mission outcome is:
```

	failureoutcome
0	1

Boosters Carried Maximum Payload

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

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

```
[21]: %sql SELECT "Booster_Version", "Payload_Mass__kg_" FROM SPACEXTBL WHERE "Payload_Mass__kg_" = (SELECT MAX("Payload_Mass__kg_") FROM SPACEXTBL);
```

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

[21]:

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

- Used **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

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.

[22]:
```
%sql SELECT substr("Date", 6, 2) AS Month, "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL WHERE "Landing_Outcome" LIKE '%drone ship%' AND "Landing_Outcome" LIKE '%Failure%' AN
```

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

[22]:

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

- Applied the **GROUP BY** clause to group the landing outcomes and the **ORDER BY** clause to order the grouped landing outcome in descending order.

Task 10

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

```
[23]: %sql SELECT "Landing_Outcome", COUNT(*) AS Outcome_Count FROM SPACEXTBL WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY Outcome_Count DESC;
 * sqlite:///my_data1.db
Done.
```

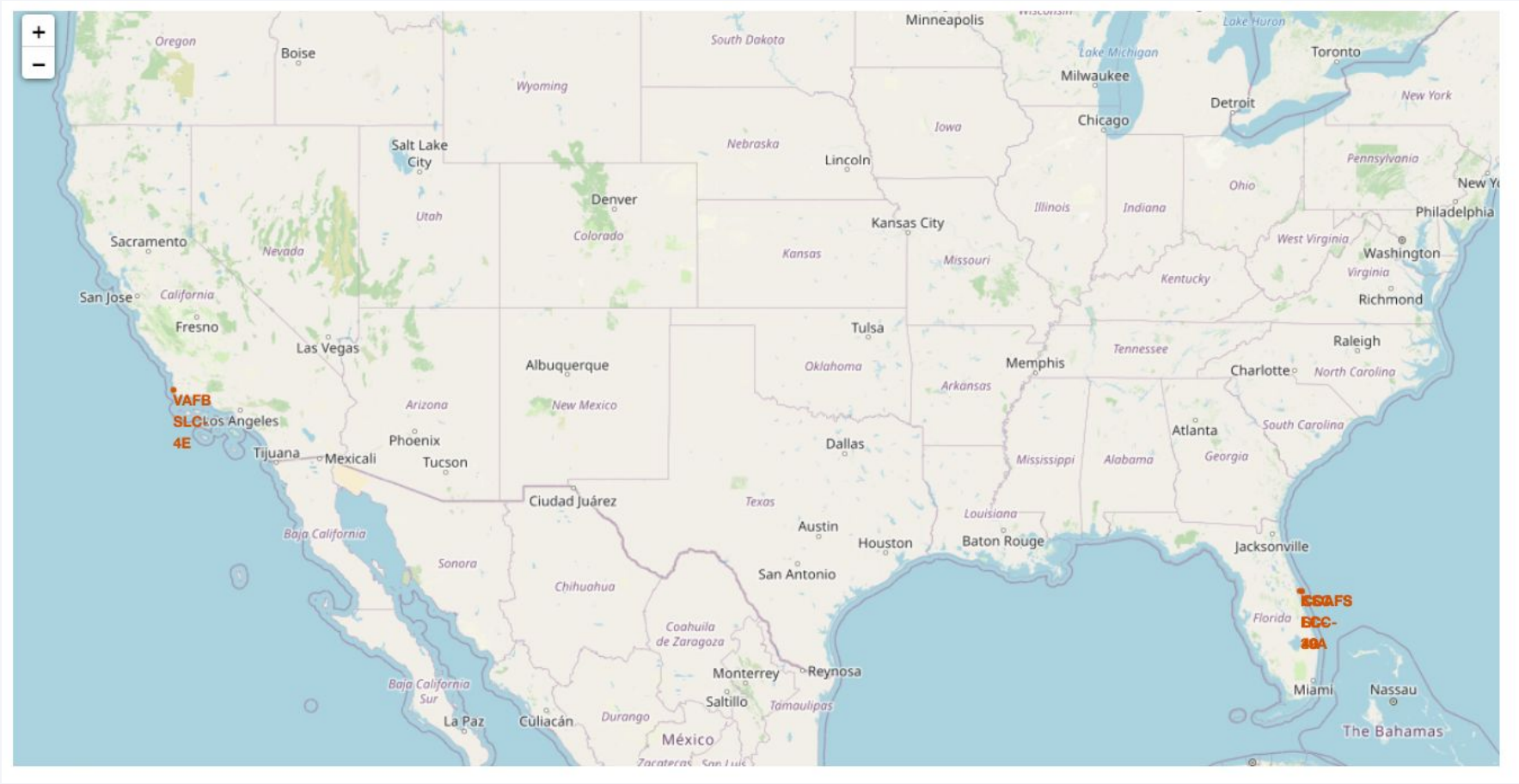
[23]:

Landing_Outcome	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

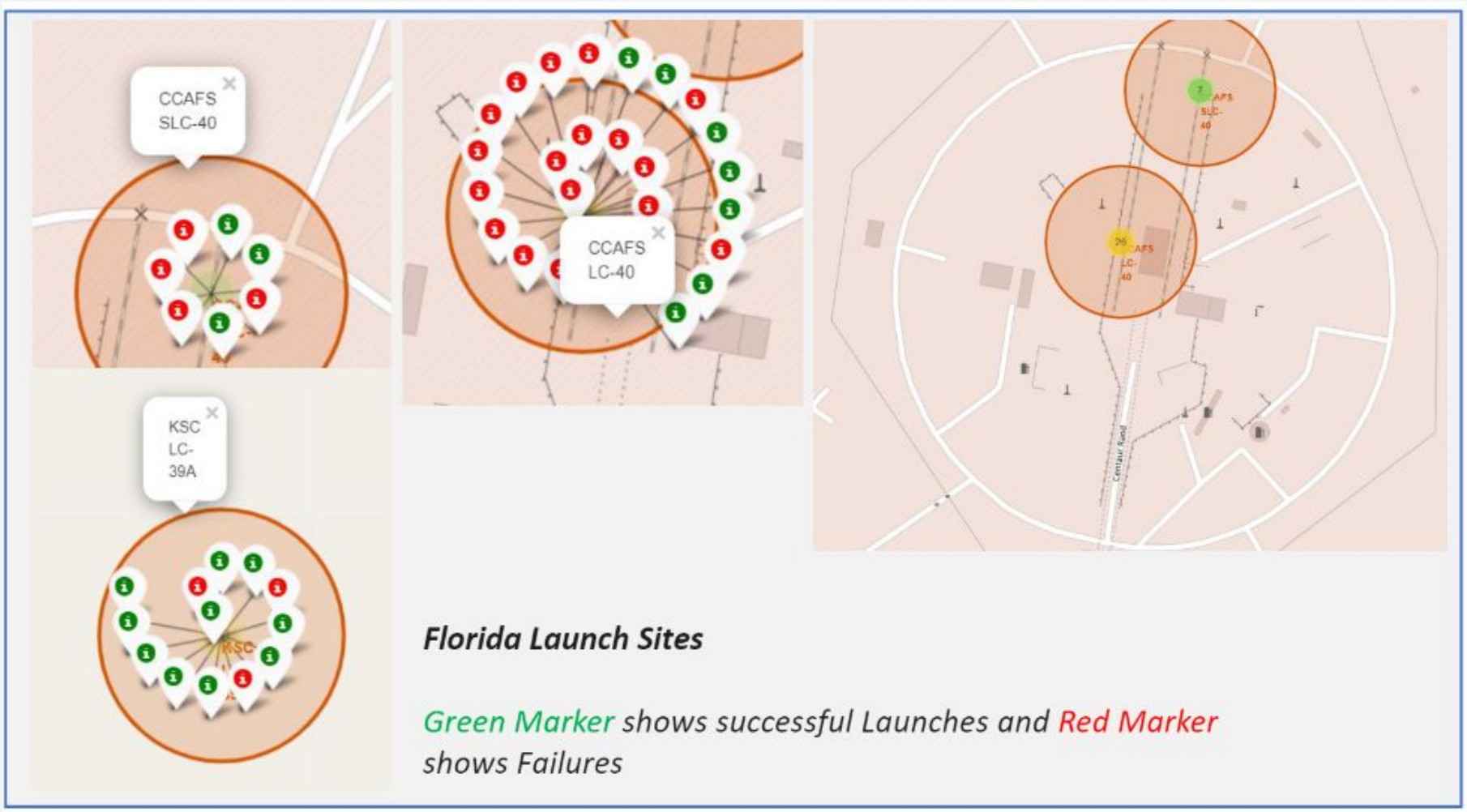

Section 3
Launch Sites
Proximities Analysis

Launch sites global map

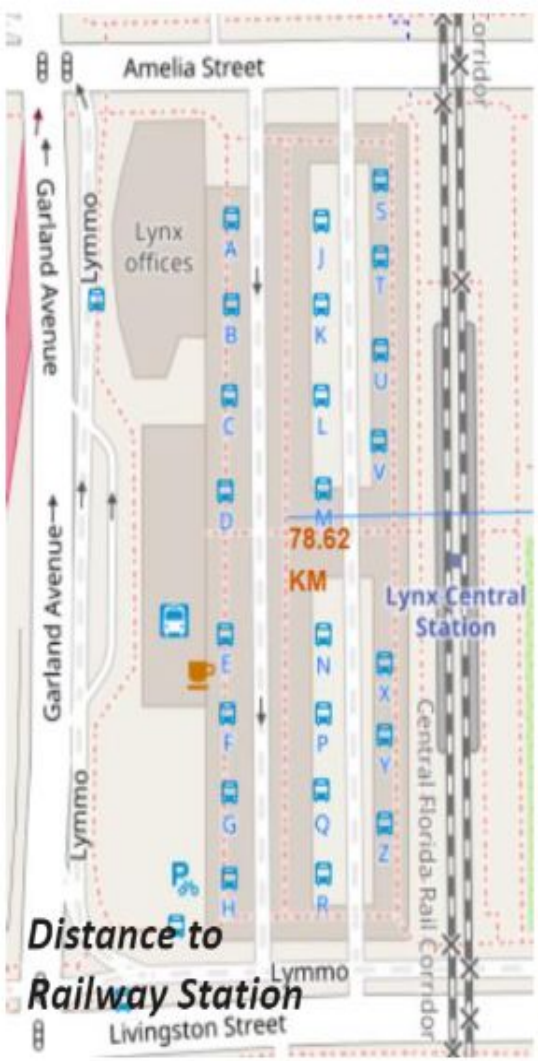
- We can see the Space X launch sites in the USA coasts.



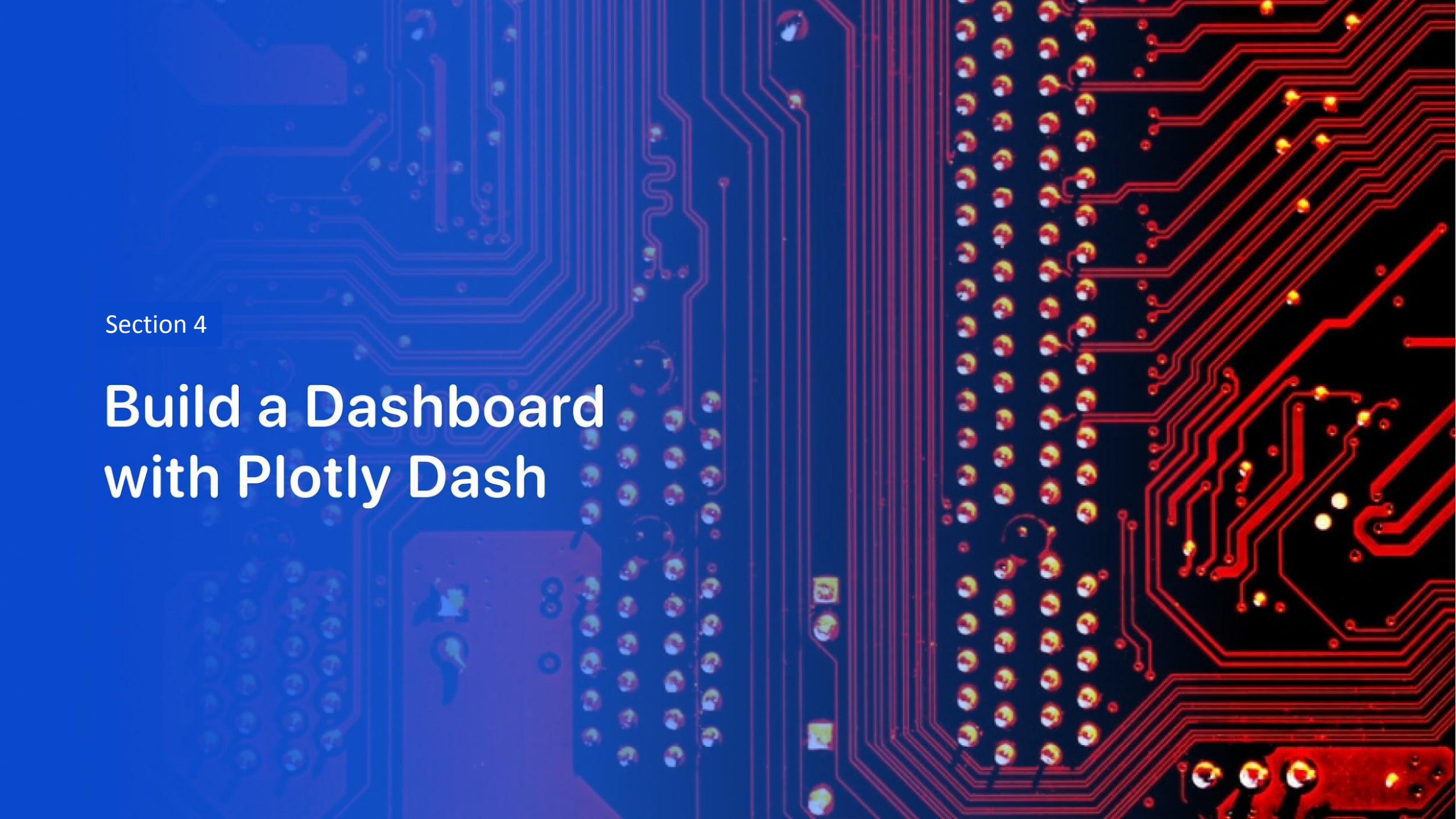
Markers showing launch sites with color labels



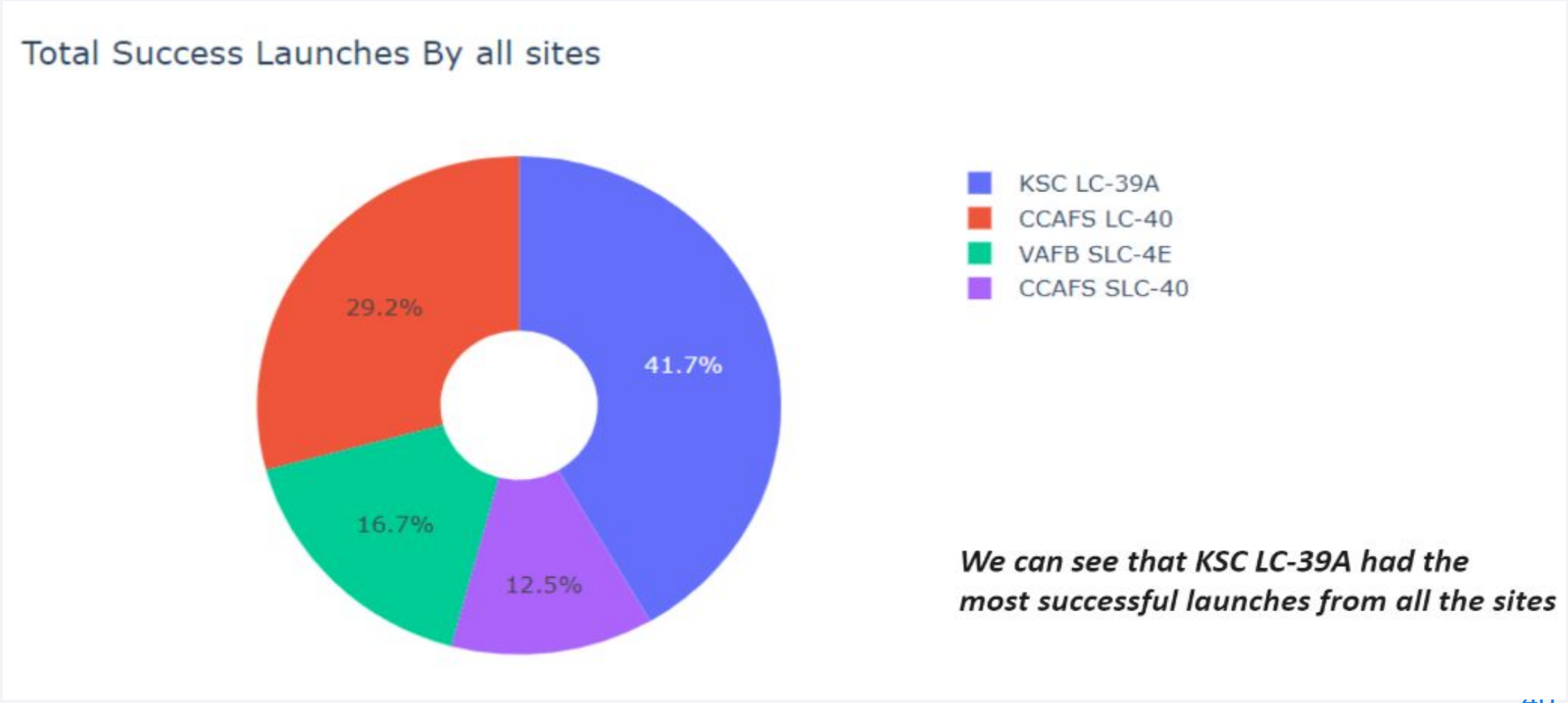
Launch Site distance to landmarks



- 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

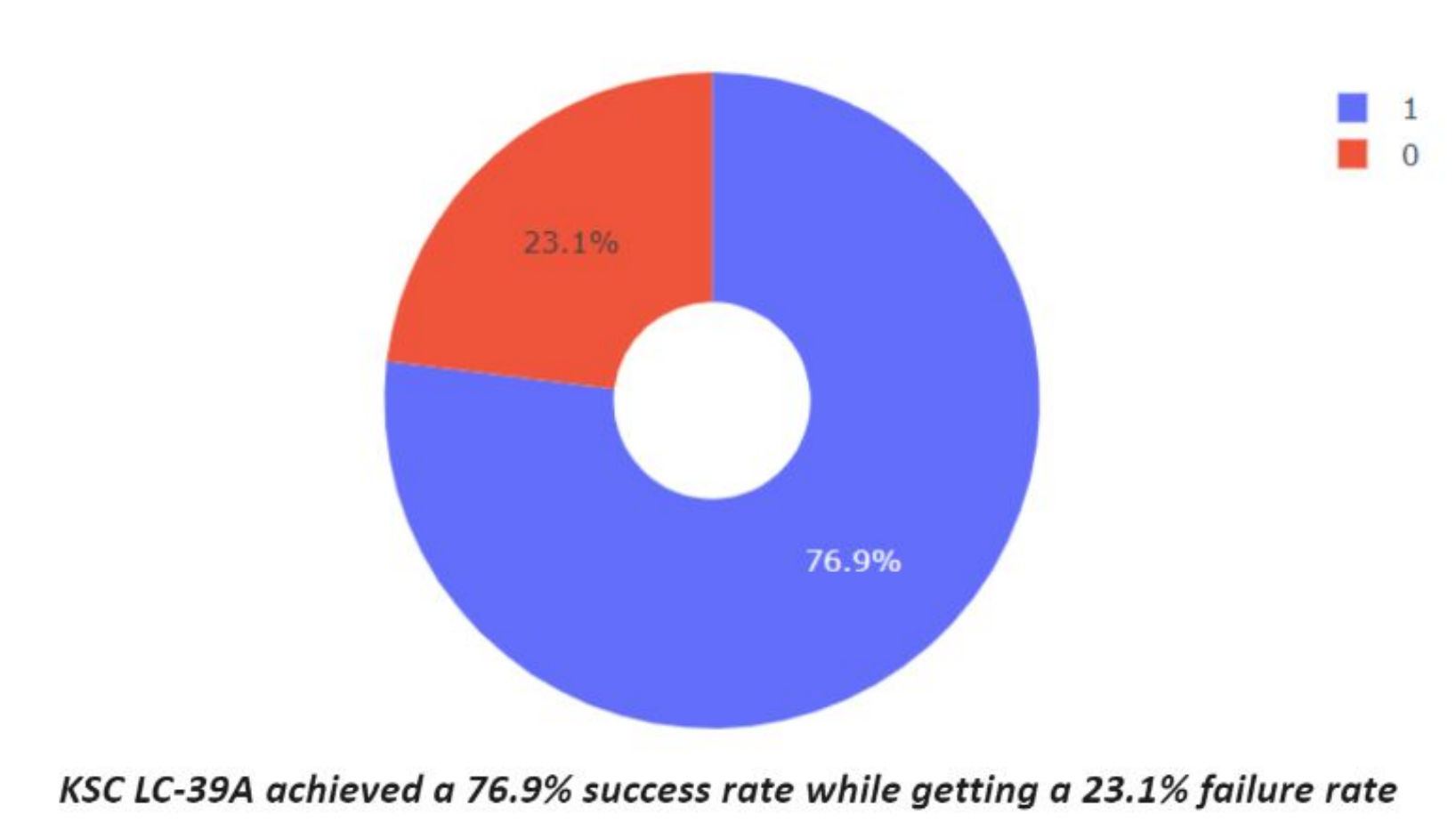

Section 4
Build a Dashboard with Plotly Dash

Pie chart showing the success percentage achieved by each launch site



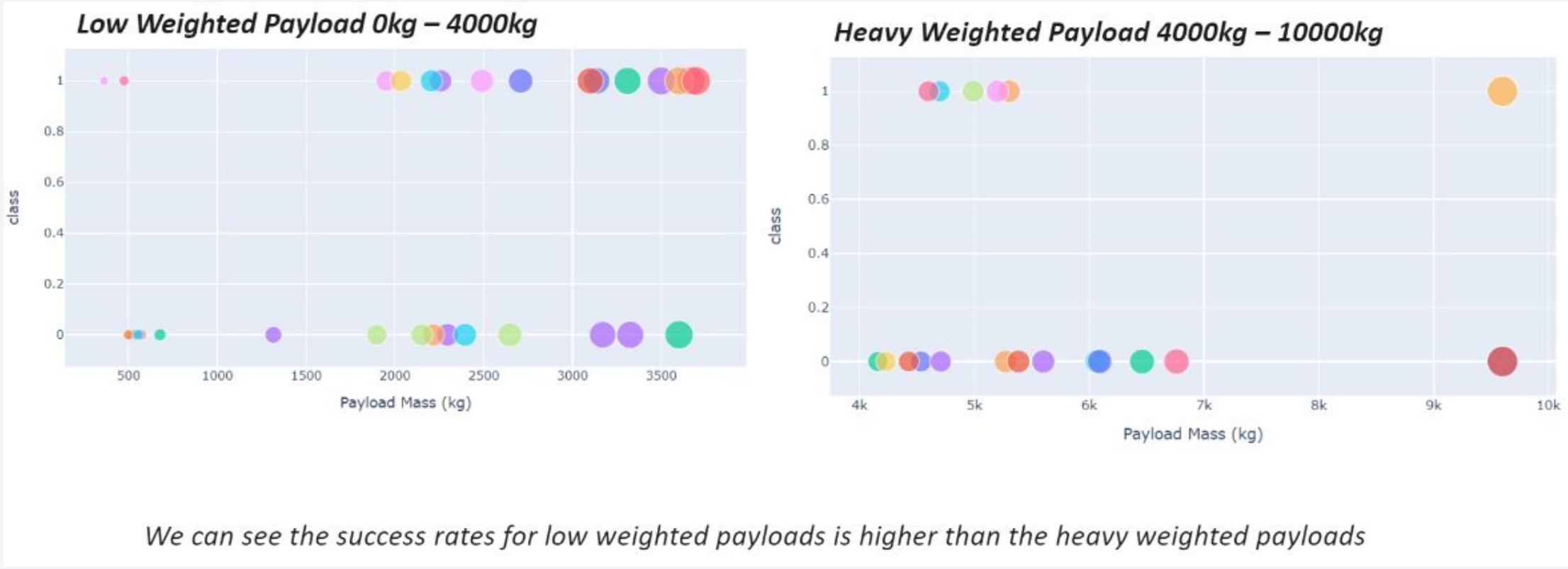
We can see that KSC LC-39A had the most successful launches from all the sites

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



KSC 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

- The Logistic Regression classifier is the model with the highest classification accuracy

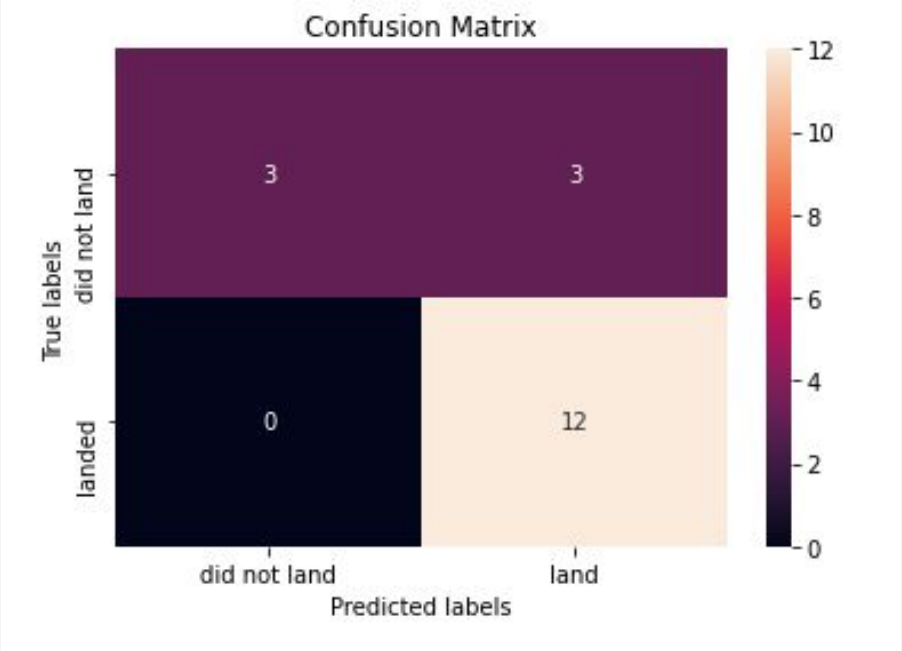
Find the method performs best:

41]:
```
accuracy_lr = logreg_cv.score(X_test, Y_test)
accuracy_svm = svm_cv.score(X_test, Y_test)
accuracy_tree = tree_cv.score(X_test, Y_test)
accuracy_knn = knn_cv.score(X_test, Y_test)

print(f"Logistic Regression Accuracy: {accuracy_lr:.4f}")
print(f"SVM Accuracy: {accuracy_svm:.4f}")
print(f"Decision Tree Accuracy: {accuracy_tree:.4f}")
print(f"KNN Accuracy: {accuracy_knn:.4f}")

best_accuracy = max(accuracy_lr, accuracy_svm, accuracy_tree, accuracy_knn)

if best_accuracy == accuracy_lr:
    best_model = "Logistic Regression"
elif best_accuracy == accuracy_svm:
    best_model = "SVM"
elif best_accuracy == accuracy_tree:
    best_model = "Decision Tree"
else:
    best_model = "KNN"

print(f"\nBest performing model is: {best_model} with accuracy {best_accuracy:.4f}")
```

```
Logistic Regression Accuracy: 0.8333
SVM Accuracy: 0.8333
Decision Tree Accuracy: 0.7778
KNN Accuracy: 0.8333

Best performing model is: Logistic Regression with accuracy 0.8333
```

Confusion Matrix

- The confusion matrix for the decision tree classifier shows distinguish between different classes.
- A problem is the false positives, unsuccessful landing marked as successful landing by the classifier.



Conclusions

- The larger the flight amount at a launch site, greater is the success rate
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites.
  - Success rate started to increase in 2013 till 2020.
- The Decision tree classifier is the best tool for this task.

Thank you!