IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Rickey D Jackson Jr
08-02-2024

Outline

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

Executive Summary

- In this report we will look at what makes SpaceX the leader in private space launches and how to replicate that success and blaze our own trail in the sky
- By collecting data from spacexdata.com and Wikipedia we are able to gain insight into the different booster versions of SpaceX launches along with their payload and success rate.
- After processing and analyzing the data: using a launch site near Orlando, FL, focusing on Low-Earth Orbits and deliveries to the International Space Station, and carrying nearly 3000 kg of cargo give us the best chance at early success to build a foundation from.

Introduction

- Project background and context – NASA and many commercial customers are looking for affordable means to carry cargo and passengers into space. Space Y founder Allon Mask is looking for a competitive edge in this young and growing industry.
    - Using Data Science whereby we gather data, process it, analyze it, and then extract solutions using machine learning we searched for that edge.
- After all the designing, engineering, and testing, the always persistent problem with space flight is cost.
- Taking passengers and cargo into space is very expensive.
    - We have to find the most efficient and cost-conscious way to fulfill our customers' requests.
    - The largest expense is the first stage of the rocket. Analyzing data from SpaceX previous launches we searched for the key factors used to launch rockets cheaper than their competitors and expand on those concepts.

Section 1

Methodology

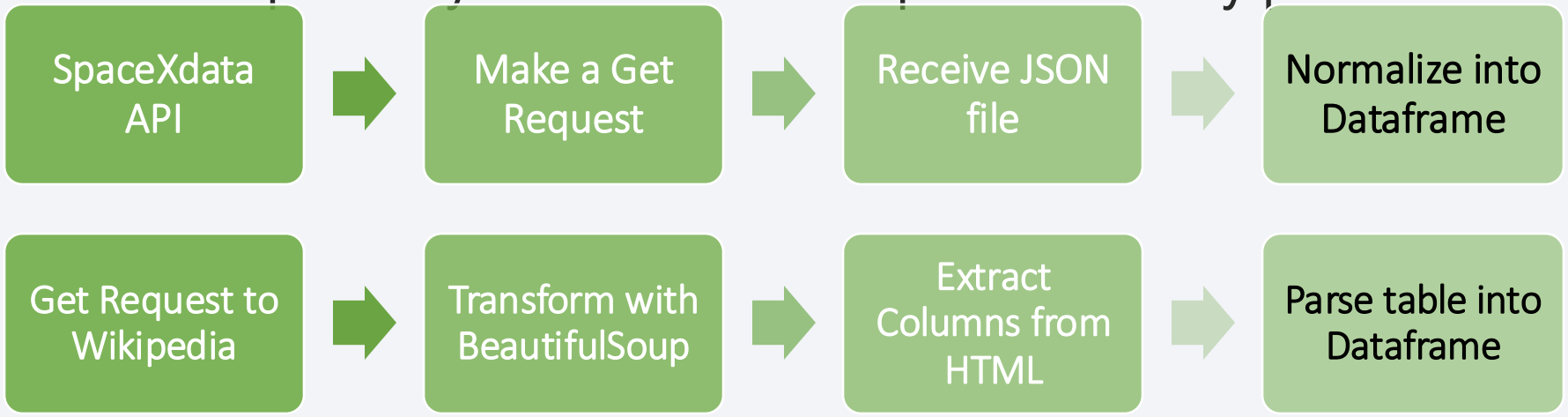
Methodology

Executive Summary

- Data collection methodology:
  - Data was collected from SpaceX API and web scrapped from Wikipedia
- Perform data wrangling
  - Extracted data was processed into accessible tables, unrelated data was removed, and focus was placed on success and failure to recover first stage boosters.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Separating data into train and test sets models were trained using logistic regression, support vector machine, tree, and KNN to determine the likelihood of a first stage successfully landing to be reused and save the company the cost build another.

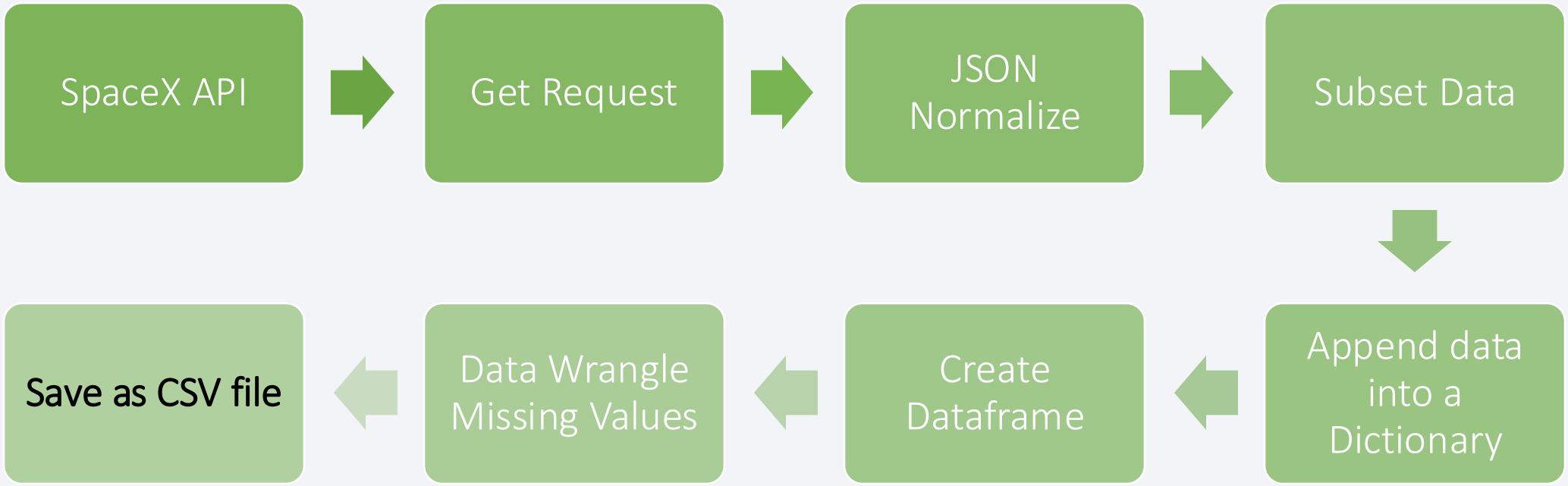
Data Collection

- Describe how data sets were collected. - To collect the data we used python's "Request" library to get a HTTP request from Spacexdata.com API. After a response we parsed the data and decoded it into a JSON file before turning it into a Dataframe to begin our research.
- We also web scrapped Wikipedia by again using a request to the website and turning its response into a BeautifulSoup object. From there we extracted columns from the HTML headers then parsed the tables into a Dataframe
- You need to present your data collection process use key phrases and flowcharts



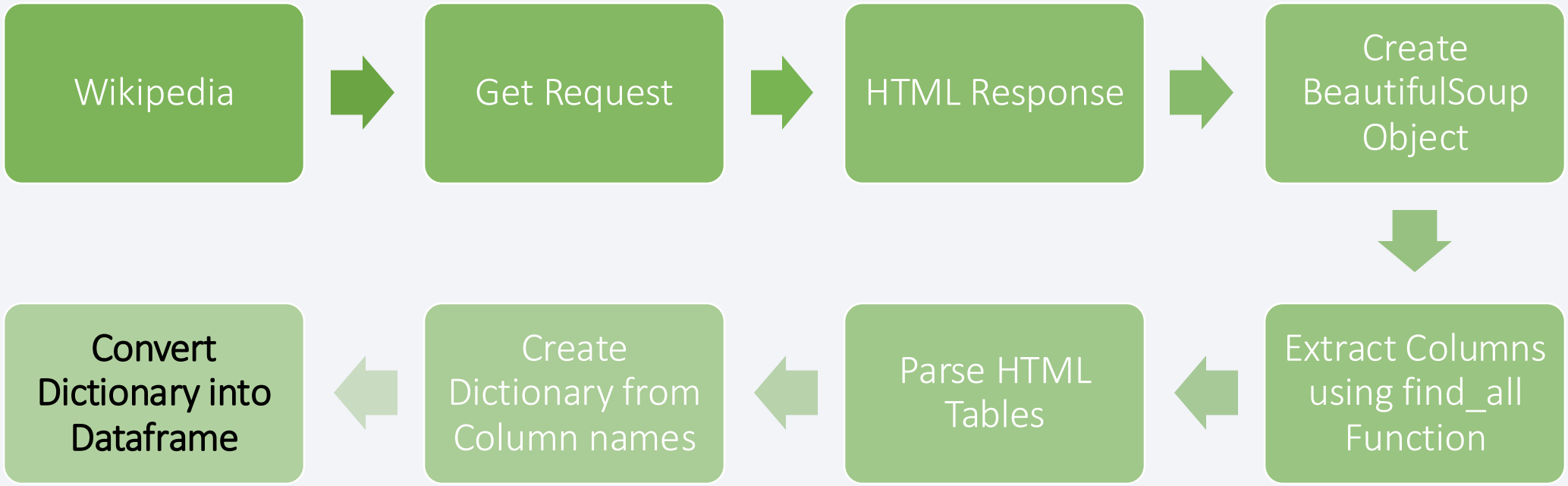
Data Collection – SpaceX API

SpaceX Data Collecing API 01a



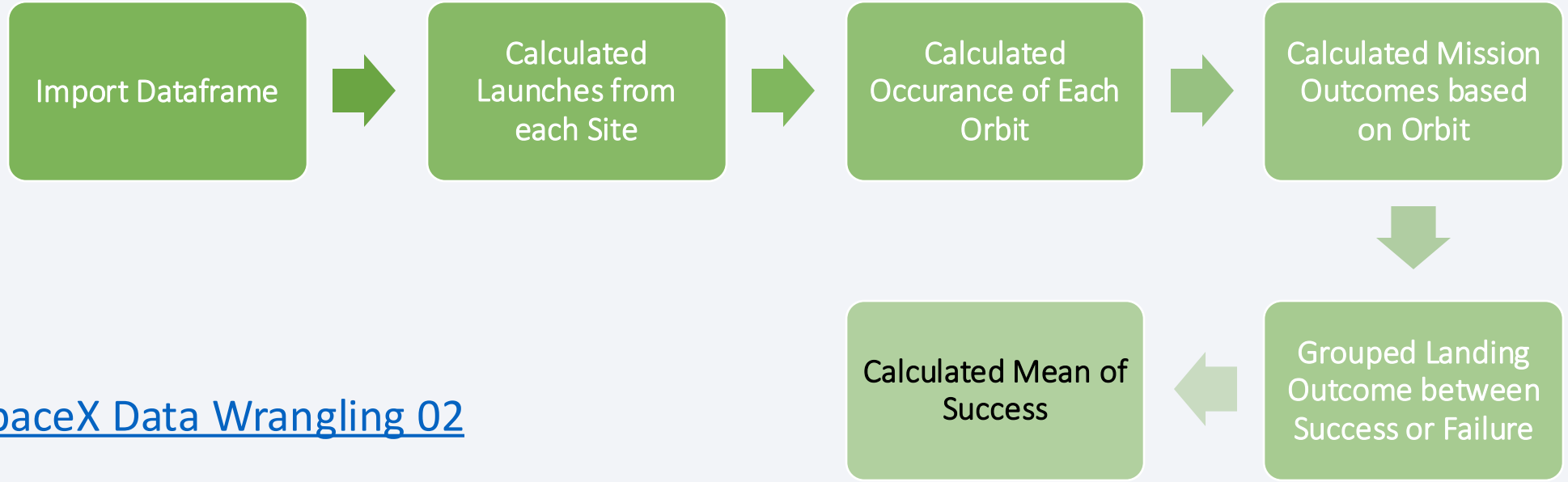
Data Collection - Scraping

SpaceX - Data Collecting - 01b Web Scrapping



Data Wrangling

- Describe how data were processed – We imported the data collected from the API and Wikipedia and loaded it into a dataset from there:
  - We identified missing values and filled them in with appropriate approximations to retain important information but also allow the data to be analyzed.
  - Identified Launch Sites and number of launches from each location
  - Identified and tallied orbits for each launch
  - And finally identified different landing sites and numbers of success and failures for the boosters to land safely.



SpaceX Data Wrangling 02

EDA with Data Visualization

- Scatterplots were used to observe payload, orbit, and launch site in relation to flight number
  - Reason: It is important to note if useful data is collected to improve each launch to see if commercial rockets are even feasible. As well as observe if there is a correlation between payload weight, orbit, and launch site and the success of a returned booster.
- A Bar graph was then used to measure success based strictly on orbit
  - This will help determine which orbit has the highest chance of successful recovery, focusing on accepting those missions will thereby minimizing loses
- A Line graph was finally used to map the progress of successful lands
  - It is important to note if SpaceX is improving their chances at recovery and know what percentage we must match to remain competitive.

SpaceX EDA with Data Visualization 03

EDA with SQL

- Using the distinct query we identified the launch sites
- Using the sum query we calculated the total payload sent up on behalf of NASA (as NASA is the currently the primary customer for SpaceX)
- Observing this we used the average payload to approximate how heavy each launch was on average (this helps give us a ballpark of how much weight we should ensure our rockets can carry successfully)
- Knowing the average weight we honed in on drone ship landings and noted how successful SpaceX was in recovering the booster. As well as from their first launch until they began recovering the boosters on the drone ship.
- We grouped the mission outcome to identify the most and least successful ways to recover the booster.
- Finally we identified which booster was used for these weight constraints and used the drone ship for recovery.

SpaceX EDA with SQL – 04

Build an Interactive Map with Folium

After examining the flights and what influences payload, orbit, and other elements might play a factor. We took a look at the launch locations for information.

- Using a Folium Interactive Map we created various objects.
  - First we marked each of the four launch locations: CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E
    - We added these locations to see if there was a commonality between them that may aid our goal of safe and efficient space flight.
  - We next added each launch and if it was successfully recovered or not using a cluster function to keep the map clean but informative while grouping them into circles for ease of reference
    - By adding each flight we were able to see which site was used the most and from there focus on why.
  - From there we added lines showing distances between cities, transportation means, and the coast
    - The city and the coast were measured for safety purposes while transportation was marked to estimate cost of moving personnel and materials

SpaceX Folium Interactive Map - 05

Build a Dashboard with Plotly Dash

- To easily visualize and access information, we built a dashboard.
  - First displays flights by Launch Sites along with a total
    - This reinforces what we see in the Folium Map by showing percentages and comparisions between the varius sights.
  - Our next dropdown shows the relationship between payload and outcome.
    - This helps to visualize potential issues with certain ranges of payload and our ability to recover the boosters.

SpaceX - Ploty Dashboard - 06

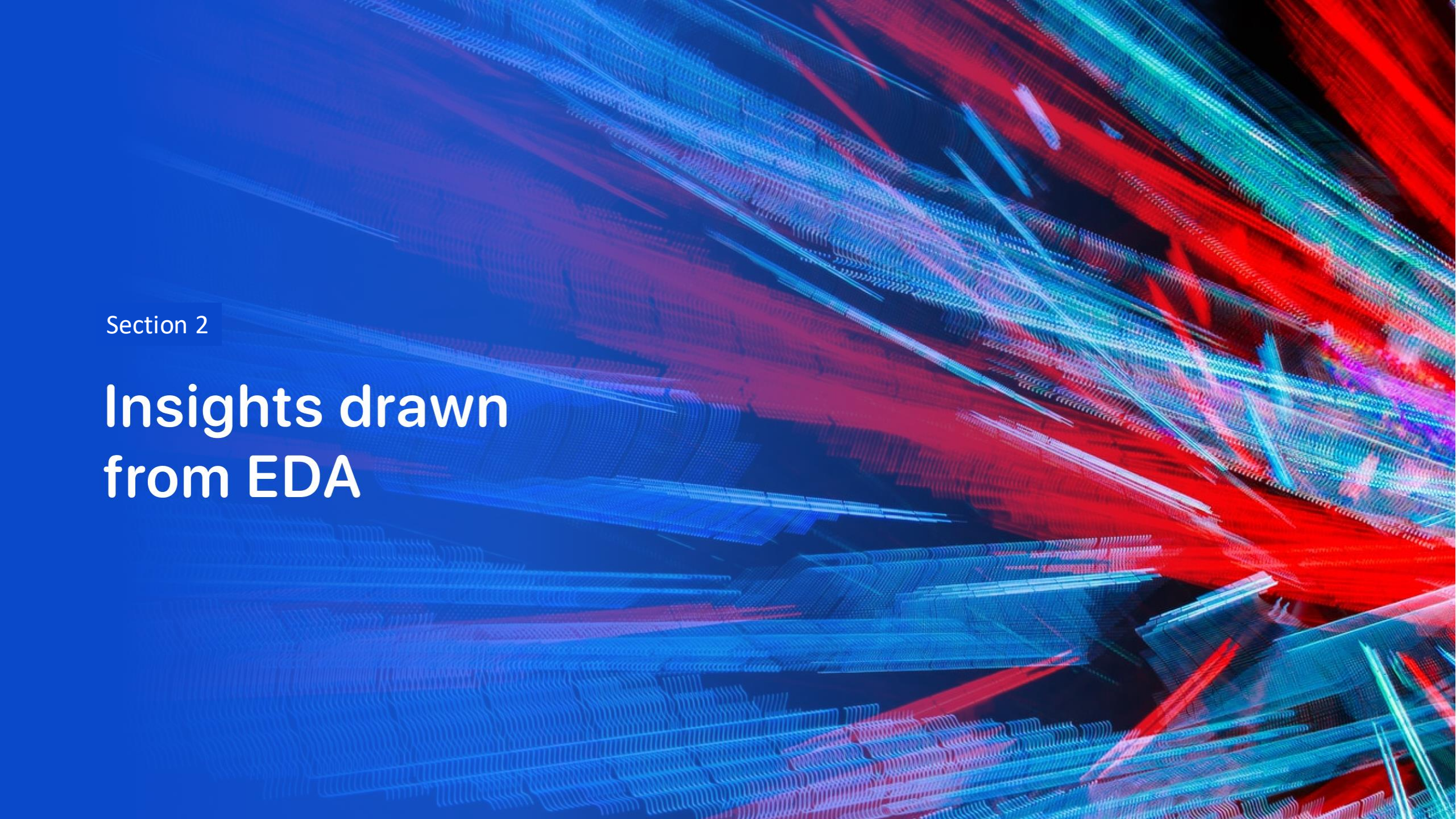
Predictive Analysis (Classification)

- After gathering various pieces of data we then set about to create a predictive model considering various factors to see if we could with a degree of certaintly predict if a booster would be recovered or not.
- To do this we collected the data, preprocess it, separated the data into train and test sets and applied varying machine learning methods.
- Standardizing parameters to minimize noise we ran the train data through four different models: Logistic regression, Support Vector Machine, Tree Classifier, and K-Nearest Number
- Summarize how you built, evaluated, improved, and found the best performing classification model
- After evaluating the accuracy of each model against the test data, we selected Tree Classification as the best model to predict SpaceX booster recovery.

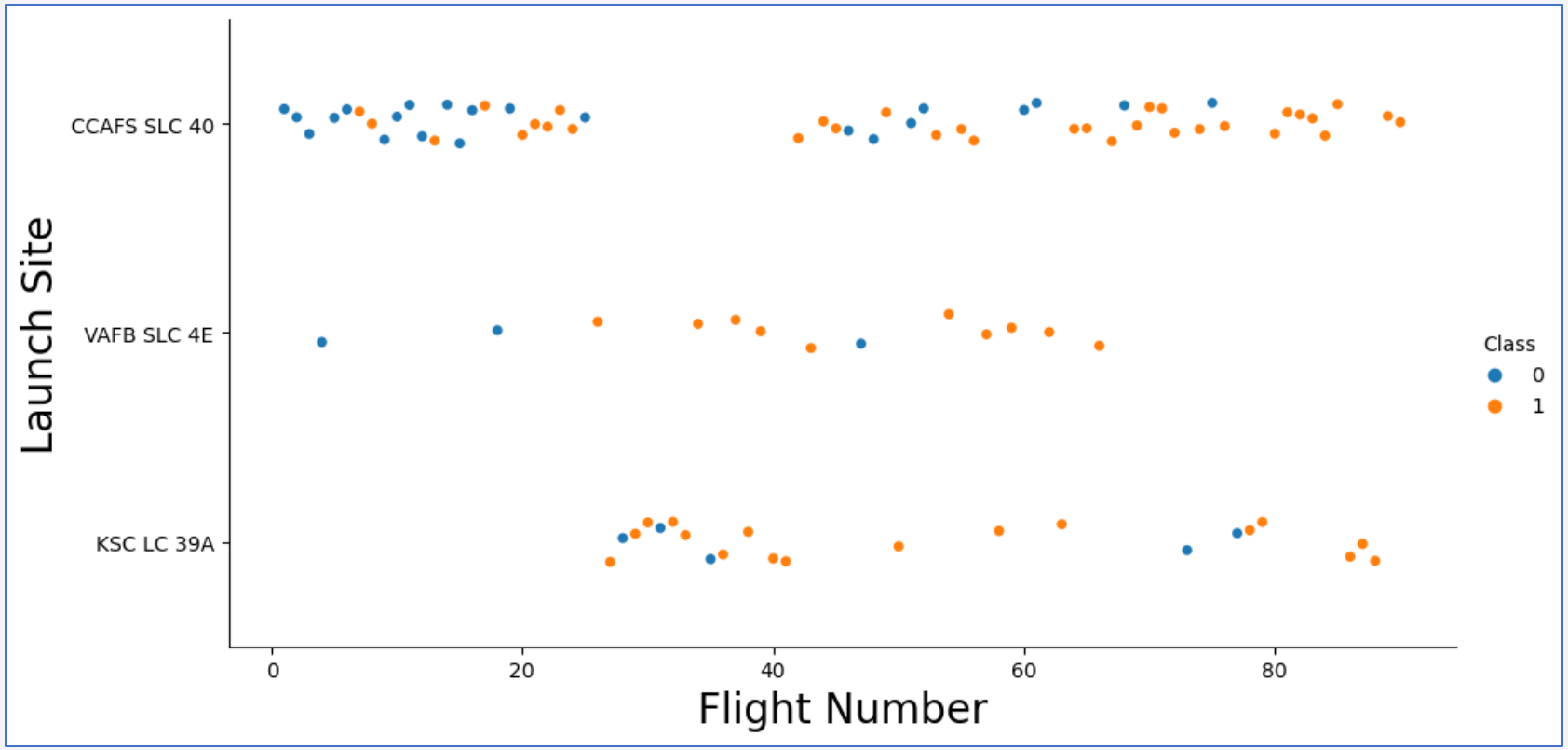
SpaceX Machine Learning Model - 07

Results

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

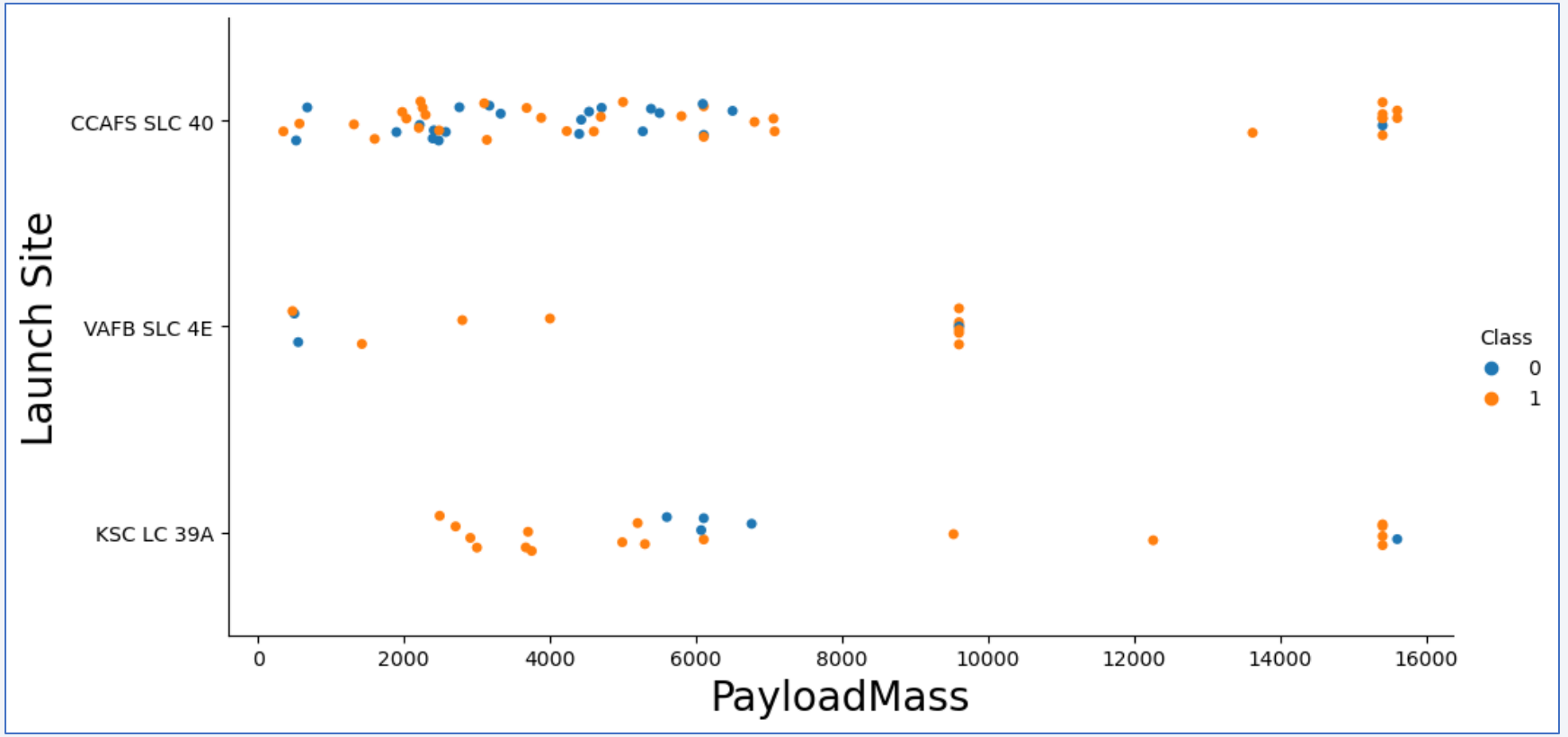

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- A scatter plot shows SpaceX primarily launches from CCAFS SLC-40 and learns from each launch thus increasing recovery rate

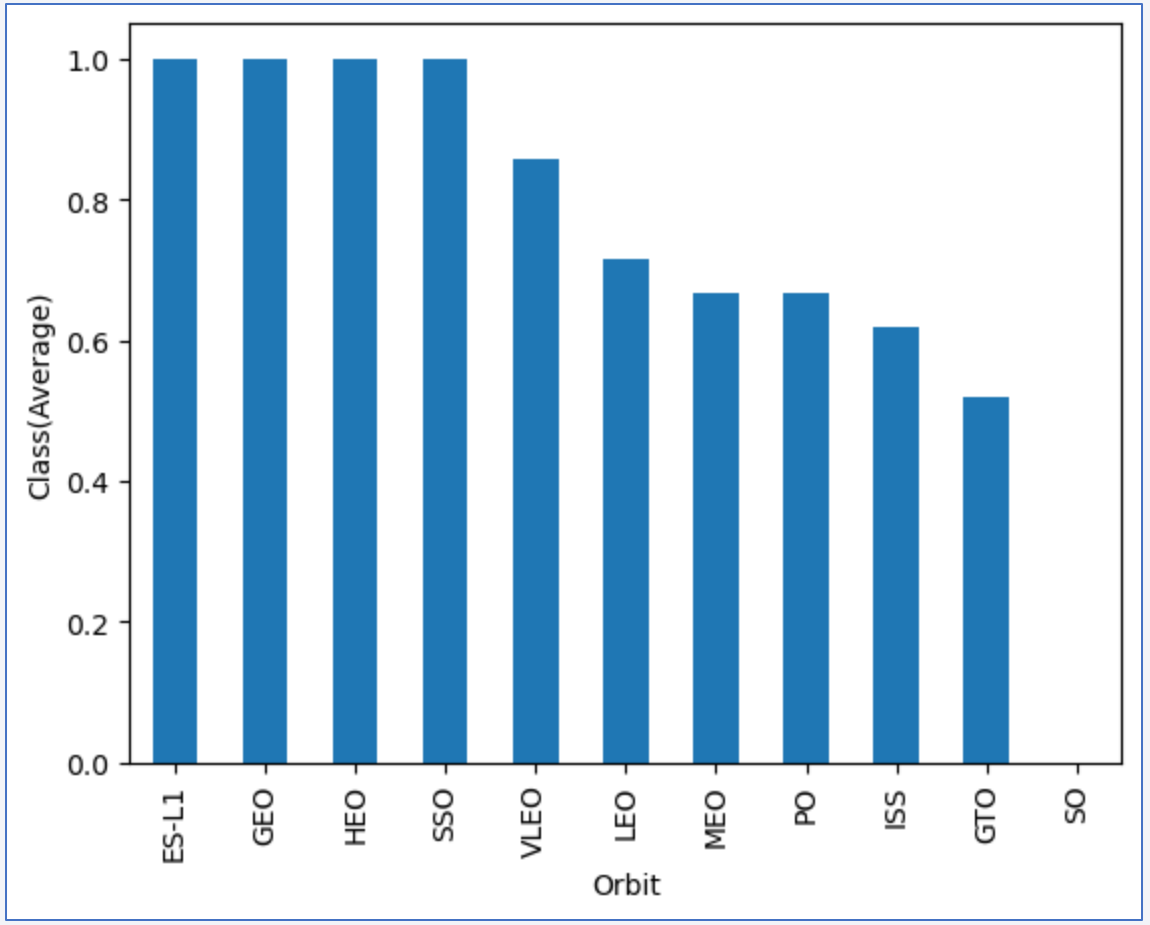
Payload vs. Launch Site



- All three sites struggle to recover the booster with the payload is on the low end.
- CCAFS SLC-40 & KSC LC-39A both are very successful at the 15,000 kg range

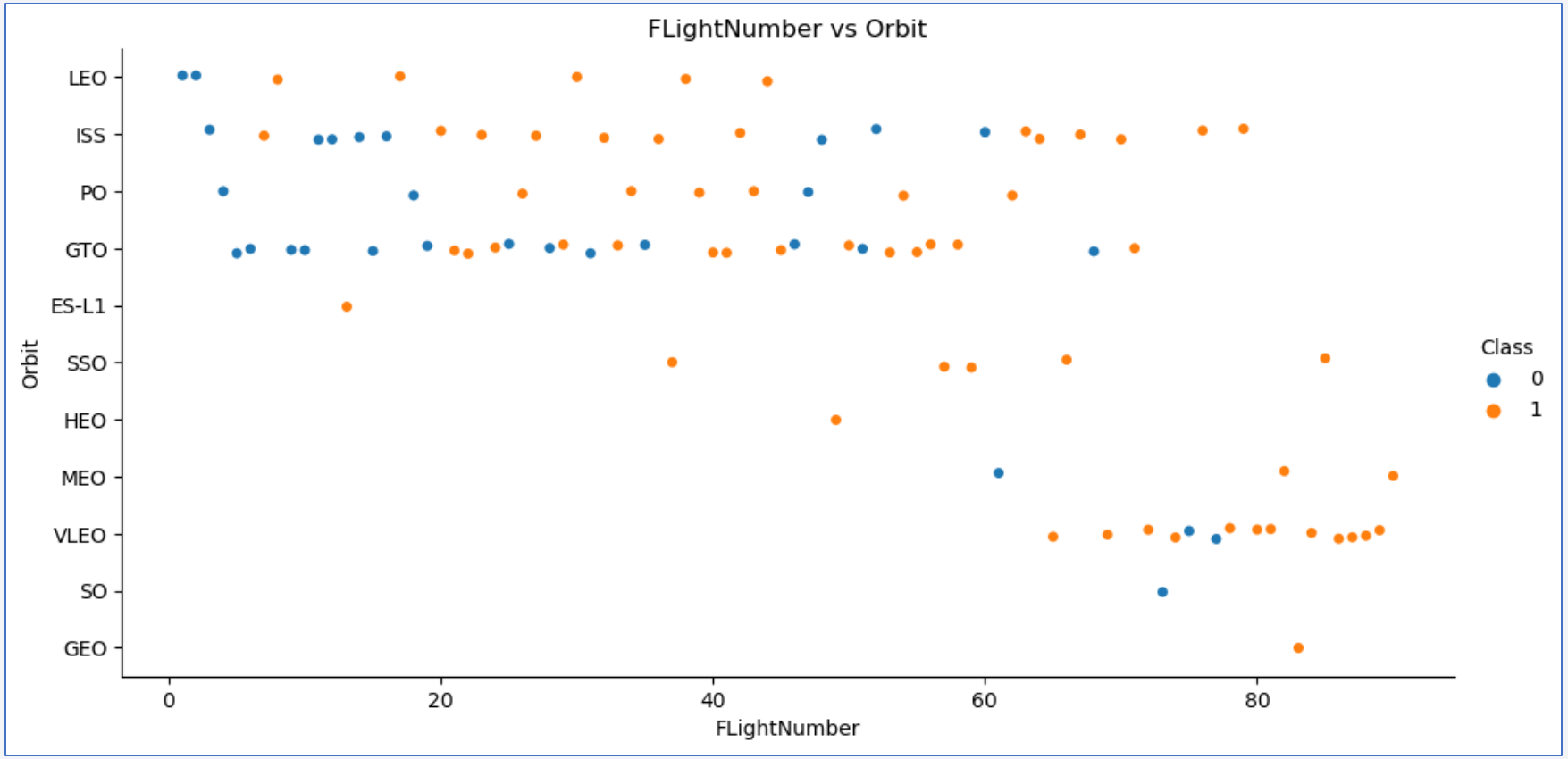
Success Rate vs. Orbit Type

- ISS: International Space Station
- MEO: Medium Geocentric orbits 2,000 km to 35,786 km
- HEO: High Geocentric - Above 35,786 km
- GEO: Geosynchronous Orbit at 35,786 km
- PO: Pole Orbit
- Geostationary Orbit
- SSO: Sun-synchronous orbit
- LEO: Low Earth orbit
- VLEO: Very Low Earth Orbits
- GTO: A Geostationary Orbit
- Point HEO: Highly Elliptical Orbit
- ES-L1: A Lagrange



- The chart shows ES-L1, GEO, HEO, and SSO have the highest average success rate

Flight Number vs. Orbit Type

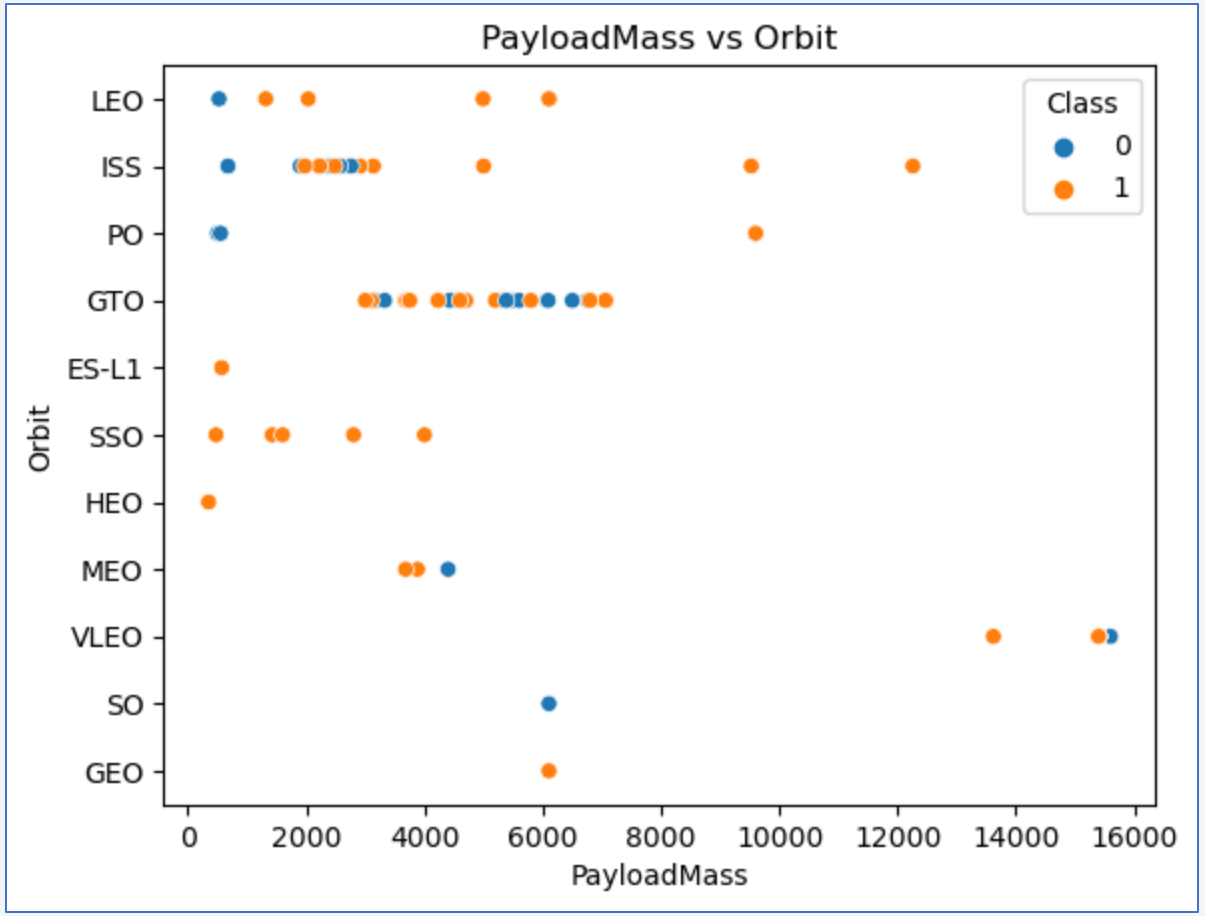


This scatter plot between flight number and orbit shows a higher probability of success for LEO orbits and no relationship with launches to GTO orbits.

Payload vs. Orbit Type

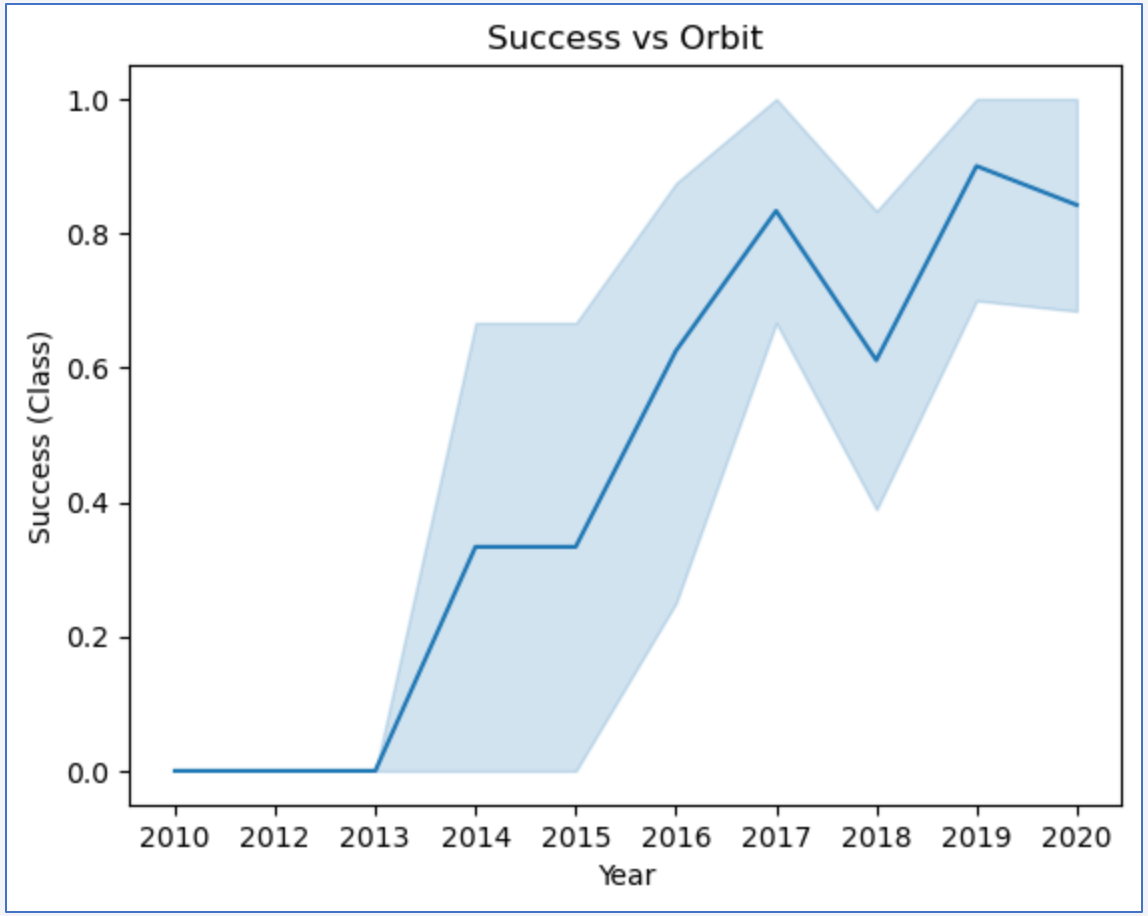
- For heavier payloads the chances for success are higher when used for LEO, ISS, and PO orbits

- However looking at GTO there appears to be no relationship between payload and chances of successful recovery of the booster.



Launch Success Yearly Trend

- This Line Chart shows the longer SpaceX has been operational, they have increased their chances at recovery and therefore reduce cost for lanuches.
- EXCEPT for 2018 which saw a drop in success rate. This may need to be further researched as to the cause.



All Launch Site Names

- Using the **Distinct** function in sql we find the Launch Sites SpaceX use
- This establishes a foundation to work from including what region we may want to use as our own base of operations

```
In [22]: %sql select distinct("Launch_Site") from SPACEXTBL
 * sqlite:///my_data1.db
Done.
```

Out[22]:

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

Launch Site Names Begin with 'CCA'

- In this sql query we focused the data using the **Like** function on launch sites with "CCA" in its name.
- Using the **Limit** function allowed us to get a glimpse of the data and decide if we want to drill farther down.

In [23]:
```
%sql select * from SPACEXTBL where LAUNCH_SITE like'CCA%' limit 5
```

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

Out[23]:

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

- Using the **Sum** and limiting the Customer to **'NASA (CRS)'** we calculated the total amount of kg SpaceX launch.
- This shows how much the largest customer of space flight as needed launch which we can use to compare cost-to-profit

In [25]:
```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer = 'NASA (CRS)'
```

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

Out[25]:

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- The **avg** and **where** functions displayed the average payload the F9 v1.1 booster launches.
- this helps give us a ballpark of how much weight we should ensure our rockets can carry successfully

In [26]:
```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
```

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

Out[26]:

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- The sql **min(Date)** and **where** functions give us SpaceX first's successful ground pad landing.
- Comparing the first launch to this point shows a timeline for us to match or exceed for our own first recovery booster.

Hint:Use min function

In [35]:
```
%sql select min(Date) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)'
```

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

Out[35]:

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using the **where** and **between** functions in sql we found which boosters were successfully recovered on the drone ship between 4000 and 6000 kg payload.
- This narrows the scope of our early boosters with a specific type and range to focus on.

In [36]:
```
%sql select Booster_Version from SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ between 4000
```

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

Out[36]:

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

Total Number of Successful and Failure Mission Outcomes

- The **group by** function separates the mission outcomes between success and failures
- This shows how viable commercial space flight is.

In [41]:
```
%sql select Mission_Outcome, count(*) from SPACEXTBL group by Mission_Outcome
```

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

Out[41]:

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

Boosters Carried Maximum Payload

- To find the boosters used for max payloads we used a **subquery**
- This is another way to focus on what type of booster we should use for specific type of mission payloads

In [39]:
```
%sql select Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

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

Out[39]:

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5

2015 Launch Records

- A sql query using **date**, **column names**, and a **subquery** displayed when and what boosters were used during failed attempts to land on the drone ship.
- Seeing where SpaceX had difficulties allows us to avoid them or spend more resources to solve those issues and gain an advantage.

In [57]:
```
%sql select substr(Date,4,2) as month, Date, Booster_Version, Launch_Site, Landing_Outcome from SPACEXTBL where Landing_Outc
```

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

Out[57]:

month	Date	Booster_Version	Launch_Site	Landing_Outcome
5-	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
5-	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Using the **count** and **order by** functions in sql to sort the many different type of Landing Outcomes.1

- This gives us a snapshot of possibilities and likelihood of outcomes if we venture into commercial space flight.

```
In [67]: %sql select Landing_Outcome, count(*) from SPACEXTBL group by Landing_Outcome order by count(*) desc
```

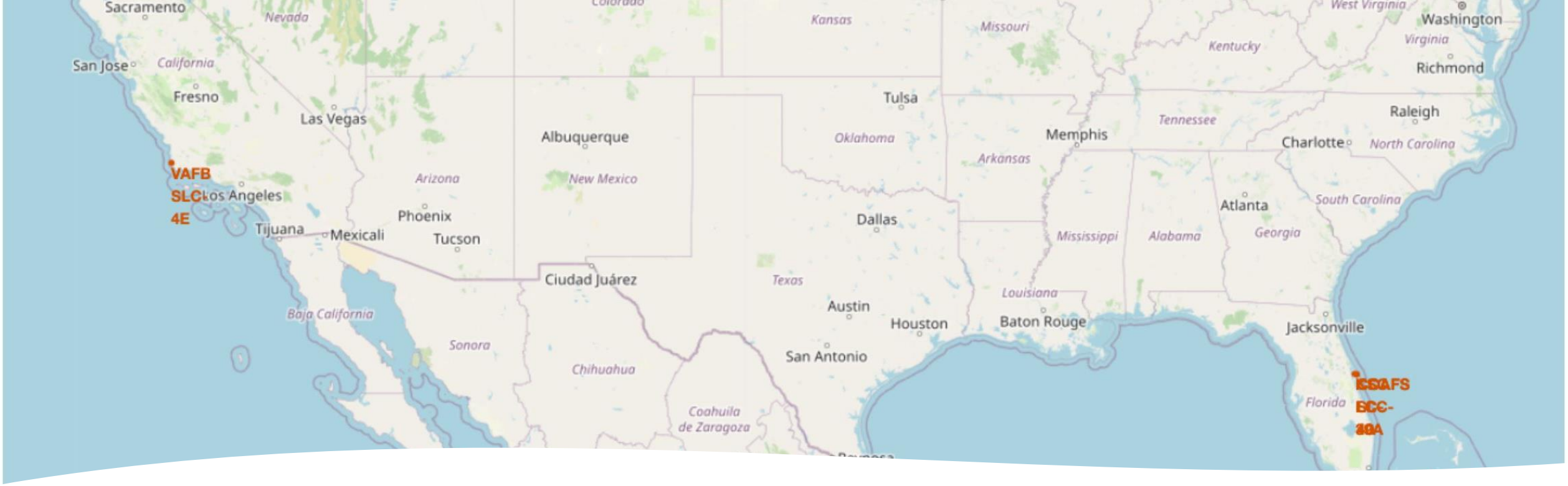
```
 * sqlite:///my_data1.db
Done.
```

Out[67]:

Landing_Outcome	count(*)
Success	38
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Failure	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1
No attempt	1

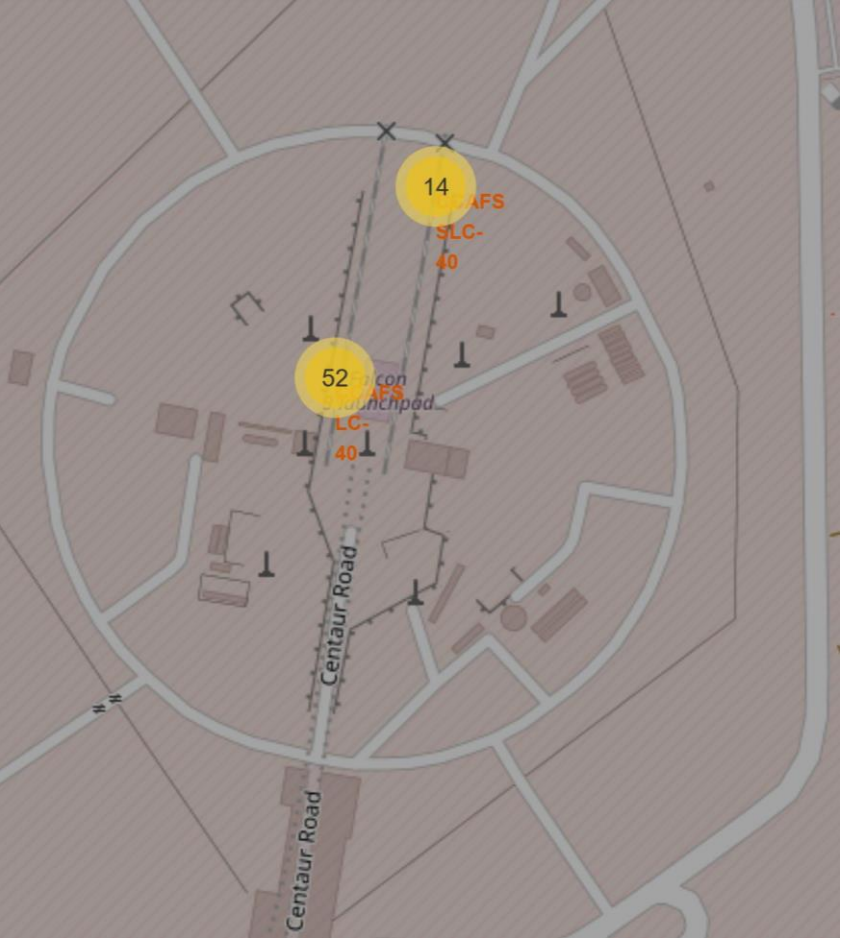
Section 3

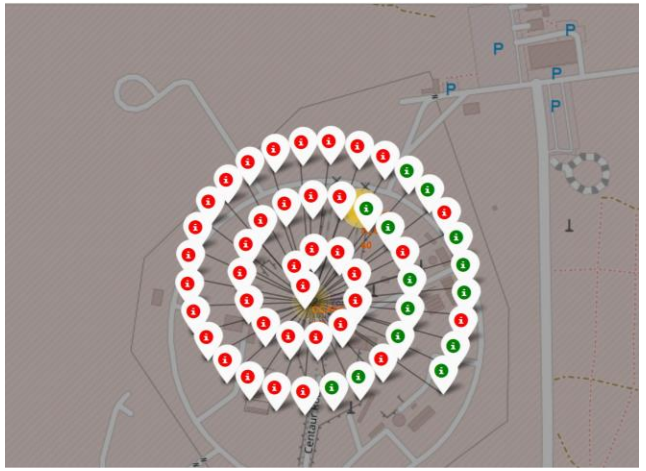
Launch Sites Proximities Analysis

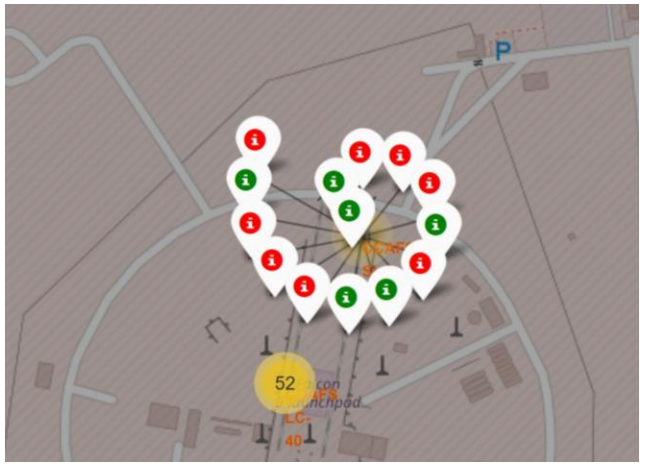


SpaceX Launch Sites

- There are four launch sites SpaceX uses, all in the United States.
  - 3 - KSC LC-39A, CCAFS SLC-40, and CCAFS LC-40 are located in Florida
  - VAFB SLC-4E
- All four sites are located on the coast. Space flight is dangerous and if there were any accidents, launching the rockets out into the sea minimizes civilian casualties.

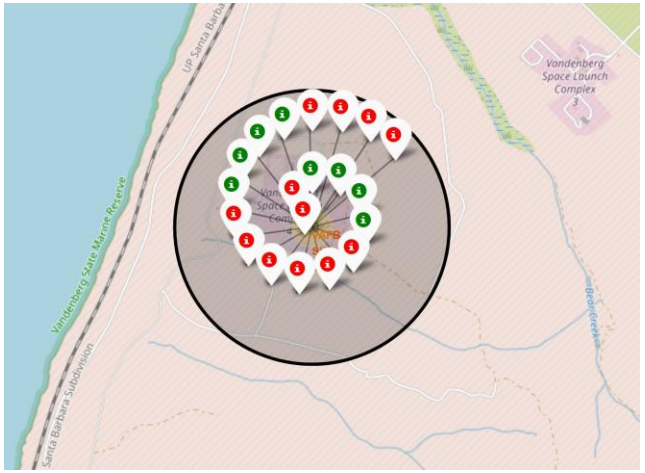




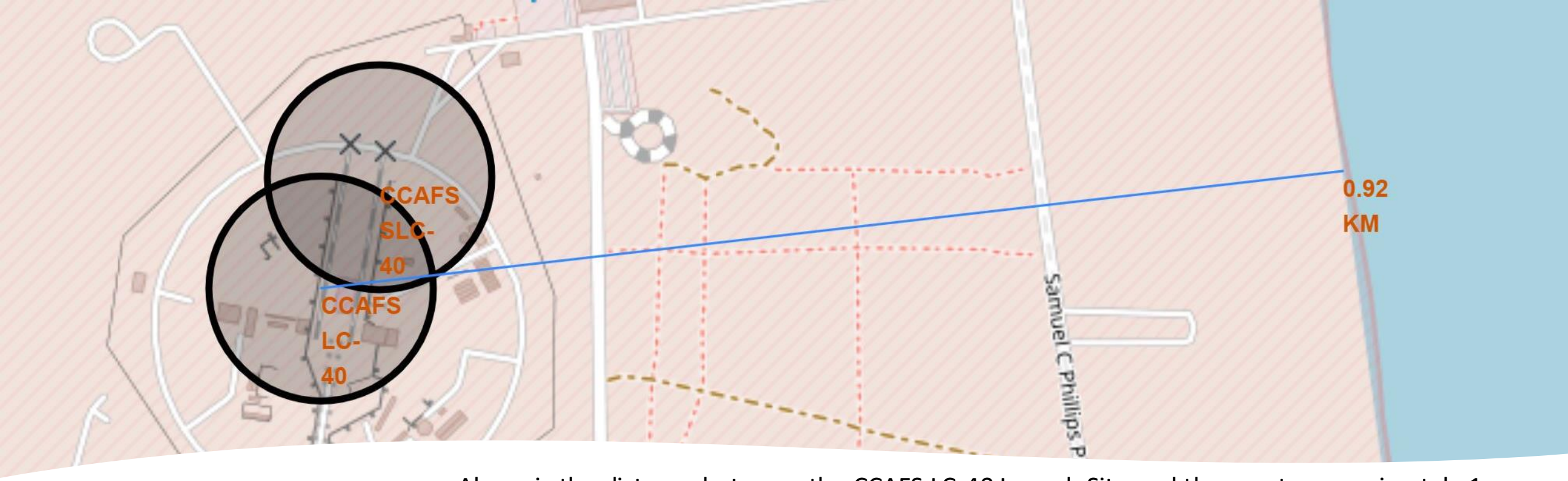


Launch Outcomes

- Left: CCAFS LC-40 & CCAFS SLC-40
  - LC-40 has the **most** launches of all sites
  - SLC-40 has the **fewest** launches of all sites

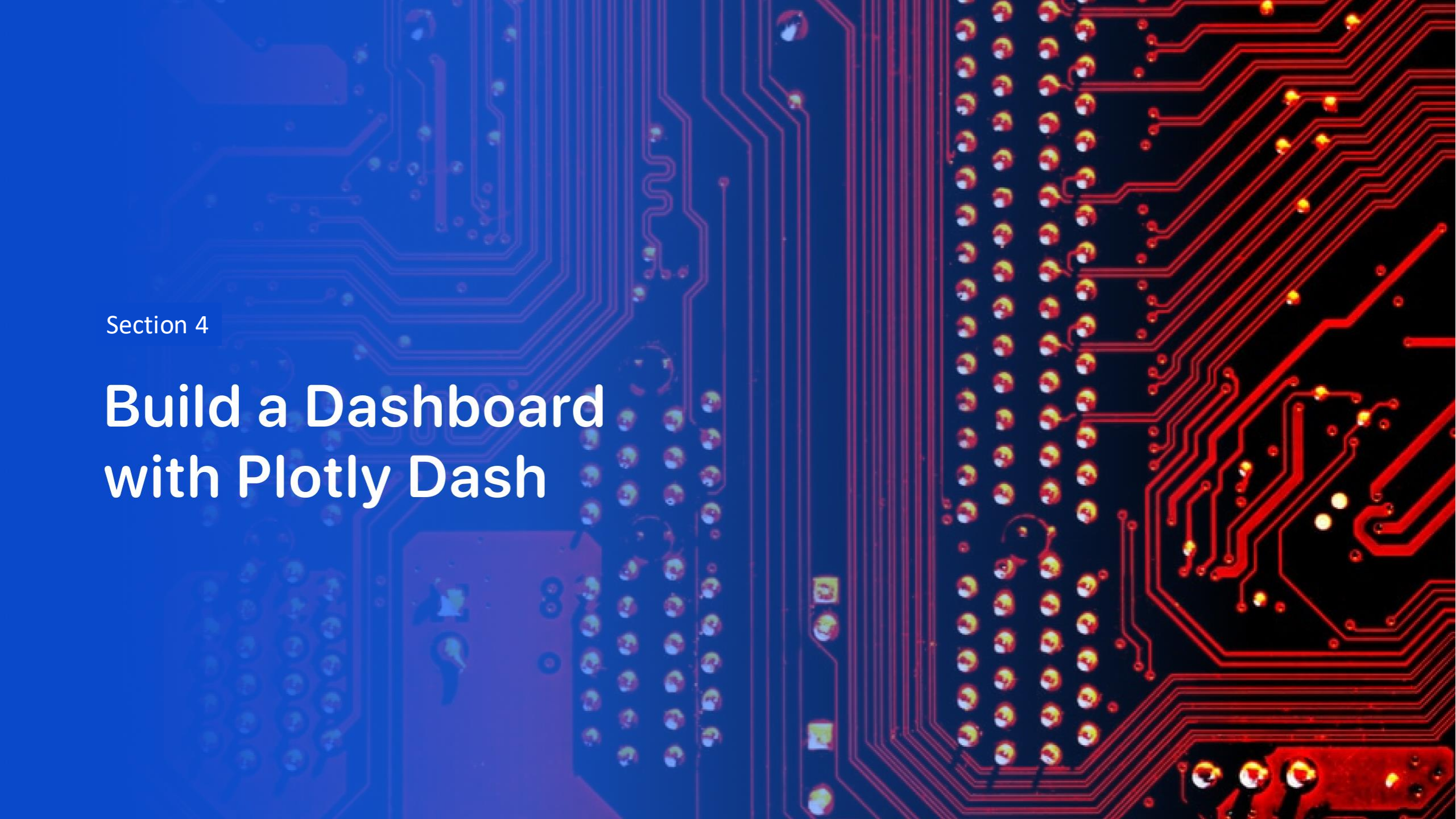


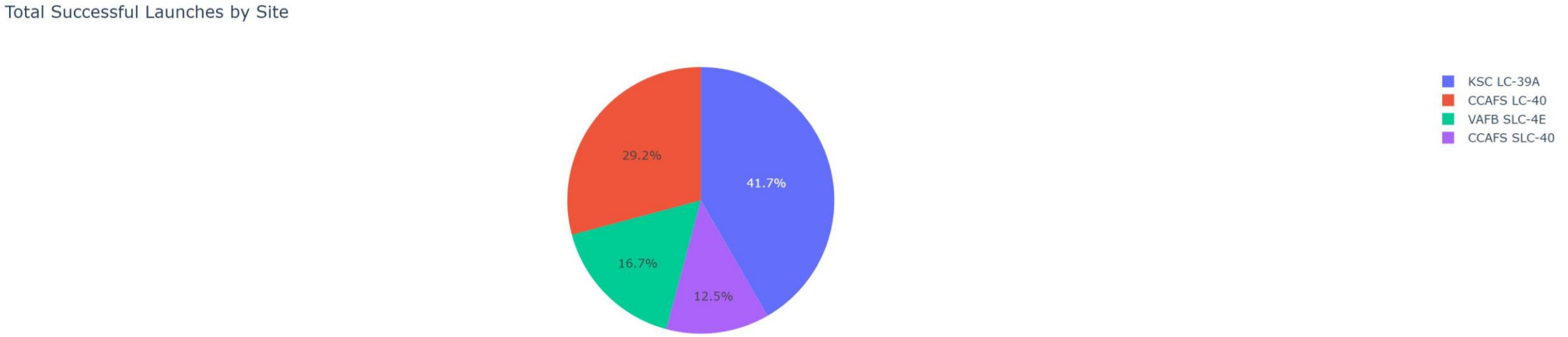
- Left Bottom - VAFB SLC-4E
  - Is the only launch site on the west coast
- Right Bottom - KSC LC-39A
  - 2nd most launches after LC-40



Distance to Points of Interest

- Above is the distance between the CCAFS LC-40 Launch Site and the coast, approximately 1 kilometer.
- Other Points of Interests are:
  - Cape Canaveral [28.3782, -80.59309] - a city 20.5 kilometer's from both CCAFS launch sites
  - The closest railway line [28.57117 -80.58545] is 1.25 kilometer's from the launch sites
    - The railway helps deliver equipment and personnel
  - The closest highway [28.55744 -80.79831] is 21.5 kilometers
    - Due to high volume of traffic on highways, the launch sites were built far away for safety reasons.

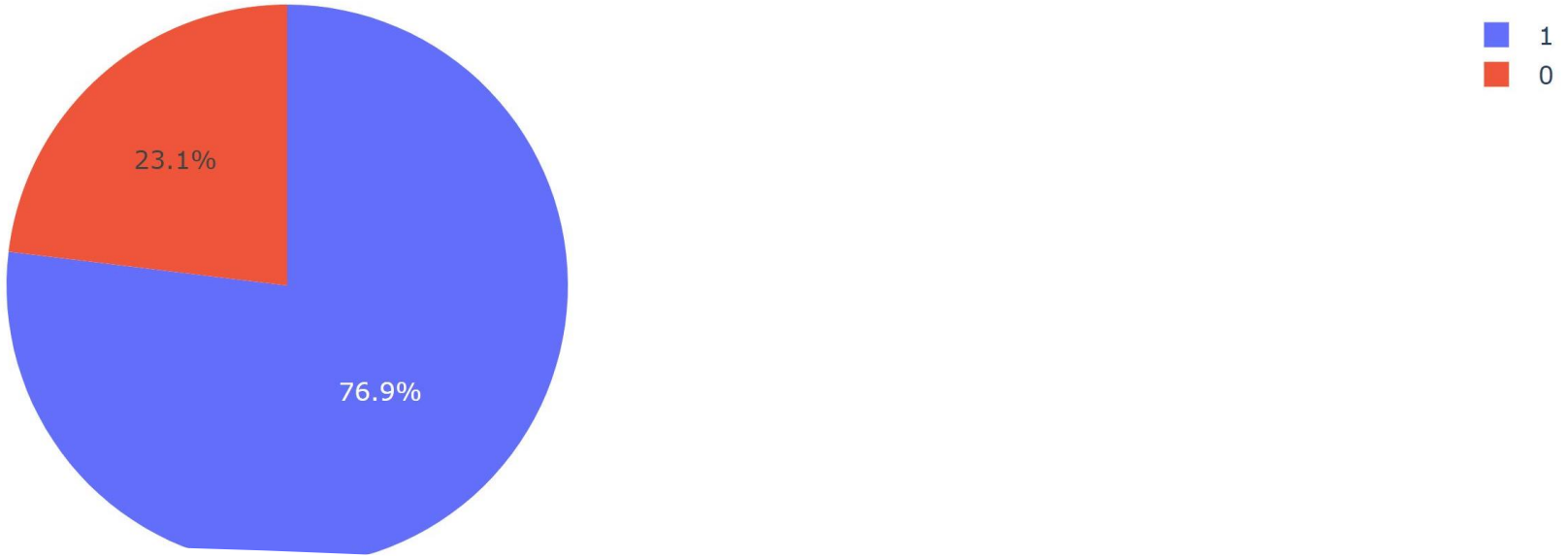

Section 4
Build a Dashboard with Plotly Dash



Total Successful Launches by Site

- This Dashboard shows despite CCAFS LC-40 launching the most rockets,
- KSC LC-39A has the most successful recoveries at 41.7 %
- CCAFS SLC-40 has the fewest recoveries at 12.5%

Total Success Launches for site KSC LC-39A

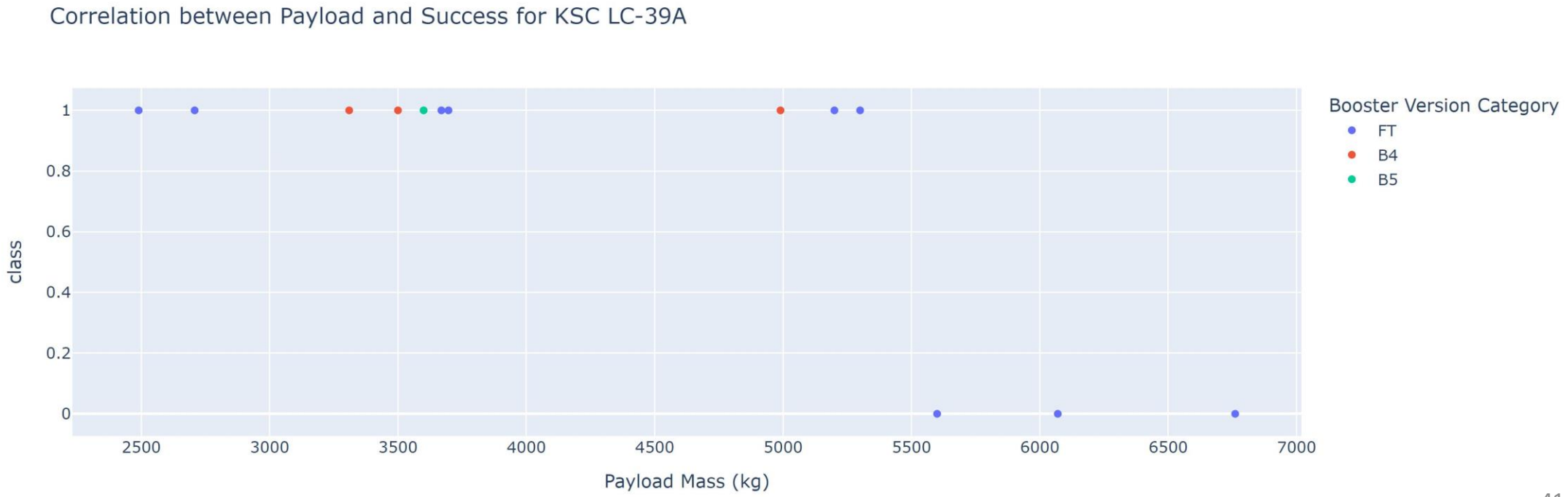


KSC LC-39A Success Rate

- Observing from the all site dashboard we noticed KSC LC-39A had the highest success rate among the sites
- Selecting it specifically we are able to see it has a success rate of over 76%!
  - The Dashboard shows in order of success which sites we should use for launches.
  - Further focusing down tells us CCAFS LC-40 is the best candidate to start our launches.

Payload vs Success

- Our second Dashboard shows correlation between payload and success.
- We narrowed the payload range to just those kg payloads KSC LC-39A was used.
- This shows us the large success rate correlates to lighter payloads. This stie and a lower payload is a good foundation for our start in commercial space flight



Section 5

Predictive Analysis (Classification)

Classification Accuracy

- We used best_score for each model after training and testing the samples.
- Tree Classifier is the most accurate at 89%
- Tree Classifier will be used in the future to predict if SpaceX flights will recover their boosters.

Find the method performs best:

51]:

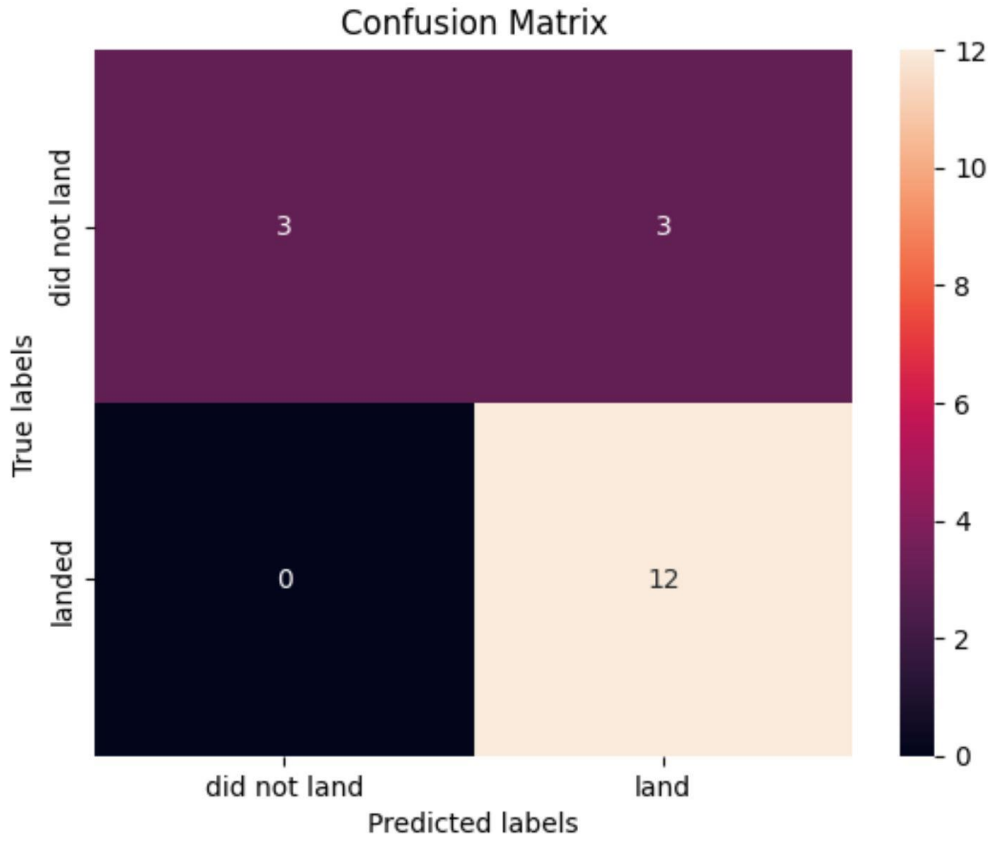
```
print("logistic regression's accuracy :",logreg_cv.best_score_)
print("SVM'accuracy :",svm_cv.best_score_)
print("Tree's accuracy :",tree_cv.best_score_)
print("KNN's accuracy :",knn_cv.best_score_)
print('Tree Classifier is the most accurate at:', tree_cv.best_score_)
```

```
logistic regression's accuracy : 0.8464285714285713
SVM'accuracy : 0.8482142857142856
Tree's accuracy : 0.8892857142857142
KNN's accuracy : 0.8482142857142858
Tree Classifier is the most accurate at: 0.8892857142857142
```

Confusion Matrix

Here is the a visualization of Tree Classifier's accuracy using a Confusion Matrix

In [37]:
```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- Observing the data there is an opportunity to patriciate in commercial space flight
- KSC LC-39A gives us the highest possibility of recovering the booster thus saving money.
- The average payload NASA request per launch is 2928.4kg
- The best orbits to deliver payload are: ES-L1, GEO, HEO, and SSO
- To best predict if SpaceX will be able to recover their booster, we should use the tree classifier model.

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
- SpaceX Data Collecing API 01a
- SpaceX - Data Collecting - 01b Web Scrapping
- SpaceX Data Wrangling 02
- SpaceX EDA with Data Visualization 03
- SpaceX EDA with SQL – 04
- SpaceX Folium Interactive Map - 05
- SpaceX - Ploty Dashboard - 06
- SpaceX Machine Learning Model - 07

Thank you!