IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

John Fessler
August 1st, 2024

Outline

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

Executive Summary

- Summary of methodologies:
  - Collection: the SpaceX launch data was collected and scraped from the SpaceX API and Wiki.
  - Data was wrangled, cleaned, and stored with the Python library Pandas.
  - Exploratory data analysis (EDA) and visualizations executed with SQL queries, Seaborn graphs, Pandas graphs and data manipulations, mapping done with Folio, and a dashboard built through Dash with Plotly Express used from graphs.
  - Predictive Analysis was completed with GridPlotCV using Logistic Regression, Support Vector Machine, Decision Tree, and k-Nearest Neighbor algorithms.
- Summary of all results:
  - EDA showed many insights about mission success. Most importantly: success increased with continued test flights, launch site KSC LC-39A was the most consistent site for successful missions, launch site VAFB SLC-4E is the only site with successful max payloads, and the orbit does impact the successful recovery of the first stage.
    - At site KSC LC-39A, no mission with a payload over 5300kg landed successfully.
  - Predictive Analysis revealed that most algorithms performed similarly in determining whether a launch was successful in landing its first stage.
    - Best algorithm: the Decision Tree algorithm achieved a 83.33% accuracy on test data and a top accuracy of 86.25%.

Introduction

- We as prospective company **Space Y** are looking to understand our target competitor **SpaceX.**
  - They are able to undercut their other competitors by 10s of millions of dollars per mission due to their rocket's first stage recovery ability.
  - We must build a mission capable rocket system to beat the SpaceX price point.
    - Using their data, we need to circumvent years of research, development, and testing
- By answering the following questions, we will jumpstart our development of a cheap, high capacity rocket to counter the SpaceX control of the market!
- Questions to be answered:
  - What factors affect the successful recovery of the rocket's first stage?
  - Can we predict whether a mission will be successful?

Section 1

Methodology

Fessler | August 2024

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

Two forms of data collection

SpaceX API

1. Identify the API calls to collect the raw data and get the URL from:
   - https://api.spacexdata.com
2. Import the requests, json, and pandas libraries
3. Using the requests library, we use the get() method to request the raw data in json form
4. From the json library, we will load the request with the loads() method
5. Finally from the pandas library, the json_normalize() method loads the json file into a data frame

If the dataframe is too large, repeat steps 3-5 for specific subsets of the data

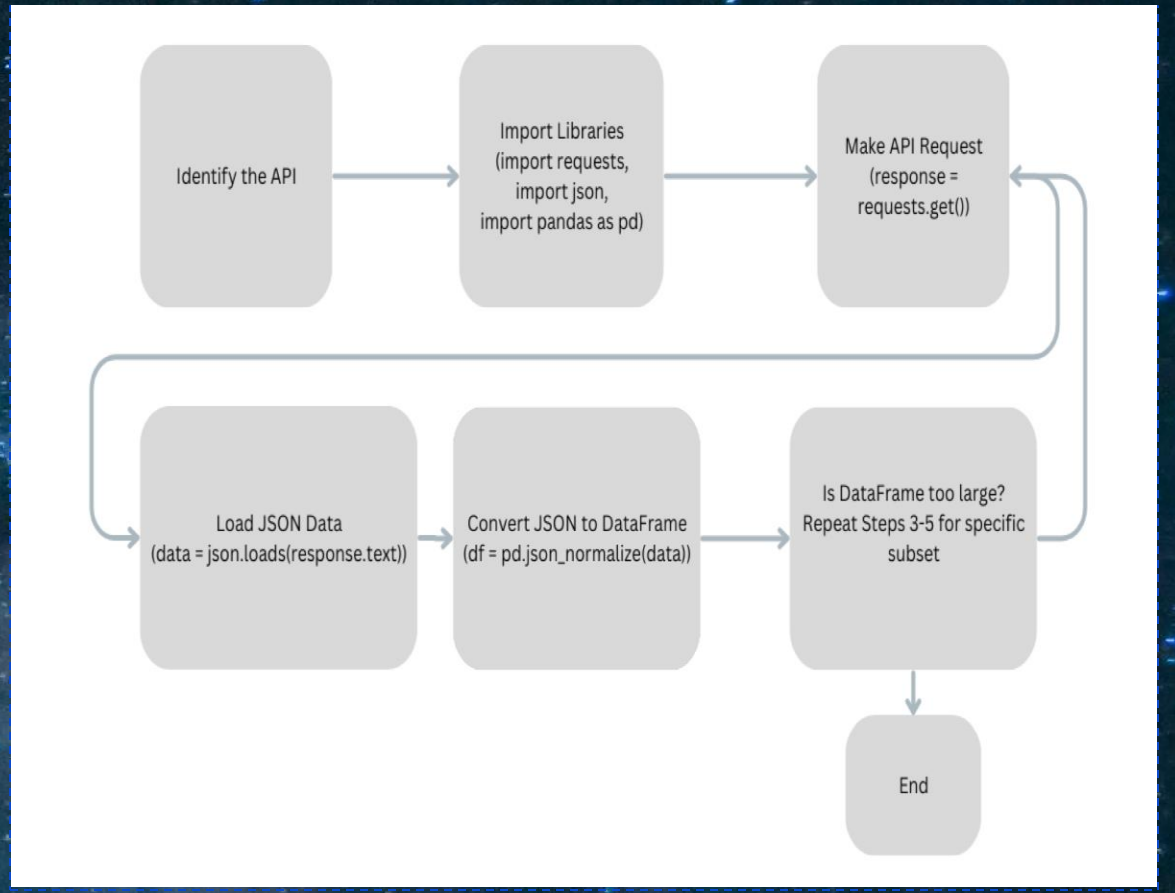
Web Scraping

1. Identify data set on the internet and get URL from:
   - https://en.wikipedia.org
2. Import requests, BeautifulSoup, and pandas libraries
3. Using the requests library, we use get() method on the URL
4. Store the BeautifulSoup parser built with the requested data
5. Using the find_all('table') method, we isolate the tables into a list and then select the desired data with slicing
6. Again using the find_all('th') method, we gather the column names, and then iterate through gathering all info for 'tr' and 'td', the rows and elements. All stored in a dictionary
7. Finally, all gathered data is loaded into a dataframe

Data Collection – SpaceX API

- API calling process to acquire the data set from the SpaceX REST API

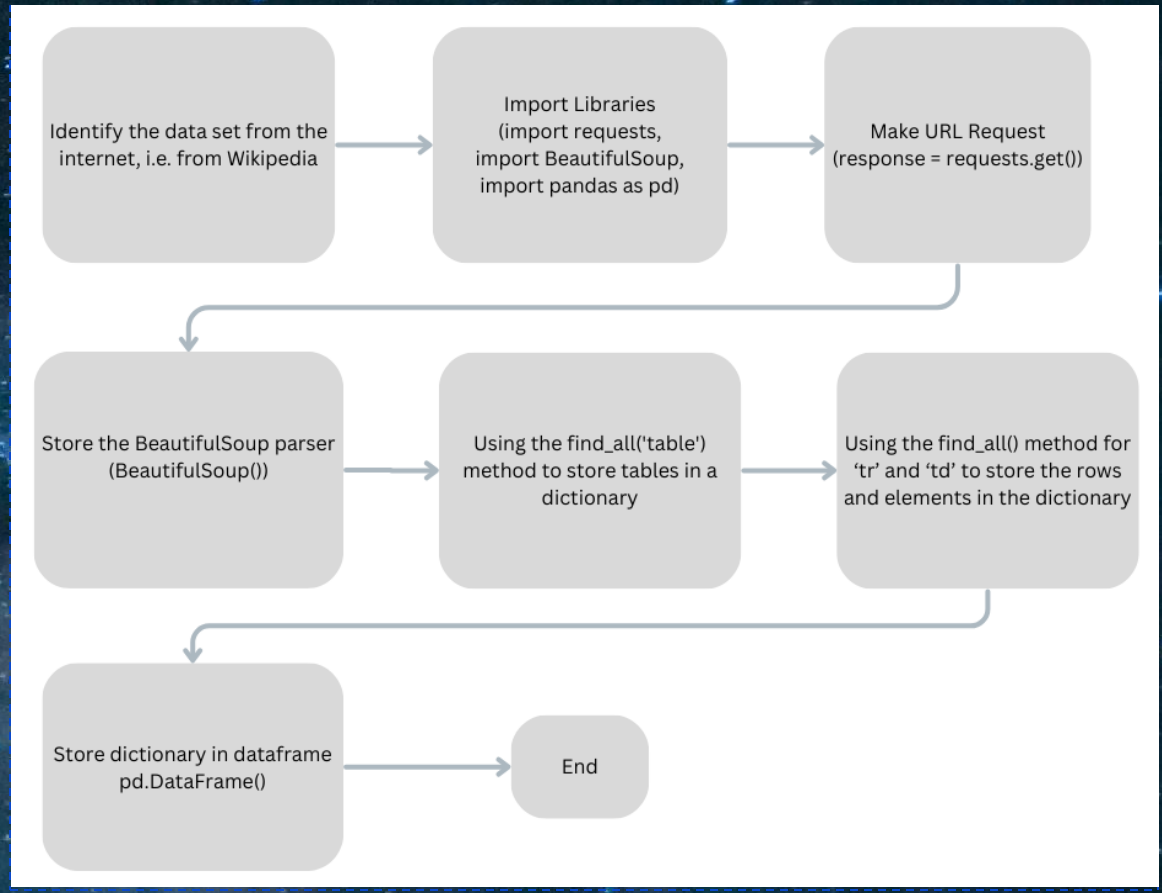
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/169bc145ae58c1c57a72fbcb719d56b83b44a720/Lab1-jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

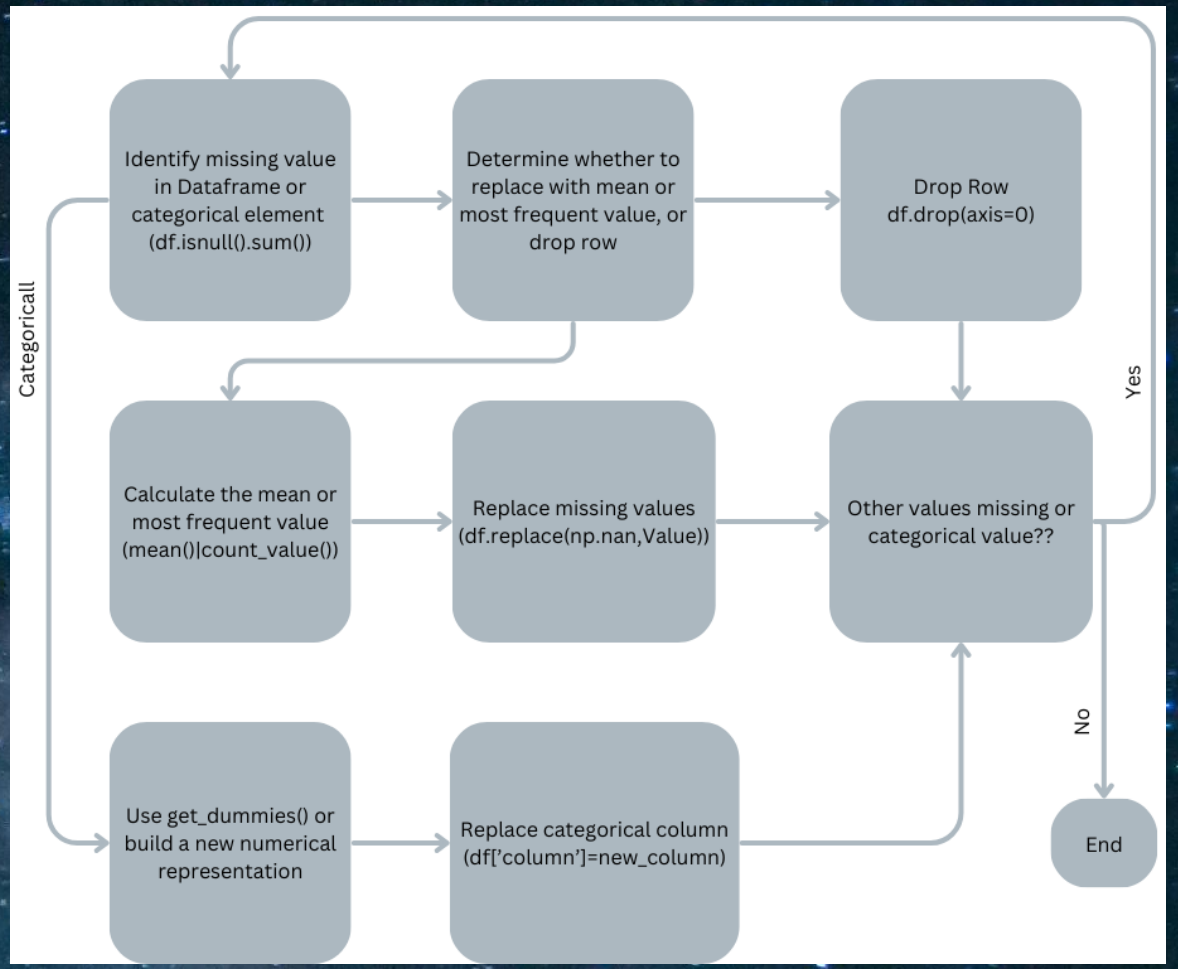
- Web scraping process to acquire the data set on the SpaceX Wikipedia page

- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/169bc145ae58c1c57a72fbcb719d56b83b44a720/Lab2-jupyter-labs-webscraping.ipynb



Data Wrangling

- It is the process of forming a data set that is effective and straightforward to manipulate and query for EDA
- The process includes the replacement of empty values and categorical values with relevant numerical values
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/169bc145ae58c1c57a72fbcb719d56b83b44a720/Lab3-labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- The graphs were chosen to better understand the SpaceX rocket launch success as affected by different factors, including with respect to time, which would reflect their continued research and development.
  - Flight Number vs Launch Site|Payload Mass|Orbit
  - Payload Mass vs Launch Site|Orbit
  - Orbit vs Success bar chart
  - Success Rate vs Years
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/169bc145ae58c1c57a72fbcb719d56b83b44a720/lab5-edadataviz.ipynb

EDA with SQL

- SQL Queries in the form: SELECT values FROM database WHERE conditions
  - Queries were most importantly used to gather the boosters, payload sizes, launch locations, dates of successful and failed launches, and the frequency of failed first stage landings.
  - Example) Earliest successful ground landing:
    - SELECT MIN(DATE) FROM SPACEXTABLE WHERE Landing_Outcome = "Success (ground pad)";

- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/1cb1ab3355660ba49a4e8221044bfa53687b5b8d/Lab4-jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

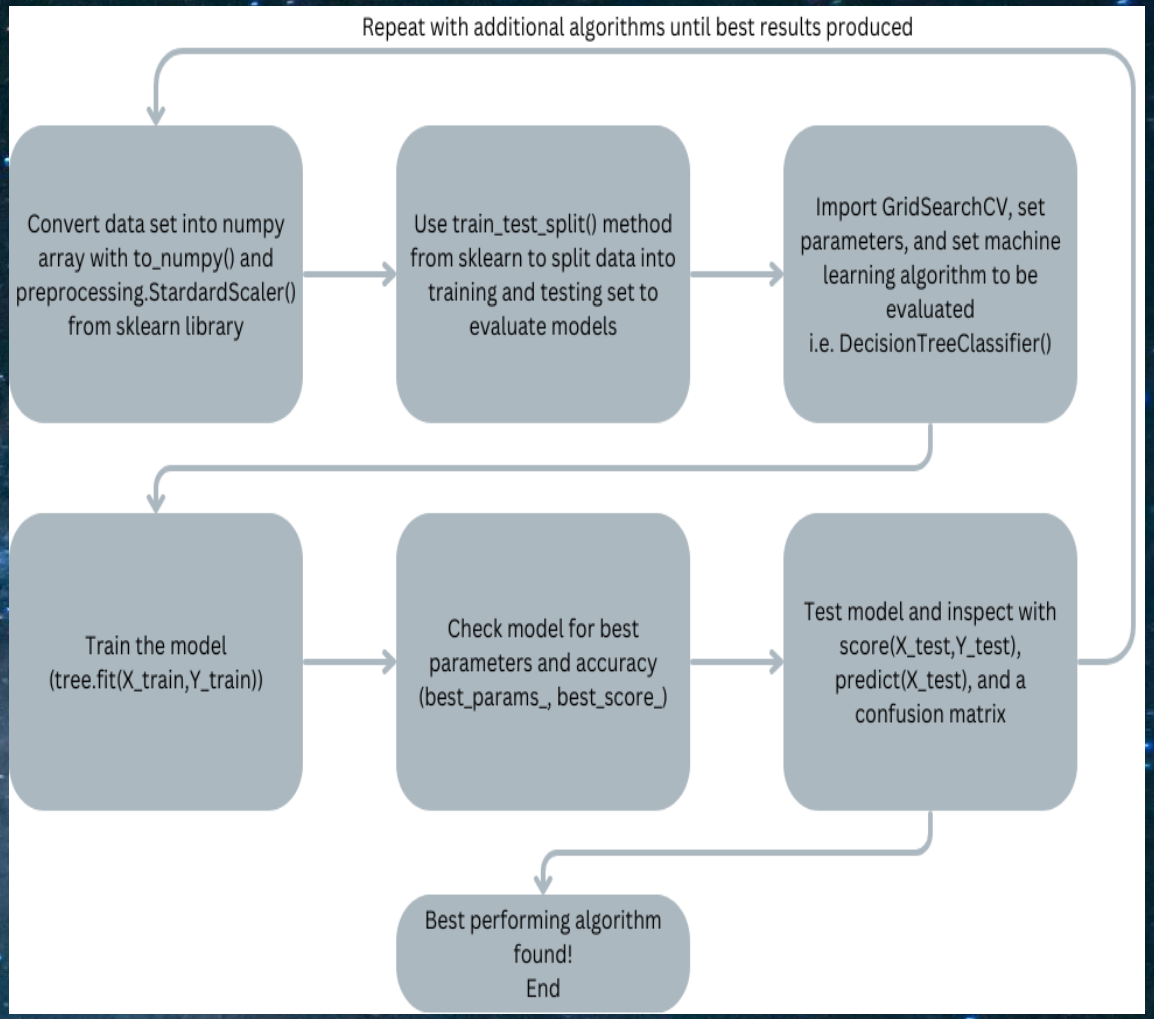
- Created circles around the launch sites and marked them
- Marked each site with colored mark for successful and failed launches in a markercluster
- Added mouse position for estimating longitude and latitude
- Drew lines marking the closest ocean, highway, train track, and city to VAFB SLC-4E Site
- These objects were added to understand spatially the position on the SpaceX launch sites, and how this might be affecting the success of the rocket landings
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/1cb1ab3355660ba49a4e8221044bfa53687b5b8d/Lab6-lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- The Dashboard contains two graphs
  - The first, pie chart, is controlled by user input of launch site via dropdown menu, and the second is controlled by user input of both launch site and payload range via slider.
  - The pie chart depicts success of each site, and the individual site pie charts show success and failure rates there.
  - The scatter plot shows the success and failure of all or individual sites with respect to their payload size.
- These interactive plots allowed for a much more granular understanding of how each site and different sized payloads impacted mission success.
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/1cb1ab3355660ba49a4e8221044bfa53687b5b8d/lab7-spacex_dash_app.py

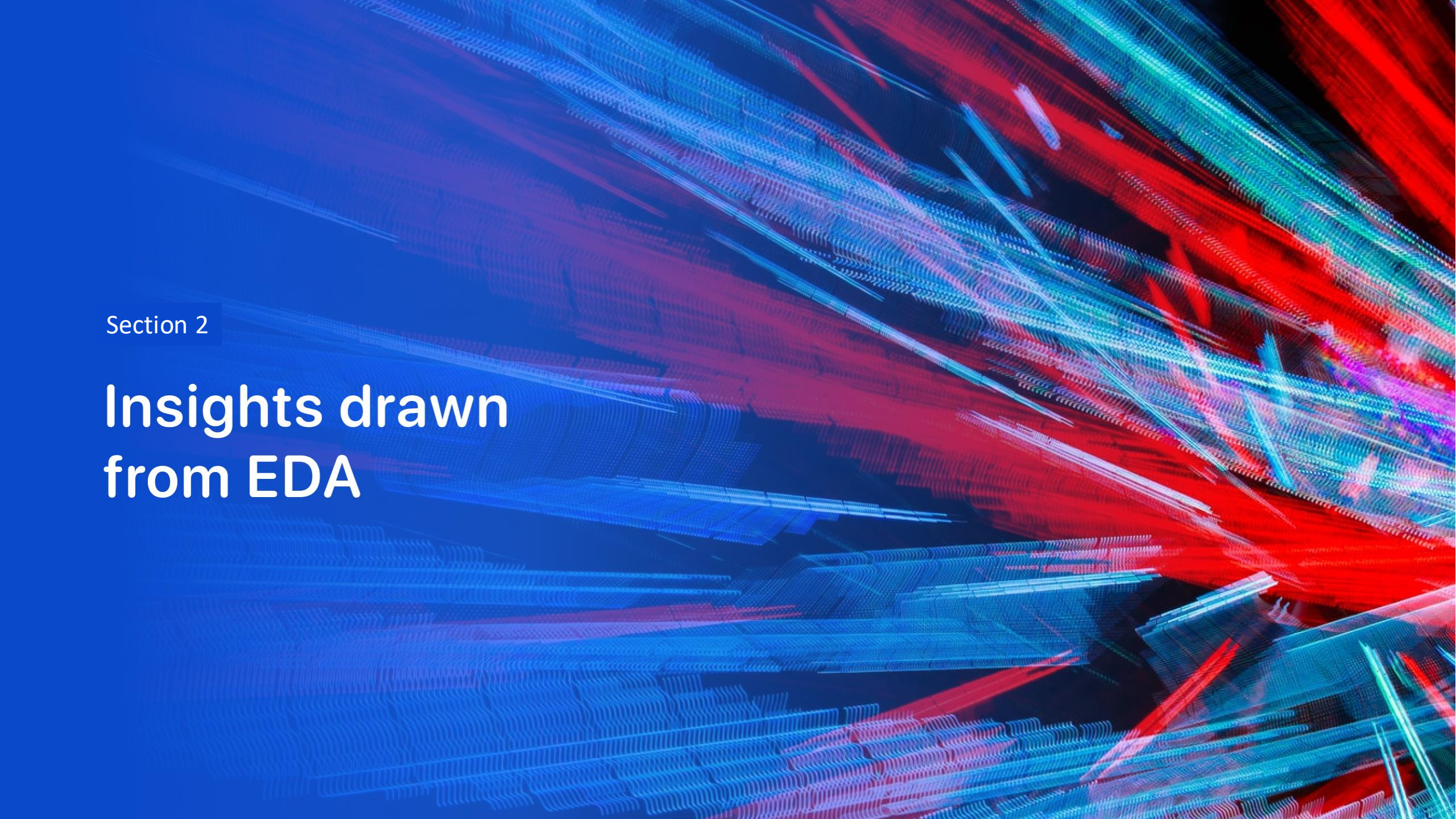
Predictive Analysis (Classification)

- Through an iterative process the best classification model was developed
- GridSearchCV allowed for the training of the models to find the best parameters for the fit.
- The accuracy score was used to separate the models
- https://github.com/jffessler/IBM_Data_Sci_Cert_Final/blob/1cb1ab3355660ba49a4e8221044bfa53687b5b8d/Lab8-SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

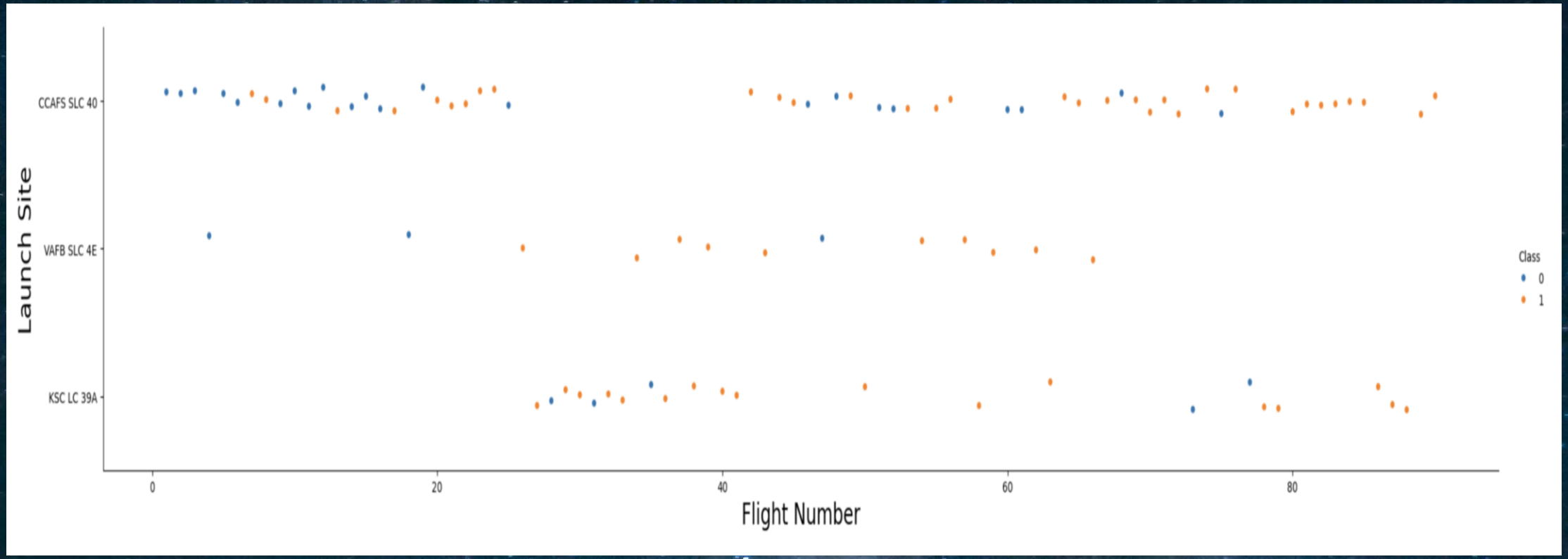


Results

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

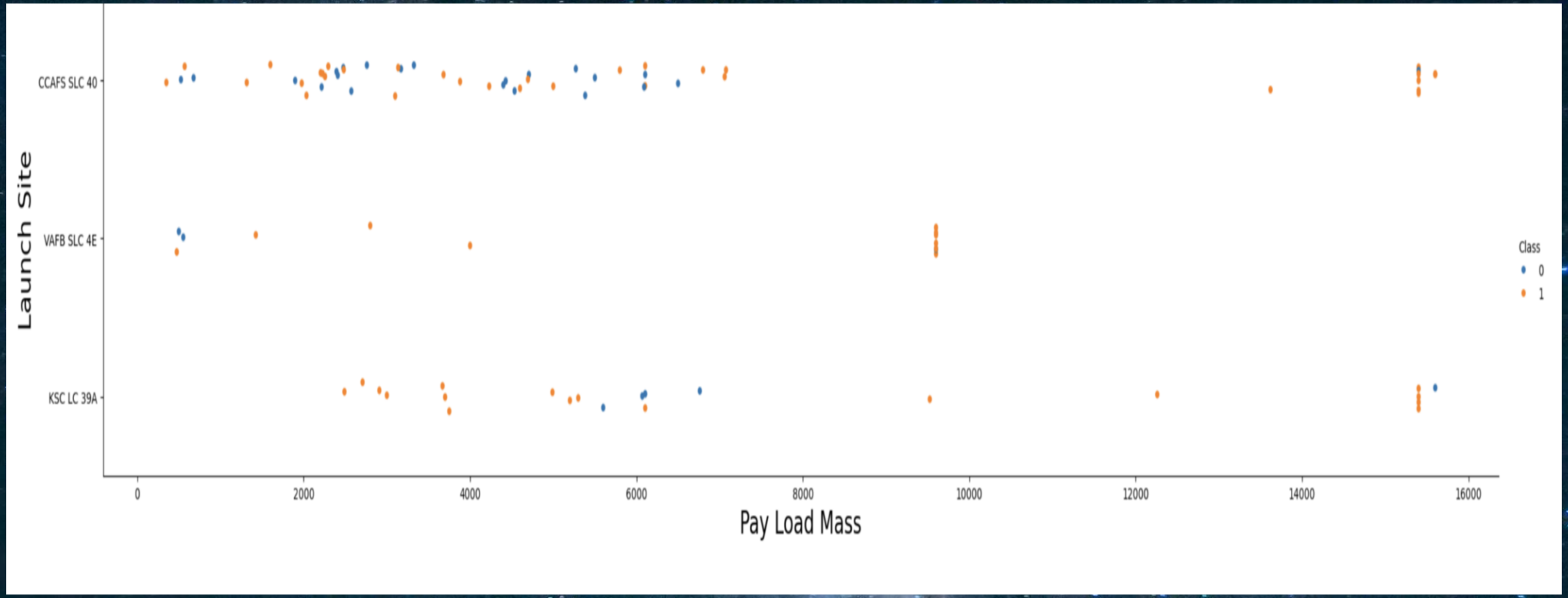

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



There are no major variations between the frequency of success at the different sites. There is a notable halt to the use of the CCAFS SLC 40 between launch 25 and 40, which is the first time that we see site KSC LC 39A be used. There is what appears to be an overall increasing trend of success as the flight number increases.

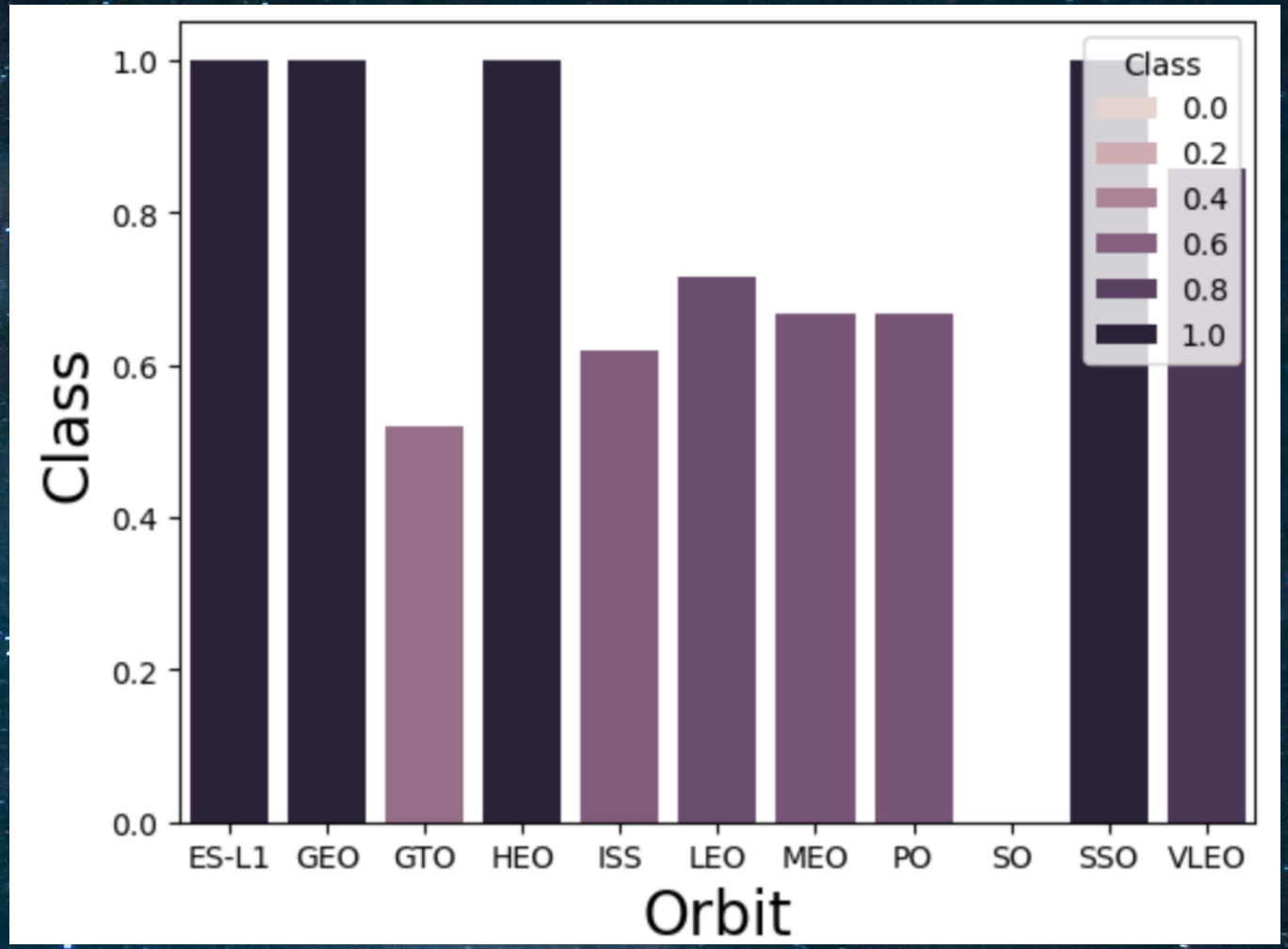
Payload vs. Launch Site



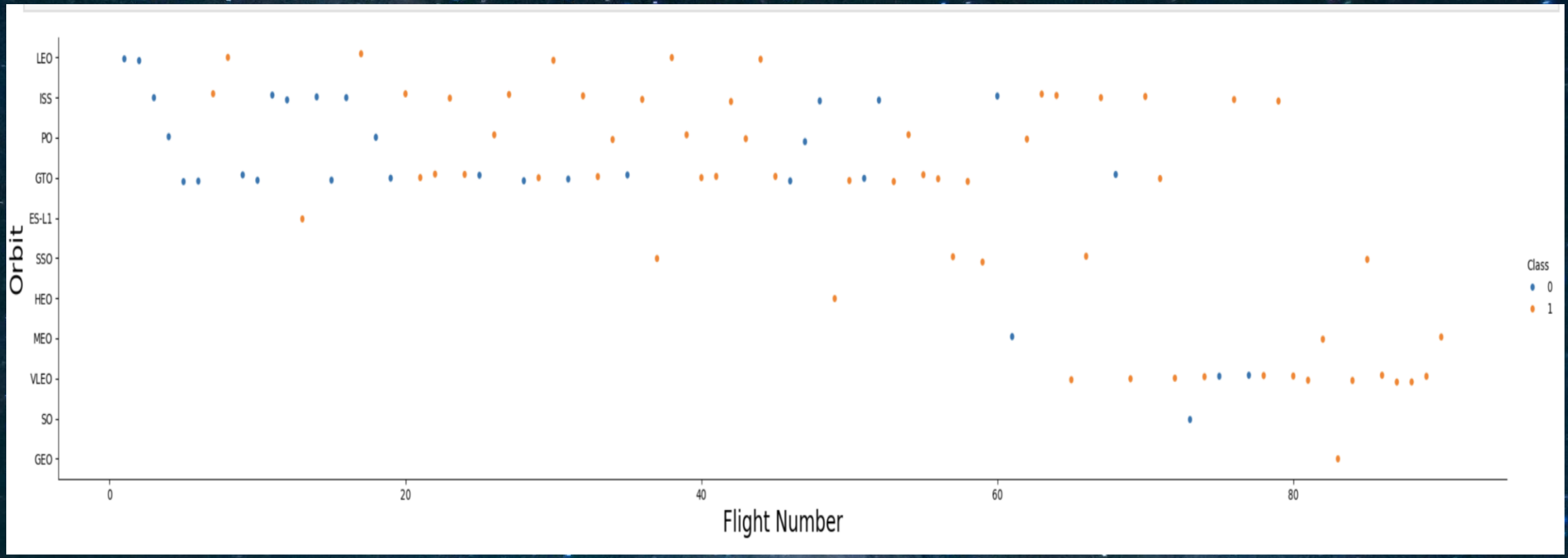
Payload mass is pretty evenly distributed across the launch sites, except site VAFB SLC 4E does not handle any payloads larger than 10000kg. Additionally, the larger payloads appear to have a higher success right.

Success Rate vs. Orbit Type

We observe a range of success rates. However, this is a very misleading representation of the data. For example ES-L1 (100% success) and SO (0% success), while GTO has 27 missions with a 40-50% success rate. It is good to note though that VLEO has 14 missions with approximately a 90% success rate, this appears to have some significance statistically.

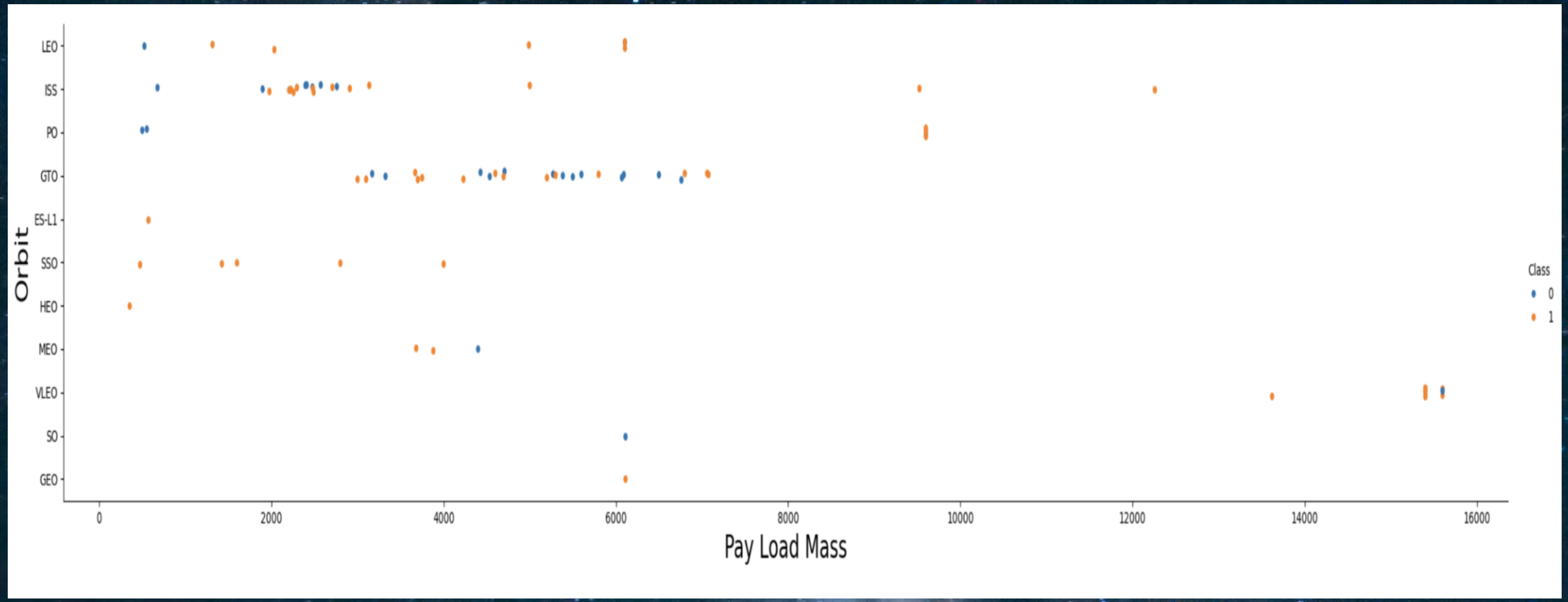


Flight Number vs. Orbit Type



There does not appear to be much relationship between orbit and flight number or success. There is a distinct beginning of launches to VLEO in the higher flight numbers, indicating later in the research. These launches also have a high success rate.

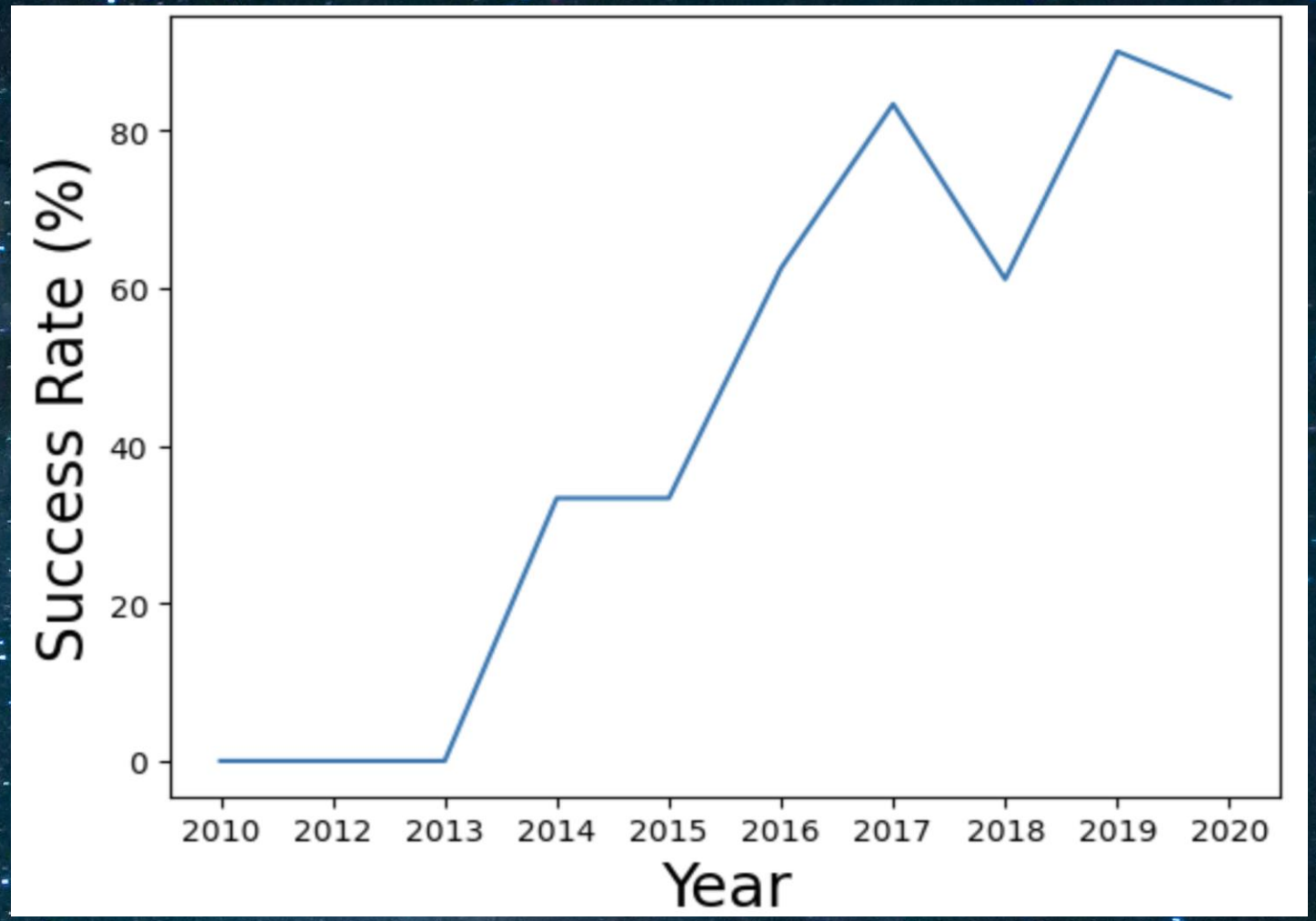
Payload vs. Orbit Type



The heavier the payload the higher the successful landing rate. The only orbits with large payloads are ISS, PO, and VLEO. Success at lower payload masses is not consistent, but SSO stands out at 100% success.

Launch Success Yearly Trend

The success rate progressively increases with time. This was already indicated by the flight numbers. With continued research and testing their success rate rapidly increased, and this point is at least 80%.



All Launch Site Names

```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTABLE
```

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

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

Within this data set we find the four launch sites using the DISTINCT statement

Launch Site Names Begin with 'CCA'

With the use of the LIKE statement we find all rows with a launch site beginning with 'CCA' and LIMIT our results to 5

```
%sql SELECT * FROM SPACEXTABLE 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_
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (p
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 (p
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	N
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	N
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	N

Total Payload Mass

By selecting the payload and using the WHERE statement, we find the mass of all payloads from the customer NASA.

```
%sql SELECT PAYLOAD_MASS__KG_ FROM SPACEXTABLE WHERE "Customer"="NASA (CRS)";
```

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

PAYLOAD_MASS__KG_
500
677
2296
2216
2395
1898
1952
3136
2257
2490
2708
3310
2205
2647
2697
2500
2495
2268
1977
2972

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS "Avg Payload Mass for F9 v1.1" FROM SPACEXTABLE WHERE "Booster_Version"="F9
```

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

Avg Payload Mass for F9 v1.1
2928.4

The AVG() and WHERE statement allowed us to determine the average payload mass of this specific rocket.

First Successful Ground Landing Date

```
%sql SELECT Date AS "First Successful Ground Landing" FROM SPACEXTABLE WHERE Date = (SELECT MIN(DATE) FROM SPACEX
```

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

First Successful Ground Landing
2015-12-22

```
%sql SELECT MIN(DATE) FROM SPACEXTABLE WHERE Landing_Outcome = "Success (ground pad)";
```

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

MIN(DATE)
2015-12-22

Here we determine the first successful ground landing with a MIN() and WHERE statement. I overcomplicated my first query by using a subquery, and I then realized that the more simple form.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTABLE WHERE (Landing_Outcome = "Success (drone ship)")
```

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

Booster_Version	PAYLOAD_MASS__KG_
F9 FT B1022	4696
F9 FT B1026	4600
F9 FT B1021.2	5300
F9 FT B1031.2	5200

There are only four occurrences of successful 'drone ship' landings when the original payload was between 4000 and 6000kgs. This query uses the AND statement for a multi-clause WHERE statement.

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT COUNT(CASE WHEN Mission_Outcome = 'Success' THEN 1 END) AS Success, COUNT(CASE WHEN Mission_Outcome L
```

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

Success	Failure
98	1

With the COUNT() statement we find the number of successes and failures in the Mission Outcomes columns. This is referring to the overall mission and not just the recovery of the first stage of the rocket.

Boosters Carried Maximum Payload

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MAS
```

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

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

Here we use a subquery to find the boosters that have carried the maximum payload. I displayed the payload mass to verify that the query had executed correctly.

2015 Launch Records

```
%sql SELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE (sub
```

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

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

Using SUBSTR() to isolate the dates of the launches, we determine that there are only two missions that had failed drone ship landings in January and April in 2015.

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

```
%sql SELECT Landing_Outcome AS Outcomes, COUNT(Landing_Outcome) AS Count FROM SPACEXTABLE WHERE Date BETWEEN '201
```

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

Outcomes	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

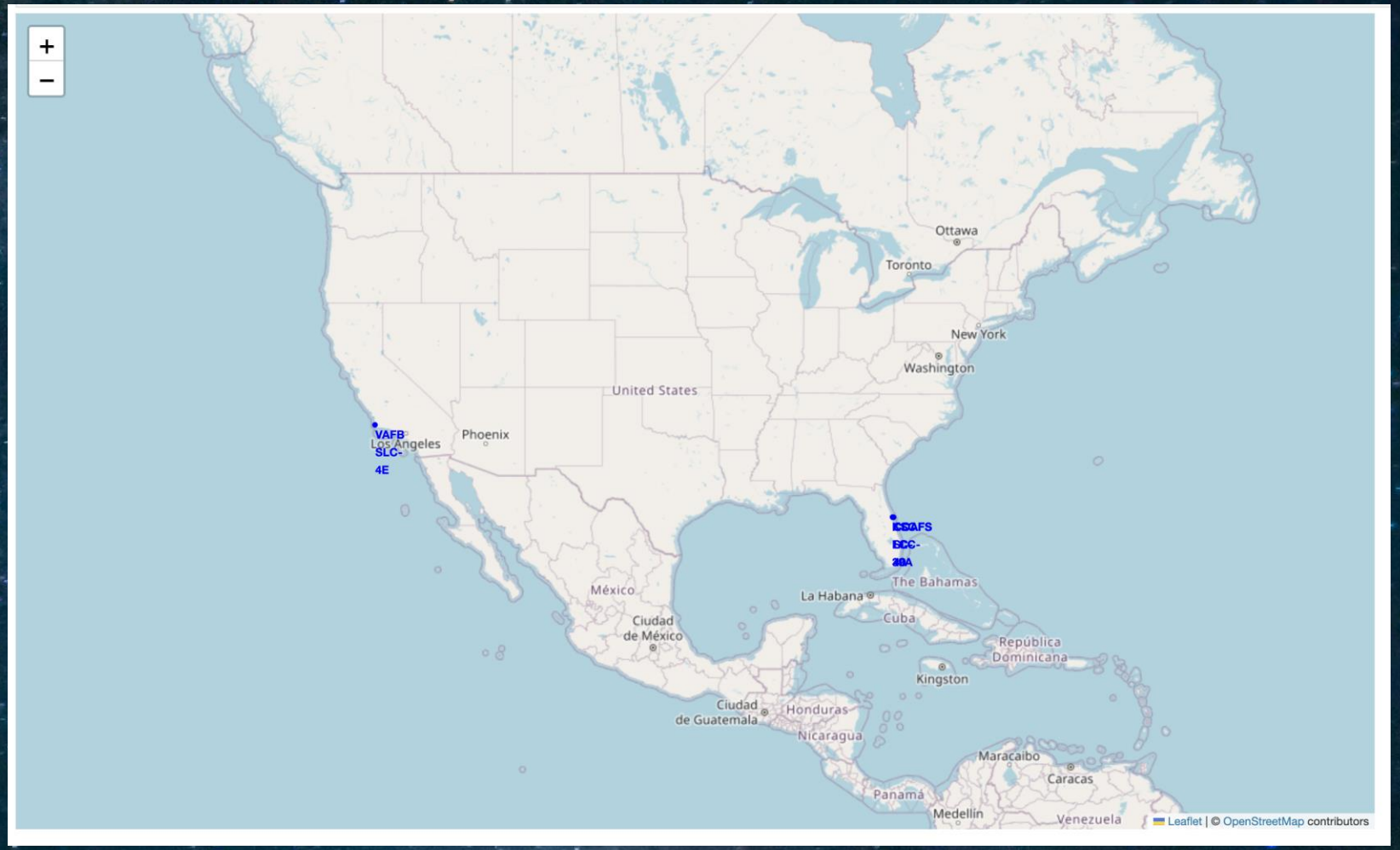
Through the COUNT() and BETWEEN statements, we determine the number of launches and their outcomes between 2010-06-04 and 2017-03-20. We find that the number 'No attempts' is nearly a third of the missions. Seems like during this time they were not confident in attempting first stage landings consistently.

Section 3

Launch Sites Proximities Analysis

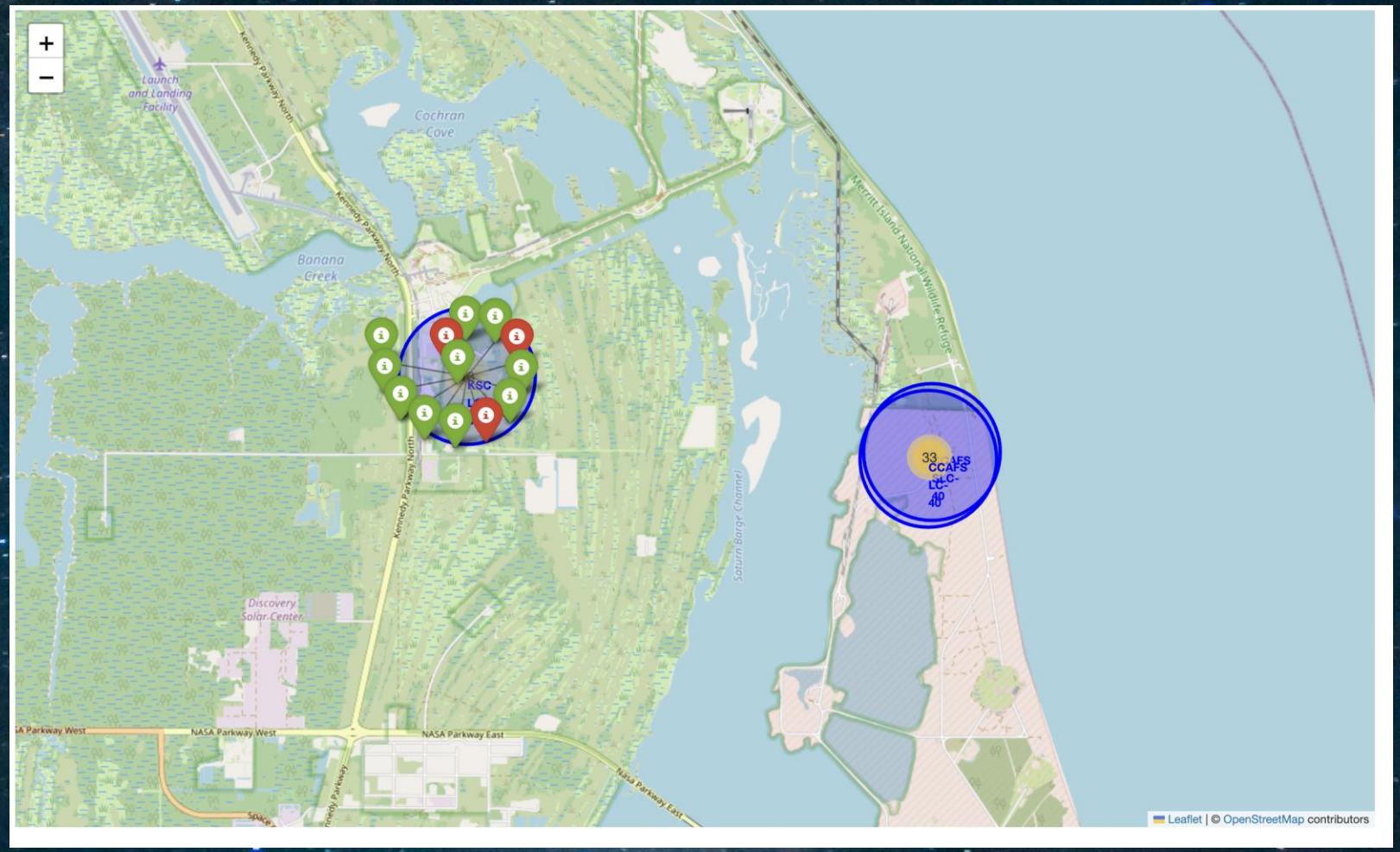
Launch Site Map

In Folium we have marked out the four launch sites for SpaceX's rocket program. There are three on the east coast of the United States, and one on the west coast.



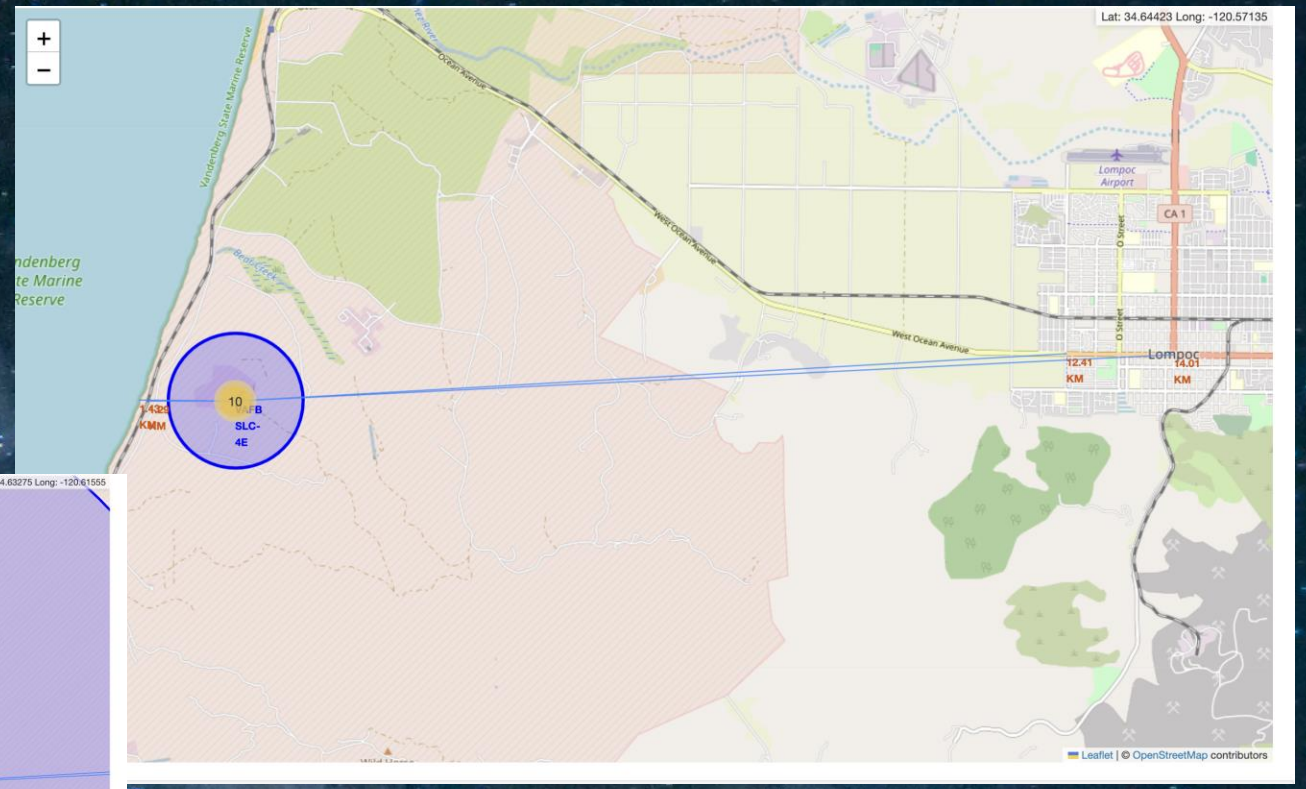
Launch Outcomes at Sites

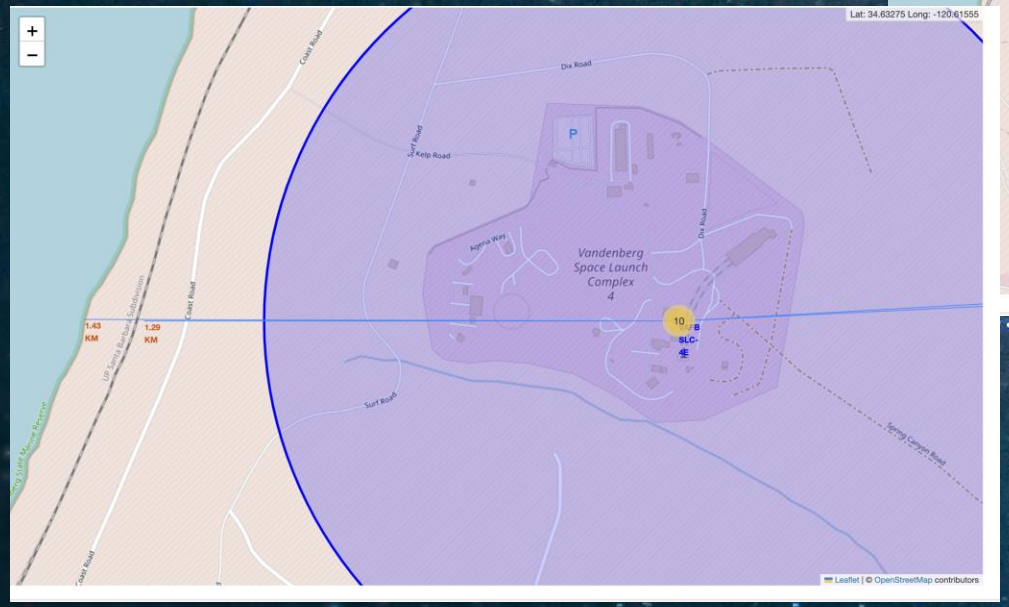
Using marker_cluster, we placed all of the launches at each site. This is indicted by the number inside of the launch site circle marks. When clicked on, all of the single color marked launch data points are shown per site. Green indicates successful missions and red is for all failed.



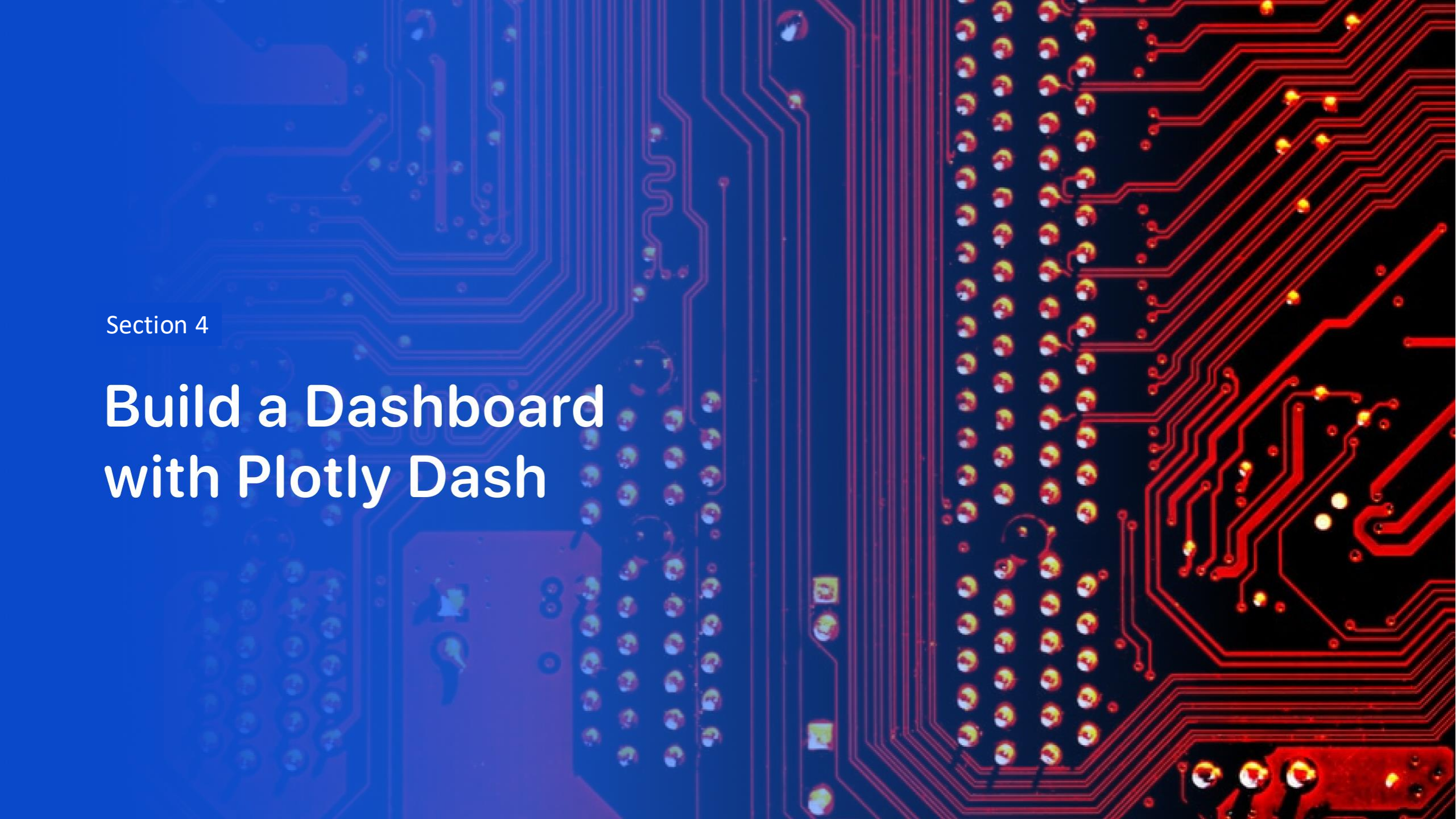
Proximity of Landmarks to Launch Site VAFB SLC-4E

This shows the closest railroad, coastline, highway, and city distances marked out for the VAFB SLC-4E site, the single west coast launch site.

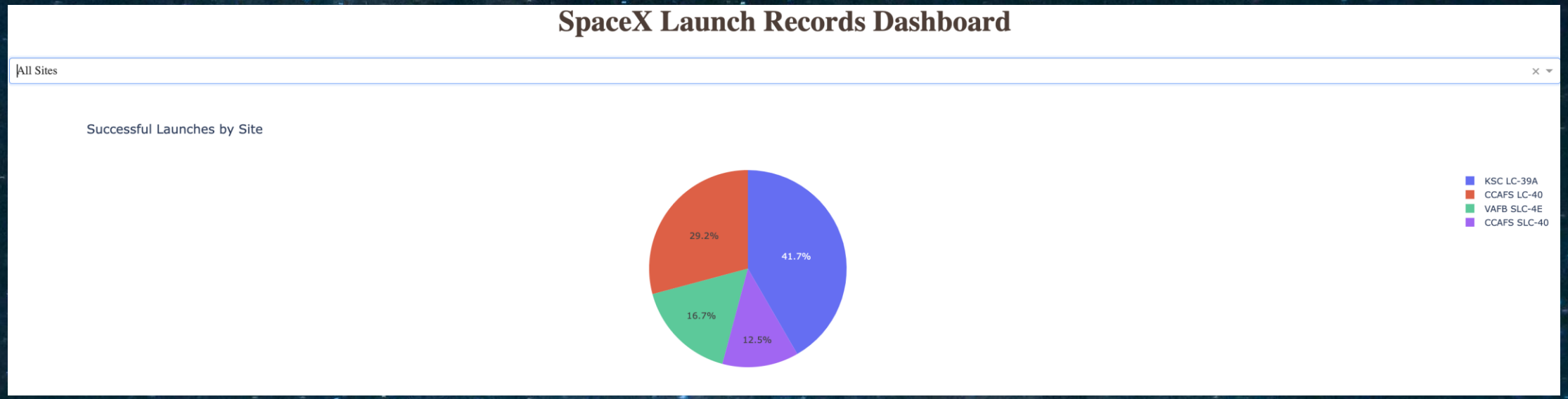




Zoom in on the site to better show the nearest railroad and coast line distances.

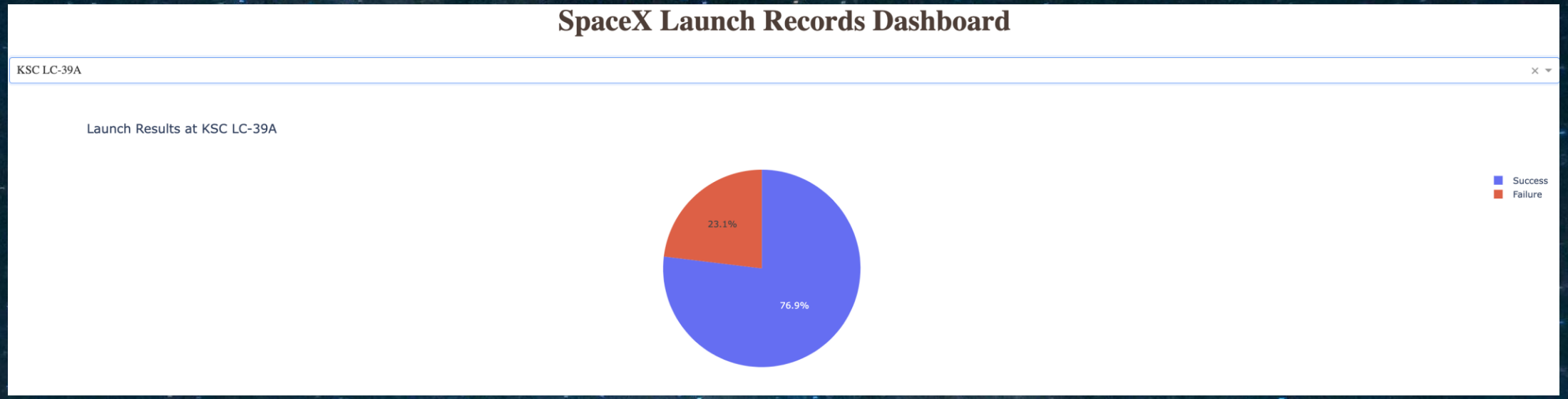

Section 4
Build a Dashboard with Plotly Dash

Successful SpaceX Launch Missions by Site



Here we observe ratios of successful missions all the sites. There is large majority of successes at the KSC LC-39A site.

Site with Highest Launch Success



As indicated by the All Sites chart, KSC LC-39A site also has the highest individual success rate, with only a 23.1% failure rate. This data is definitely skewed in some way, since site CCAFS LC-40 had 26 launches, while this site only had 13. CCAFS LC-40 might have been used more heavily in the early research phase

Success vs Payload Mass and Booster type

We observe below 5000kg that nearly all the failed launch are boosters v1.0 and v1.1

While over 5000kg, we only see only two boosters are used, and their success rates are still very low.

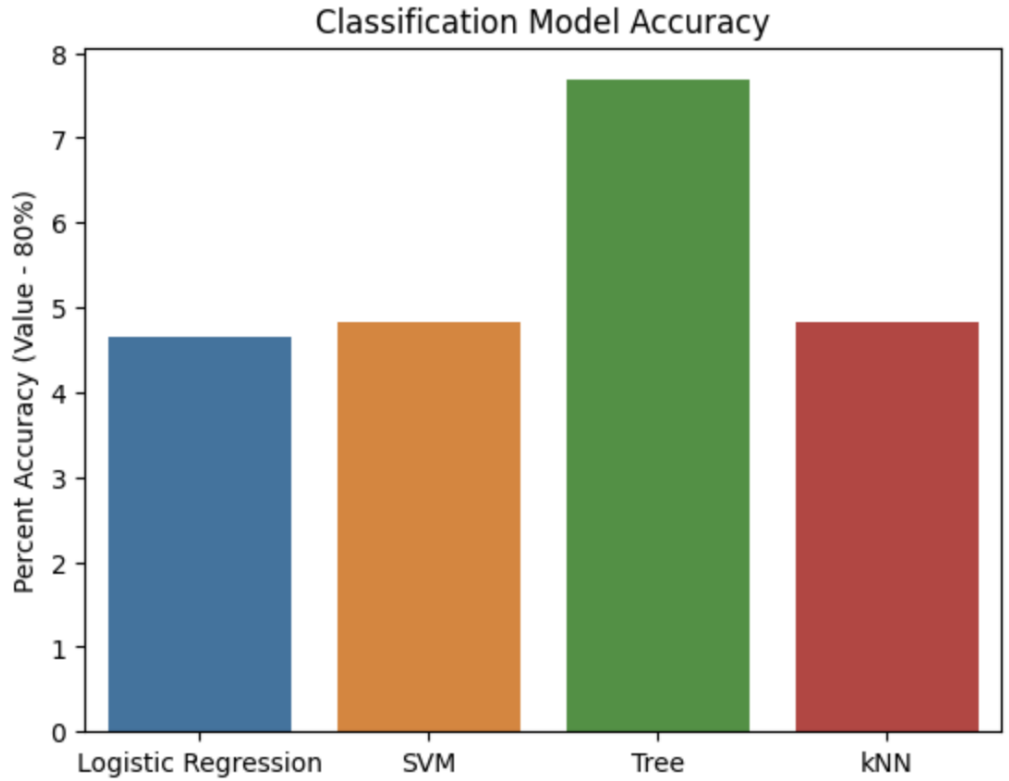
Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
[63]: sns.barplot((bests*100-80))
      plt.ylabel("Percent Accuracy (Value - 80%)")
      plt.title("Classification Model Accuracy")
      plt.show
```

```
[63]: <function matplotlib.pyplot.show(close=None, block=None)>
```

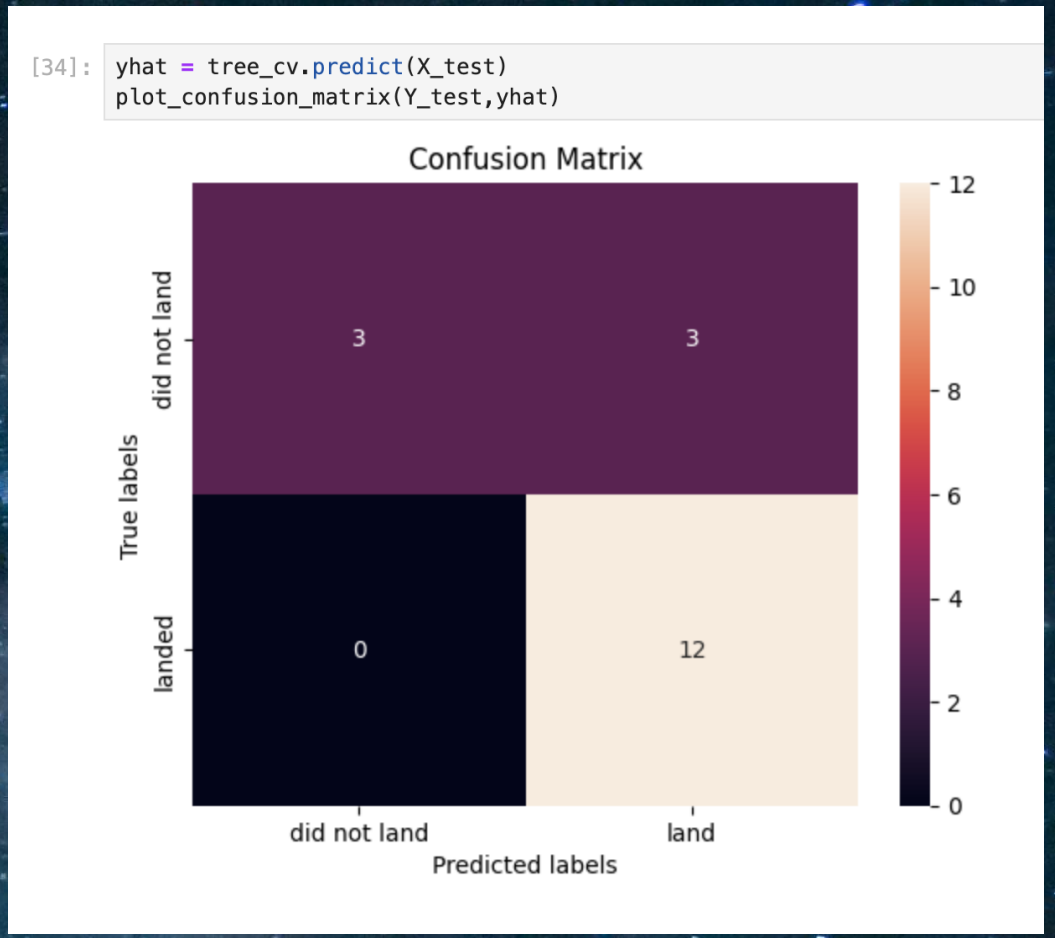


When evaluating classification models, we found that the Decision Tree algorithm had the highest accuracy at approximately 87.7%. All of the different algorithms had similar accuracies, so I subtracted 80 from the values to give more visibility.

Confusion Matrix

The Decision Tree algorithm does a good job in classifying the test set. From the confusion matrix we can see that it only misinterpreted three data points, putting out three false-positive predictions.

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



Conclusions

1. The VLEO orbit missions have high success rates.
2. Progressive research and time have seen higher successful returns with all missions.
3. All launch sites are on the southern edge of the country and close to a major city, supply lines (highways and train tracks), and the coasts.
4. The KSC LC-39A launch site has a very high success rate, though no v1.0 or v1.1 were launch for there.
5. Heavy payloads missions still struggle to land successfully.
6. Breaking down the differences between the v1.0 and v1.1 rockets and FT, B4, and B5 rockets will give us a better understanding of how to get a head start on having successful missions. There is a distinct increase in success with the latter set of rockets.
7. The best classification model for predicting launch outcomes is the Decision Tree, at 87.7% accuracy.

Appendix

- Repository of all files: https://github.com/jffessler/IBM_Data_Sci_Cert_Final/tree/main
- Final barplot for classification accuracy:
  - bests = pd.DataFrame(best_scores, index =[0])
  - import seaborn as sns
  - sns.barplot((bests*100-80))
  - plt.ylabel("Percent Accuracy (Value - 80%)")
  - plt.title("Classification Model Accuracy")
  - plt.show

Thank you!