IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jeffrey Andes
Dec 22, 2023

Outline

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

Executive Summary

- Methodology:
  - Extracted the data using REST API and Web scraping
  - Wrangled the data to fill in missing pieces and create numerical values for categorical data
  - Performed Visual Analysis on the data to see if there was any insight or correlations that can be acquired.
  - Created interactive maps and a dashboard to help drill down further.
  - Performed a CV Grid Search with several different classification types to arrive at the best model for predictive success and failure of a mission.
- Results:
  - The visual map is telling to highlight location for where launches occur from
  - The Payload Mass influences the overall success rate of the mission
  - A predictive model can be created with a strong accuracy for data available.

Introduction

- Background
  - Commercial space flight has become very competitive recently with several companies offering a service.
  - The most successful has been SpaceX with their Falcon 9 rocket.
  - A new company is looking to join this industry, following the SpaceX model will they be successful?
- Questions to answer:
  - How much does it cost SpaceX to launch each rocket?
  - Will the booster be recovered?
  - Will the launch be a success?

Section 1

Methodology

Methodology

Executive Summary

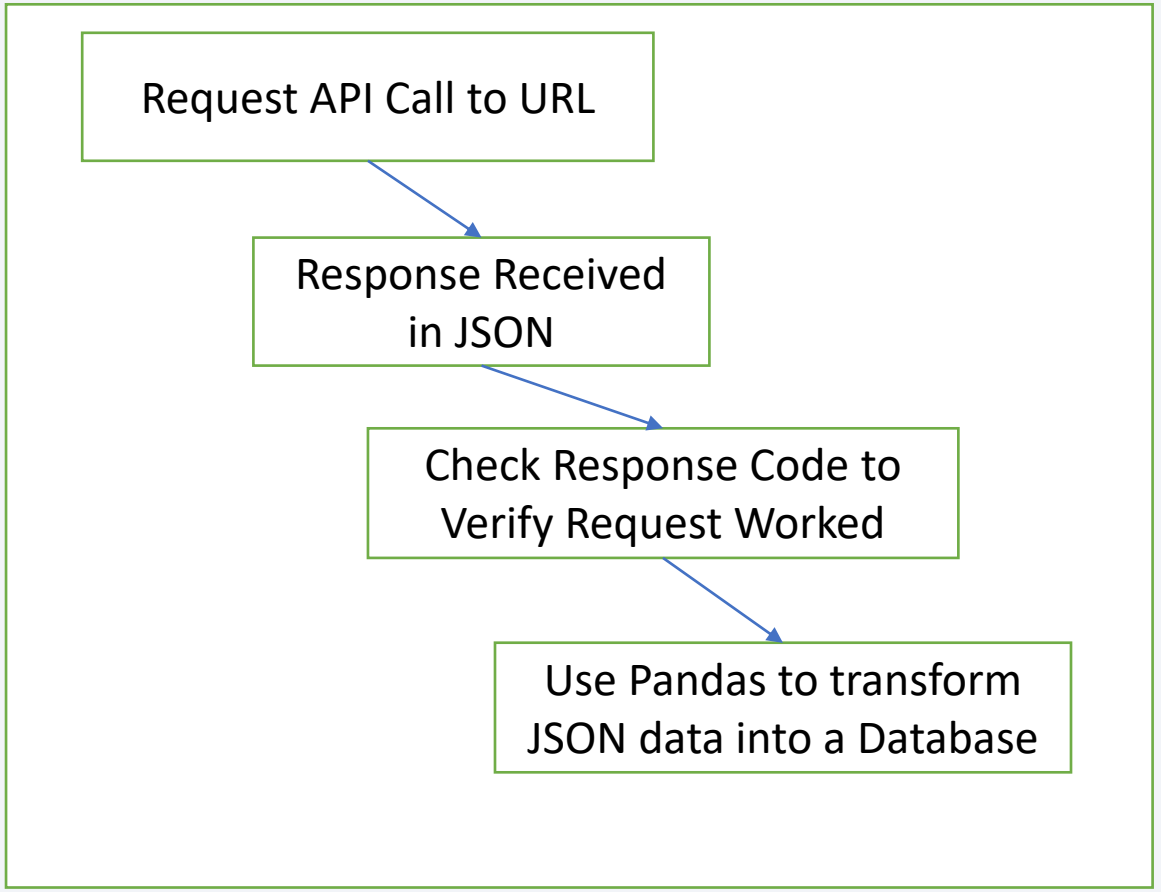
- Data collection methodology:
  - Used SpaceX API to collect data
  - Use Web scraping on Wikipedia to collect data
- Perform data wrangling
  - Using Pandas, we verified the relevant data was present and created our target in pass or fail.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Set-up a Grid Search CV and performed on multiple model types to determine the best.

Data Collection

- SpaceX API
  - Used the requests library to access the SpaceX API and pull relevant data that we turned into our initial data frame.
- Wikipedia Web Scraping
  - Used the Falcon 9 and Falcon Heavy Wikipedia page to pull launch data from the table along with the outcome.

Data Collection – SpaceX API

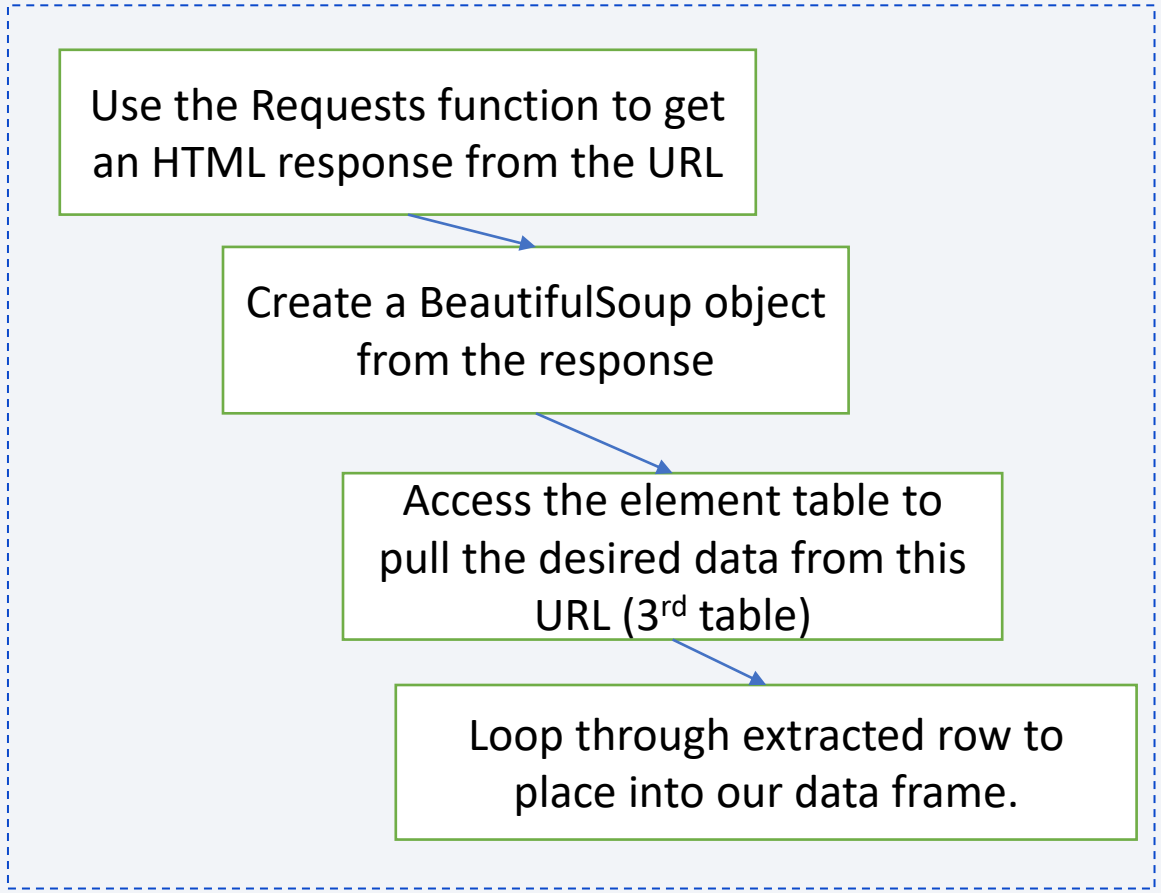
- Used the Requests library to access the REST API and receive a response back in JSON format.
- Then transformed the JSON format into a data frame that can be worked through Pandas
- API Notebook



Data Collection - Scraping

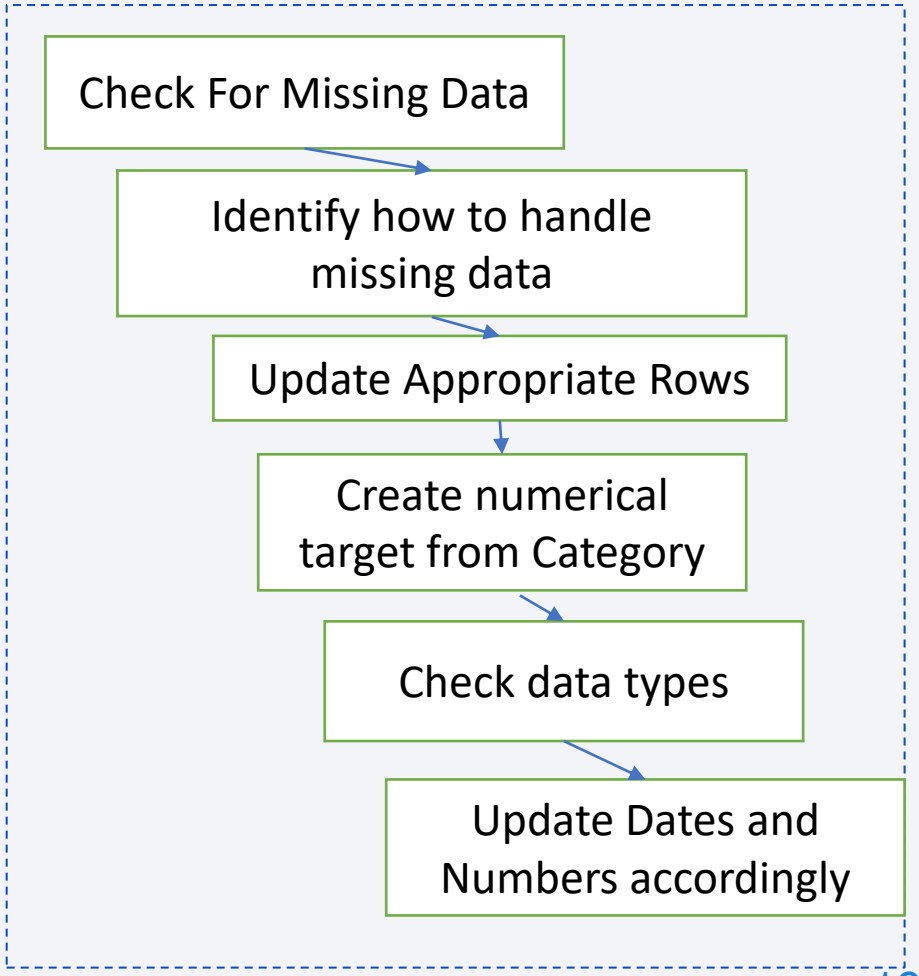
- Used BeautifulSoup and requests libraries to access the URL and pull the desired data. Once the HTML data was retrieved, used the elements to access the pieces desired to build my data frame from.

- Web Scraping Notebook



Data Wrangling

- Using Pandas we checked for missing data and based on the % missing either removed the lines or updated with a best approximation using either the mean of the present data or the most frequent depending on what was appropriate for the data that was missing.
- From the landing outcomes created a success for failure target represented as 1 or 0.
- Check data types and adjust numerical and data columns accordingly.
- API Notebook with Data Wrangling at the End
- Data Wrangling Notebook



EDA with Data Visualization

- Scatter Plots
  - Payload Mass vs Flight Number – See if the weight contributed to the success for failure
  - Launch Site vs Flight Number – Are launches more successful at one site then another?
  - Payload Mass vs Launch Site – Does one site handle heavy or light loads better then others?
  - Orbit vs Flight Number – Does the orbit effect the success rate, has one orbit become more consistant?
  - Orbit vs Payload Mass – Is the a correlation between orbit and payload in relation to success?
- Bar Chart
  - Orbits vs Outcome – Which orbits have better success rates?
- Line Graph
  - Date (Year) vs Outcome – Has the success rate improved over the years at SpaceX?
- EDA Data Visualization Notebook

EDA with SQL

- Identify the Unique Launch Sites
- Review a few CCA Launch Sites
- Calculate the total payload mass carried for NASA (CRS) as the customer
- Calculate the average payload mass carried by the Falcon 9 v1.1 booster
- Date of the first successful ground pad landing
- Boosters with successful drone ship landing carrying a payload in a given range
- Listing of the count of successful and failed missions
- List of the boosters that carried the max payload
- List of the month, booster and launch site that had a failed drone ship landing in 2015
- Rank of the landing outcomes in a given date range in descending order
- EDA SQL Notebook

Build an Interactive Map with Folium

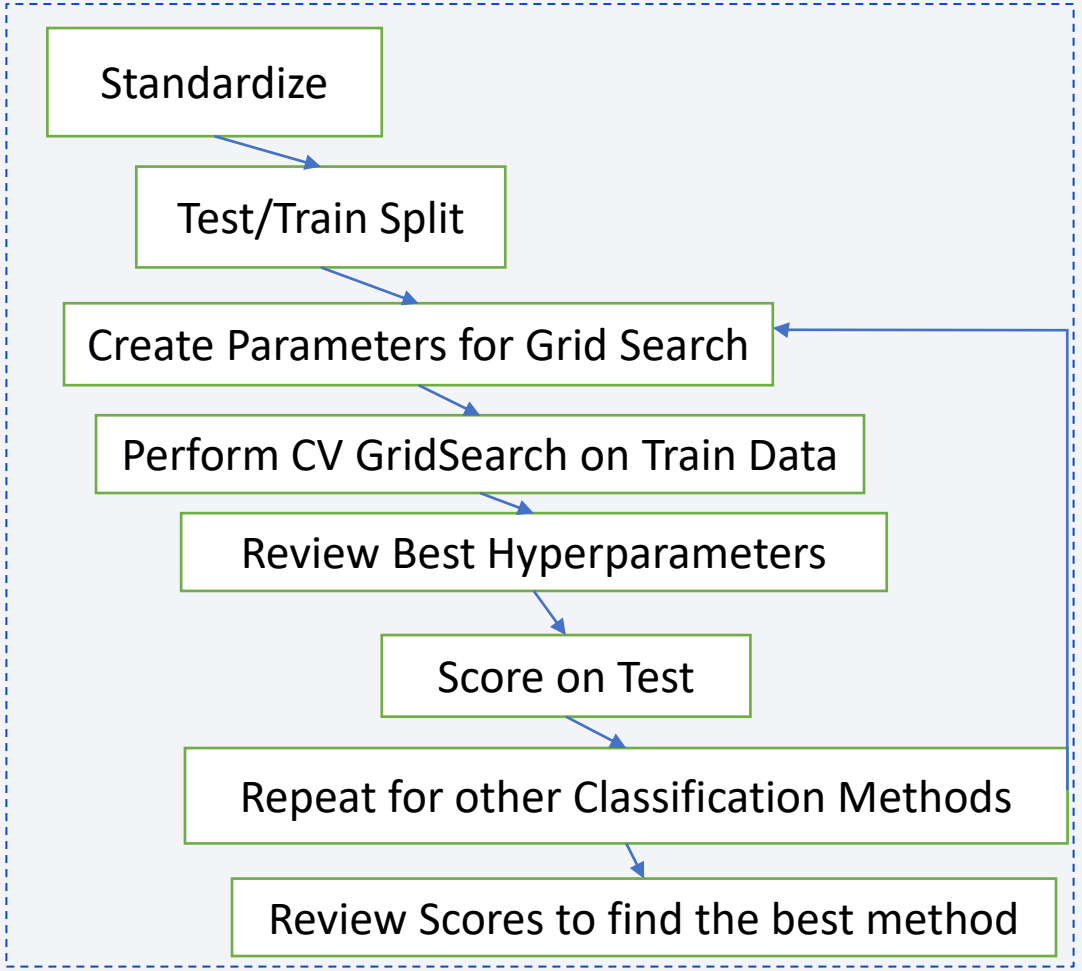
- On an interactive map there are the following features:
  - Circled – Launch Sites
  - Marked – Each successful and failed launch
  - Line and Distance – To The nearest coastline, highway, railroad and city.
- On the map these features give a visual representation of where rockets are launch from and how successful they are. Along with an idea of the infostructure in highways and railroads that might be needed to access the launch site. Additionally, the proximity or distance from a city and coastline appears to be important when planning where you would build your own launch site.
- Folium Map Notebook

Build a Dashboard with Plotly Dash

- Plots/Graphs
  - Pie plot to show the breakdown of successful launches between the sites
  - Pie plot to show the breakdown of successful vs failed launches at 1 chosen site
  - Scatter plot between the payload mass and outcome class while highlighting the different boosters to show at what mass the success rate increases or decreases for and which boosters have worked the best.
- Interactions
  - A dropdown menu helps provide the ability to look at all launch sites or focus on just one.
  - A Slider for the payload mass that can be toggled to select the payload range to look at.
- Plotly Dash Python code

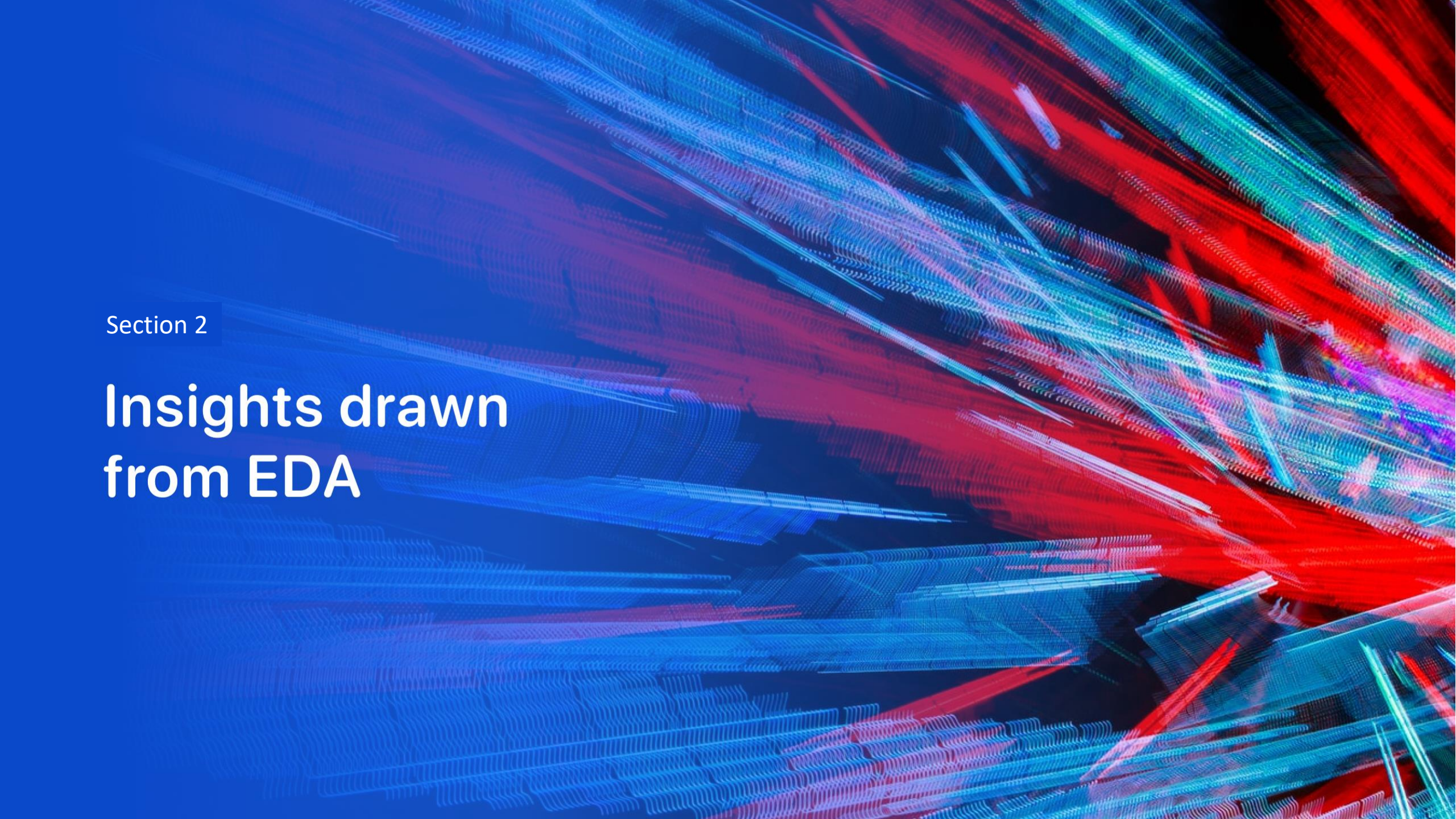
Predictive Analysis (Classification)

- Following the method of standardizing the data to make sure no 1 parameter has a larger influence on the prediction then another we split the data into test and train.
- Then to help with under and over fitting we did a CV training of the model with our GridSearch to get our hyperparameters.
- We repeated this to collect the score for several different classification methods in order to arrive at the best for this task.
- ML Predictive Analysis Notebook

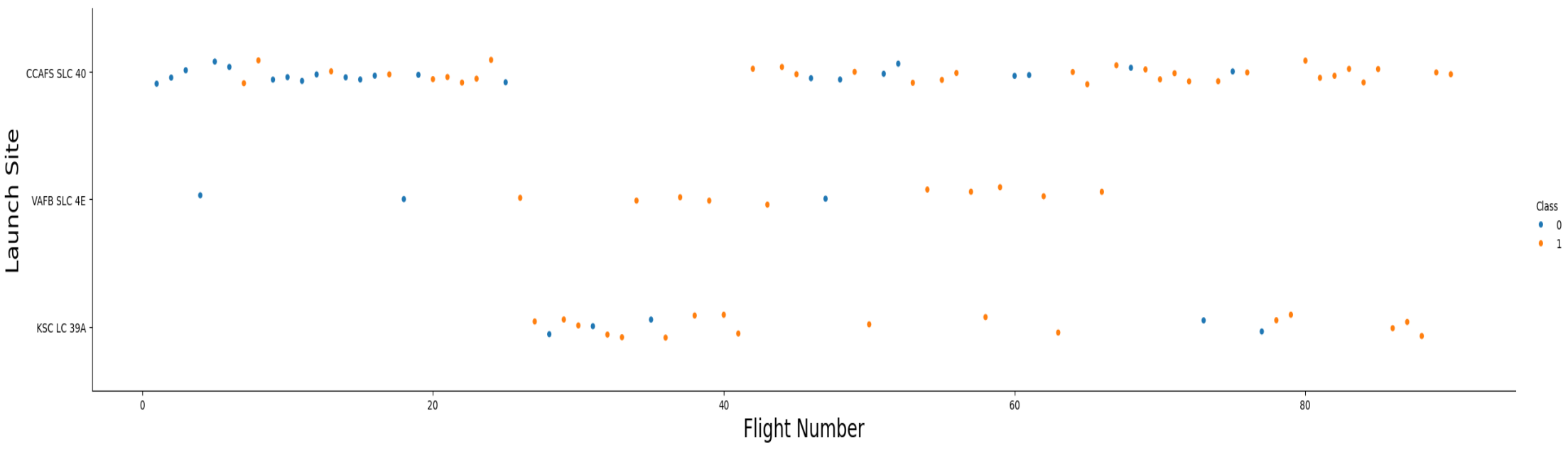


Results

- Used SQL and Seaborn with Pandas to do initial Exploratory data analysis.
  - Reviewed data through Scatter Plots, Bar Charts and Line Graphs.
  - In SQL we looked at Unique Launch Sites, Booster Success and Failures, different time periods, average payload as well at total payload.
- Through Folium and Plotly we can have some interactive exploration of that data as capture in screen shots for this presentation.
  - With Folium we created visual maps to see where the launch sites are located and outcome of launch at each site.
  - We also looked at proximity to key infostructure and cities.
  - With Plotly we created a dashboard that is captured in screen shots to see success vs failure along with how the payload effect it for the different launch sites.
- Performed Predictive Analysis
  - Used CV Grid Search with several different classification approaches.
  - After we were able to compare the different approaches to see which one was most accurate.

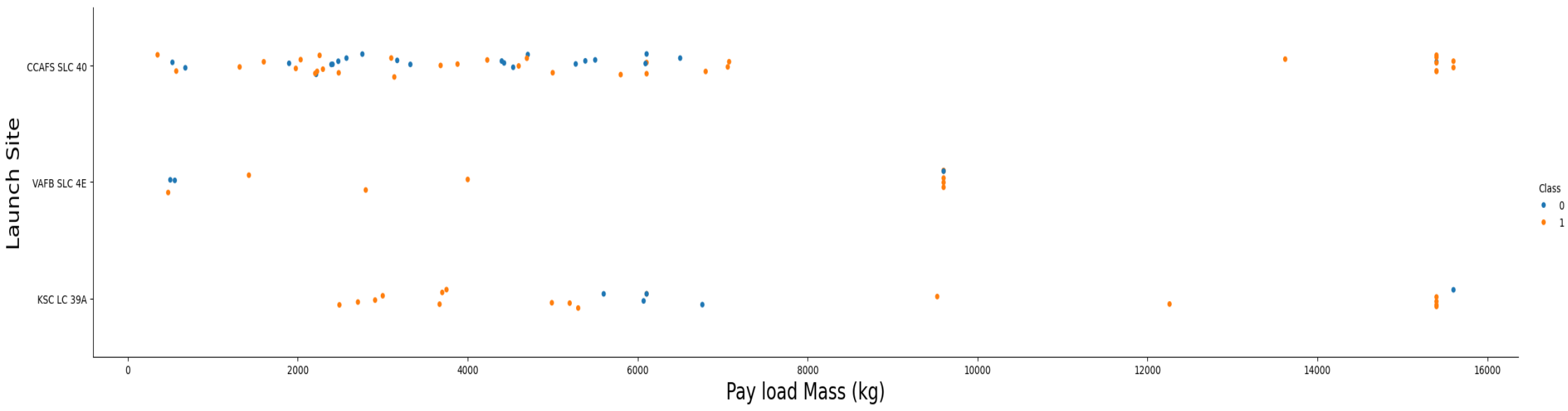

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



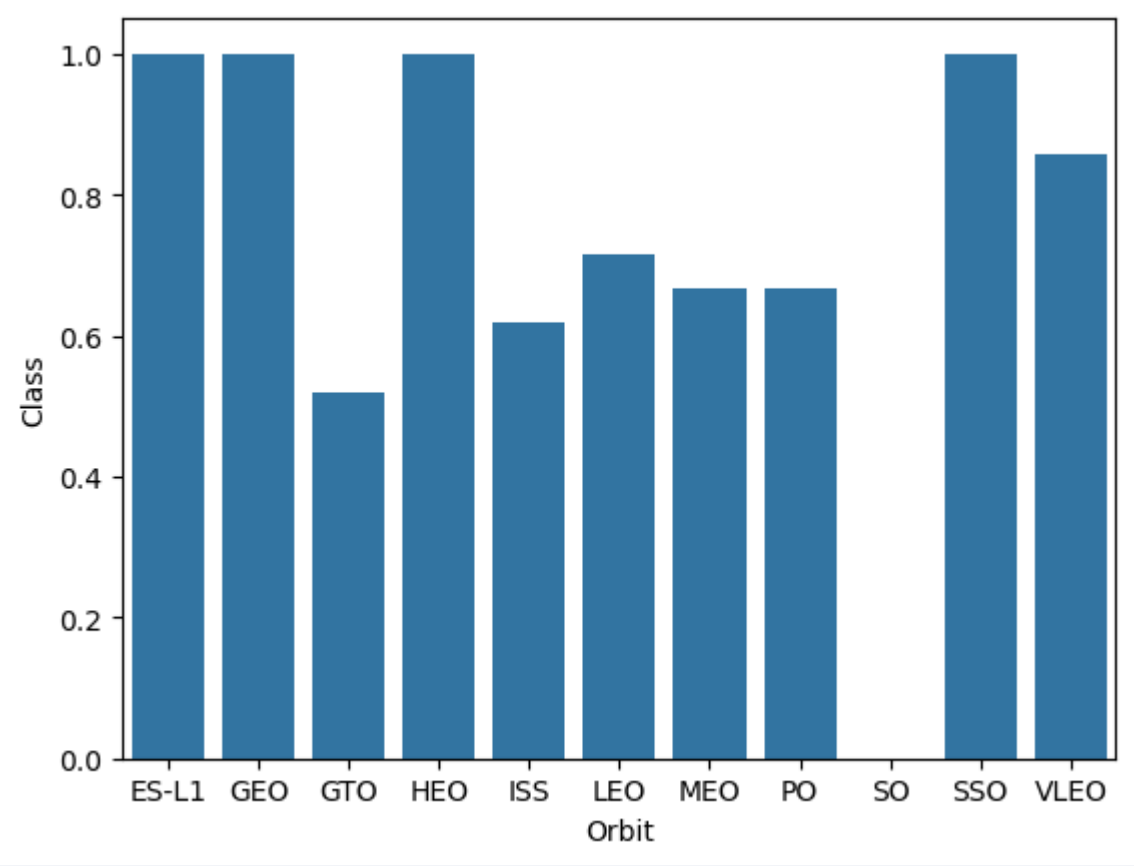
- We can see that different launch sites have higher success rates even in the higher flight numbers, but all launch sites appear to get have more success in general at the high flight numbers.

Payload vs. Launch Site



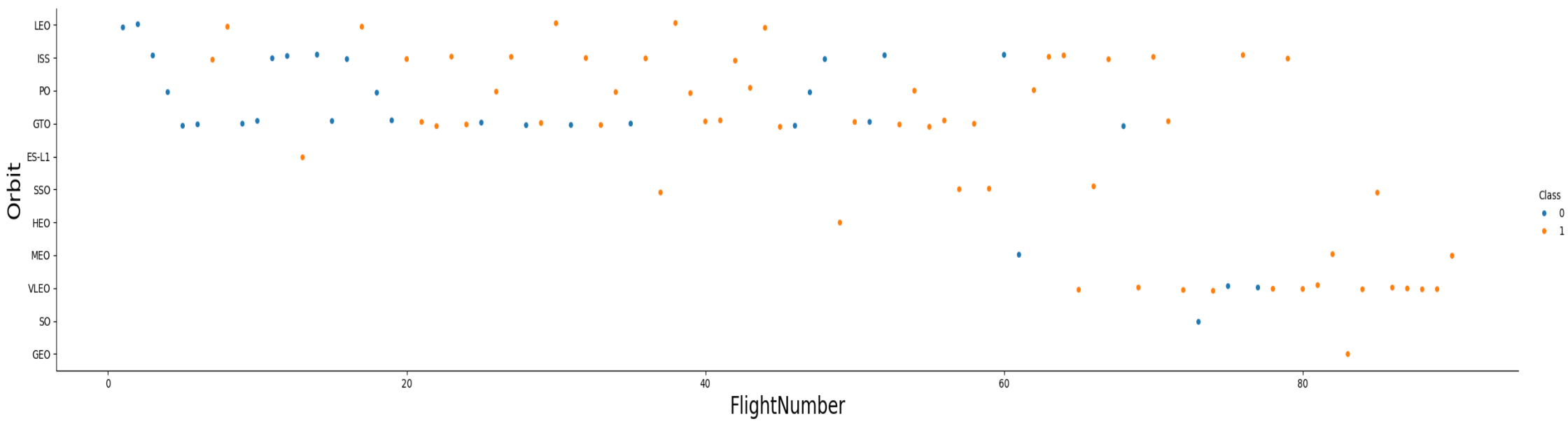
- Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000).

Success Rate vs. Orbit Type



- There are several orbits (ES-L1, GE, HEO and SSO) with 100% success rate as shown on the bar chart with VLEO also having a high success rate over 80%.
- The orbits of GTO and SO do not have a good success rate.

Flight Number vs. Orbit Type

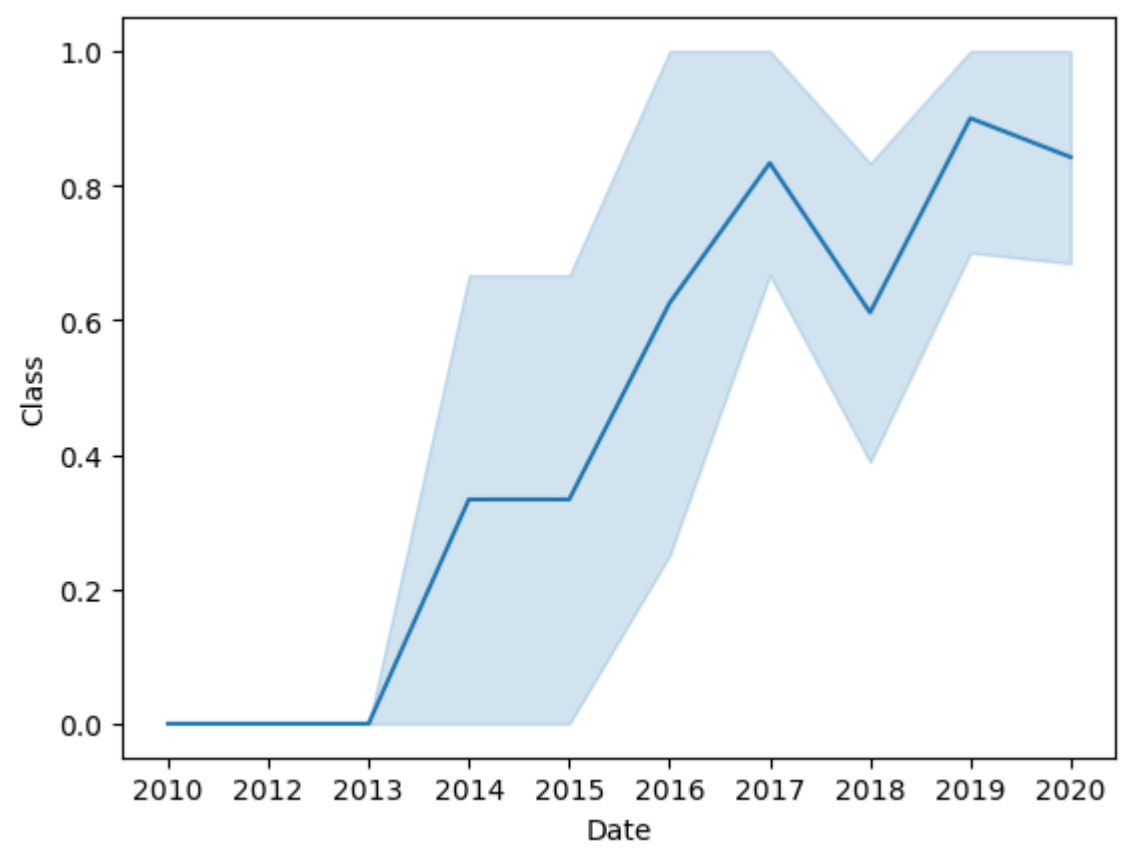


- You should see that in the LEO orbit the success appears related to the number of flights.
- On the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend



- You can observe that the overall success rate since 2013 kept increasing till 2018
- After 2018 the success increased again in 2019 but decreased in 2020 again.
- However, the overall success appears on an upward trend from where it started in 2013.

All Launch Site Names

```
SELECT Launch_Site
FROM SPACEXTABLE
GROUP BY Launch_Site
```

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

- After we pulled the data, we wanted to see how many unique launch sites rockets launched from.
- Using SQL, we were able to do a quick query to see that there are 4 different launch sites used.
- In SQL we were able to get this list by grouping the data by Launch Site.

Launch Site Names Begin with 'CCA'

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

```
SELECT *
FROM SPACEXTABLE
WHERE Launch_Site Like "CCA%"
LIMIT 5
```

- With 2 sites having CCA in the name we wanted to take a closer look at a few of the records.
- Using the “Where” clause in our SQL query along with limiting our results to 5 we were able to get a snapshot of what type of launches originated from these 2 sites.

Total Payload Mass

```
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Customer == "NASA (CRS)"
GROUP BY Customer
```

SUM(PAYLOAD_MASS__KG_)
45596

- We want to get an idea of how much Payload has been launched for NASA as the customer.
- In SQL this is a quick sum while limiting the records we look at to just NASA as the customer.
- The group by allows SQL to add up the desired records we want to know the totals for, in this case the customer that is NASA.

Average Payload Mass by F9 v1.1

```
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Booster_Version Like "F9 v1.1%"
```

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

- We want to get a quick idea of what the average payload has been using a specific rocket booster.
- Using the AVG function in SQL and limiting the booster version gives us the quick reference we are looking for.

First Successful Ground Landing Date

```
SELECT MIN(Date)
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE "Success%"
```

MIN(Date)
2015-12-22

- How long has Space X been able to pull off a successful ground landing?
- How long after the first rocket launch were they able to successfully land?
- In order to start answering these questions we need to know when the first successful landing occurred.
- Using the MIN function on the date and limiting out results to only successful outcomes we can find this date.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
SELECT Booster_Version
FROM SPACEXTABLE
WHERE Landing_Outcome == "Success (drone ship)" AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

- We seen in the scatter plots that a payload between 4000-6000 has a lot of data point.
- So which boosters were used within this range and were successful
- Using the WHERE clause in SQL we can limit our results to see only the boosters with a successful outcome in that payload range.

Total Number of Successful and Failure Mission Outcomes

```
SELECT Landing_Outcome, COUNT(*)
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE ("Failure%") OR Landing_Outcome Like ("Success%")
GROUP BY Landing_Outcome
```

Landing_Outcome	COUNT(*)
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
Success	38
Success (drone ship)	14
Success (ground pad)	9

- What type of outcomes were achieved and how many of each?
- Using a count and grouping by the Landing Outcome we can get these numbers.
- We also want to make sure we only look at successes and failures, as it appears there were times when the outcome didn't matter.

Boosters Carried Maximum Payload

```
SELECT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ == (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
```

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

- We wanted to see which boosters carried were used with the max payload.
- First, we needed to know the max payload that was carried, then use that number within our query to limit our results to only those records that carried the max payload.

2015 Launch Records

```
SELECT substr(Date,6,2) AS Month,
        Landing_Outcome,
        Booster_Version,
        Launch_Site
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE ("Failure (drone ship)") AND substr(Date,0,5)='2015'
```

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

- We know that in 2015 there was a successful ground landing, but were they also testing drone ship landing and how many times did they fail if they were?
- We can see they made 2 failed attempts at the beginning of the year (January and February).
- We were able to look at this data by limiting our results based on outcome and the year portion of the date.

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

```
SELECT Landing_Outcome,
        COUNT(Date) AS Outcome_Count,
        Rank() OVER(ORDER BY COUNT(Date) DESC) AS Rank_Outcome
FROM SPACEXTABLE
WHERE Date BETWEEN date('2010-06-04') AND date('2017-03-20')
GROUP BY Landing_Outcome
```

- Between June 2010 and March 2017, we wanted to see how the rocket testing and missions went at Space X.
- We wanted to see the results ranked in most frequent outcome to least frequent within that date range.

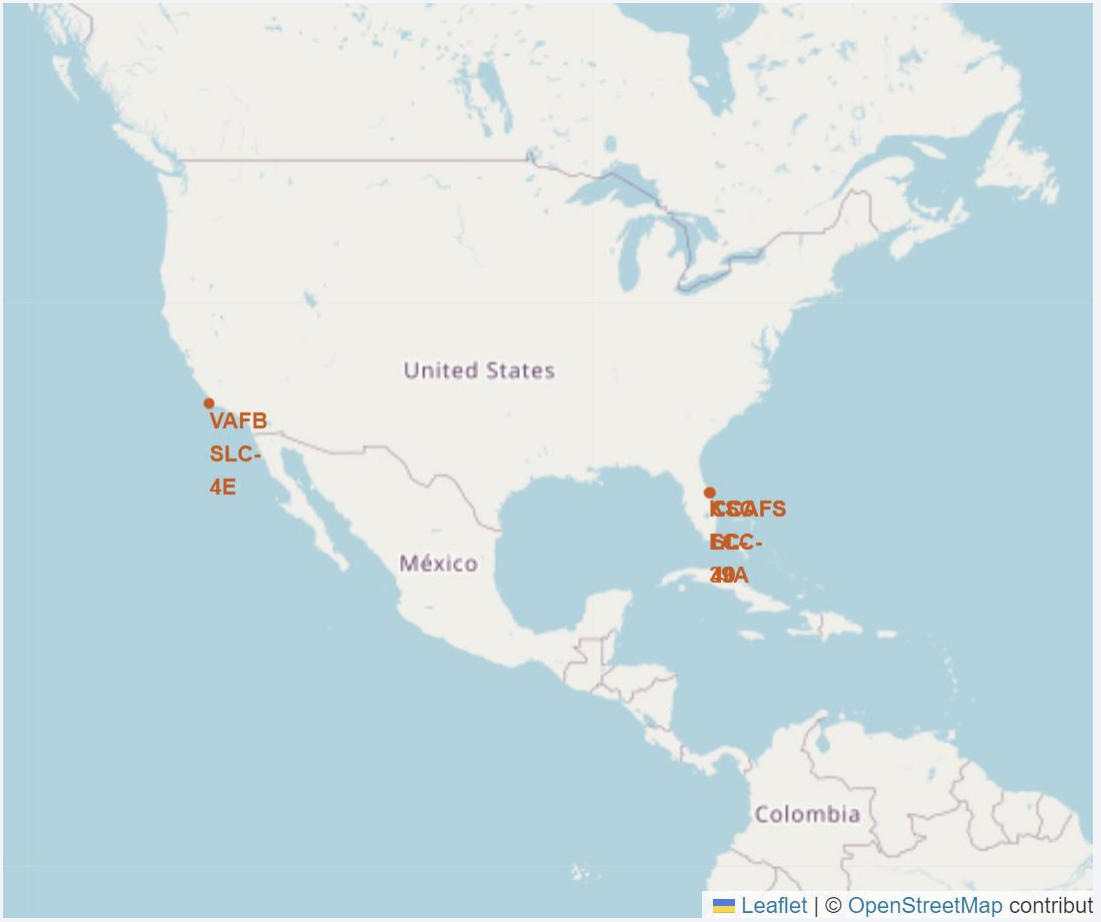
Landing_Outcome	Outcome_Count	Rank_Outcome
No attempt	10	1
Success (drone ship)	5	2
Failure (drone ship)	5	2
Success (ground pad)	3	4
Controlled (ocean)	3	4
Uncontrolled (ocean)	2	6
Failure (parachute)	2	6
Precluded (drone ship)	1	8

Section 3

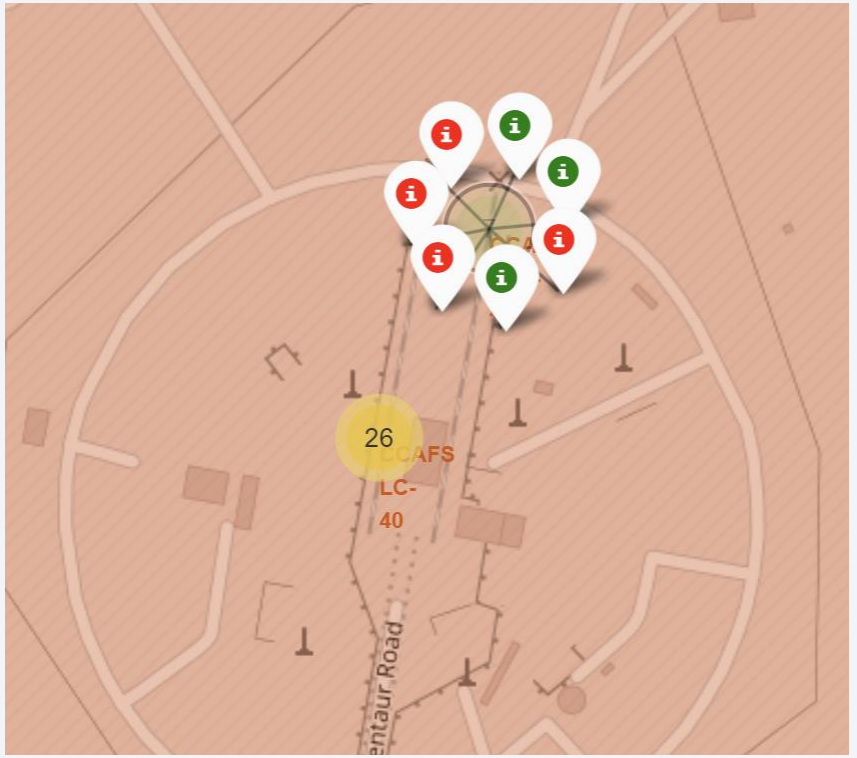
Launch Sites Proximities Analysis

Launch Sites Map

- The 4 launch sites are all located in the United States with 3 of them being very close together on the southeastern coast.
- Their locations are marked and circled on the map along with the site names marked for easy identification.
- Ocean coastline appears to be import to their placement.

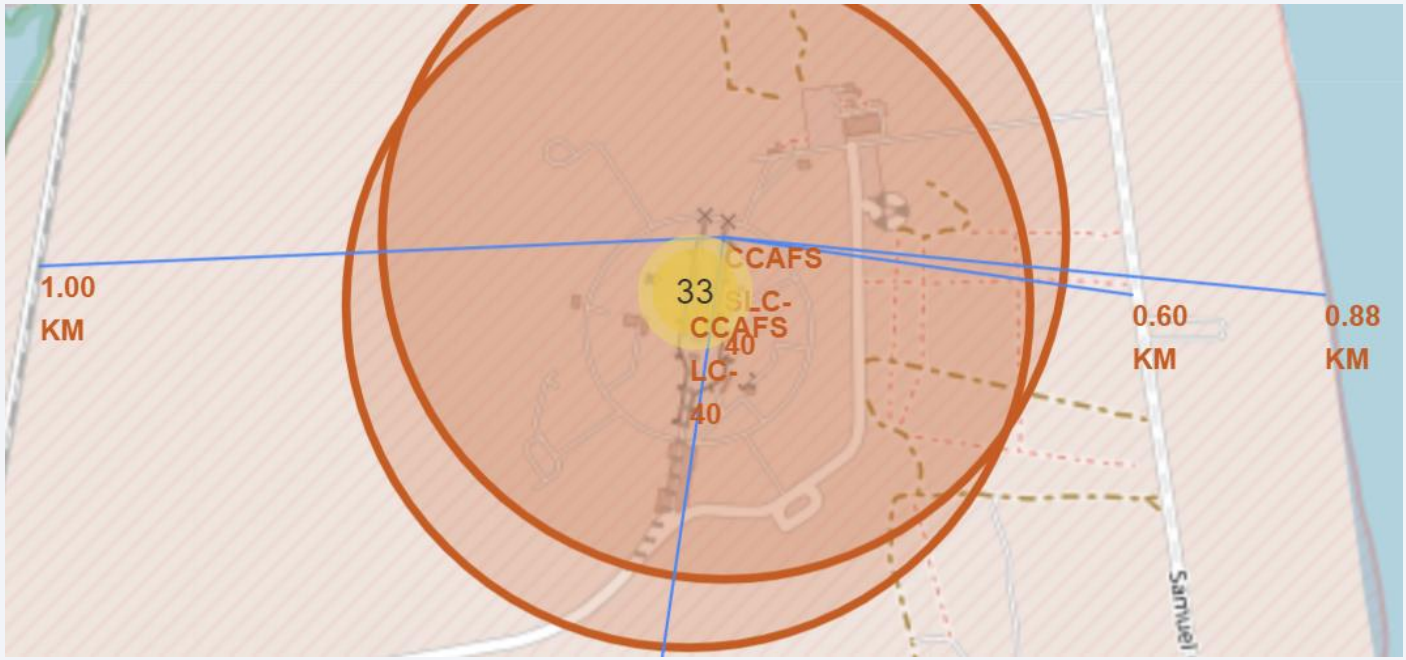


Global Map with Successes and Failures Marked

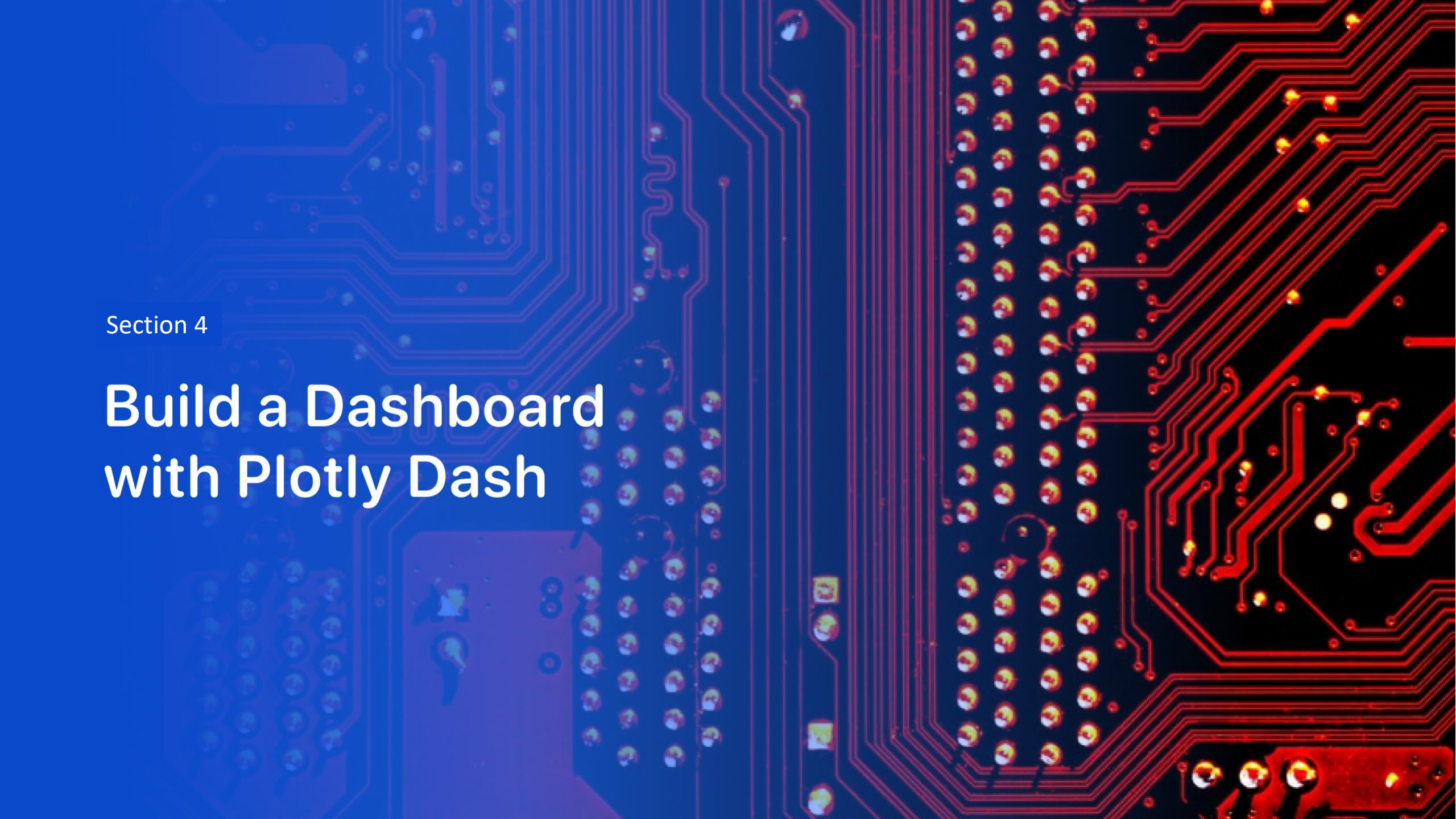


- With the launches marked on the map we can see where the majority have launched from.
- When zoomed in we can see the successes (green) and failures (red) also marked on the map.

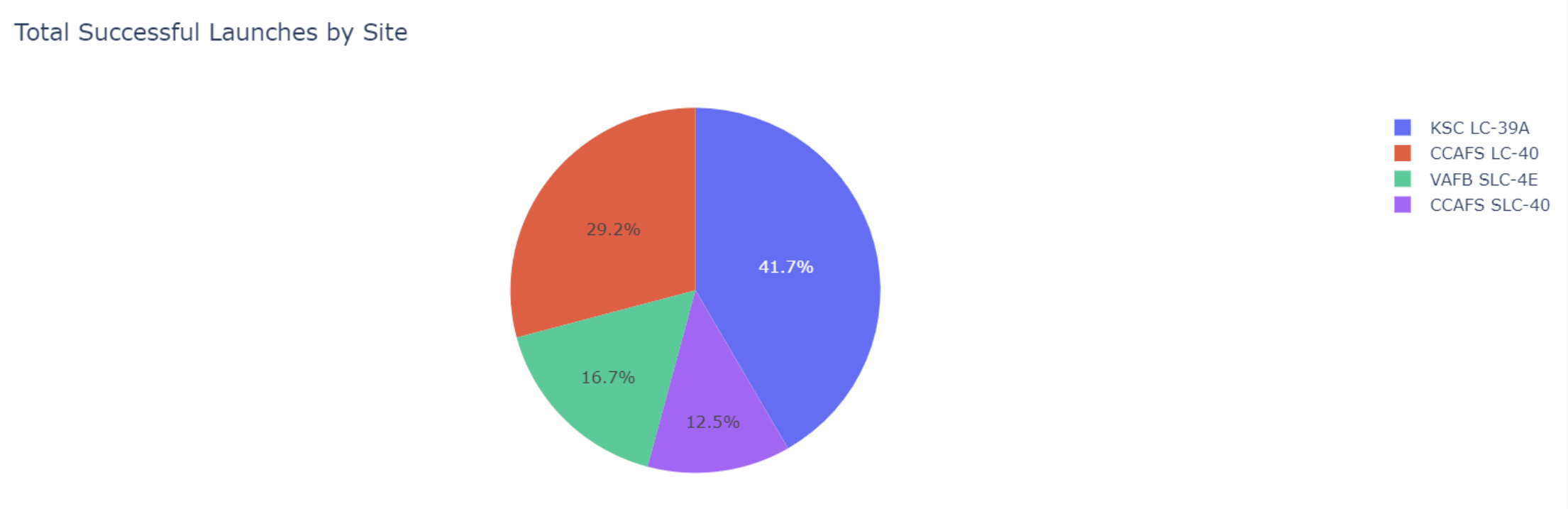
Proximities Map



- Looking at the distance from launch site to coastline, highway, railroad and city we see that usually launch sites are close to the coast, highway and railroad, but located a distance away from any city. This is import for planning any future launch sites.

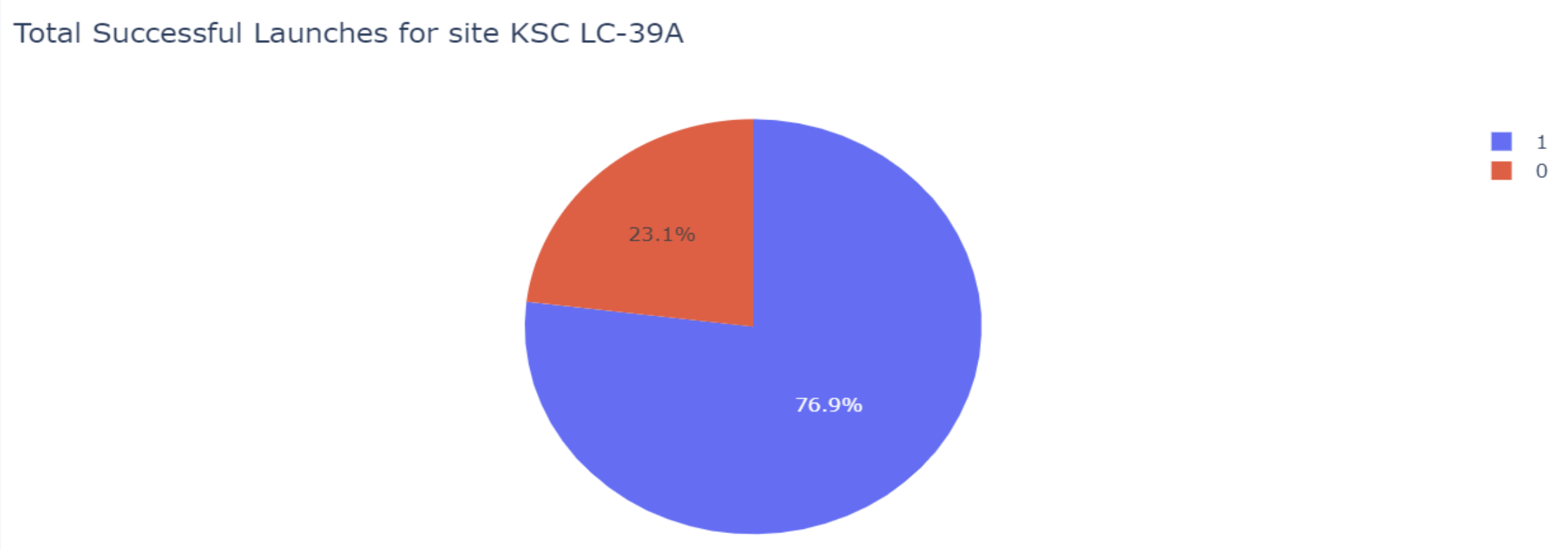

Section 4
Build a Dashboard with Plotly Dash

Successful Launches by Launch Site



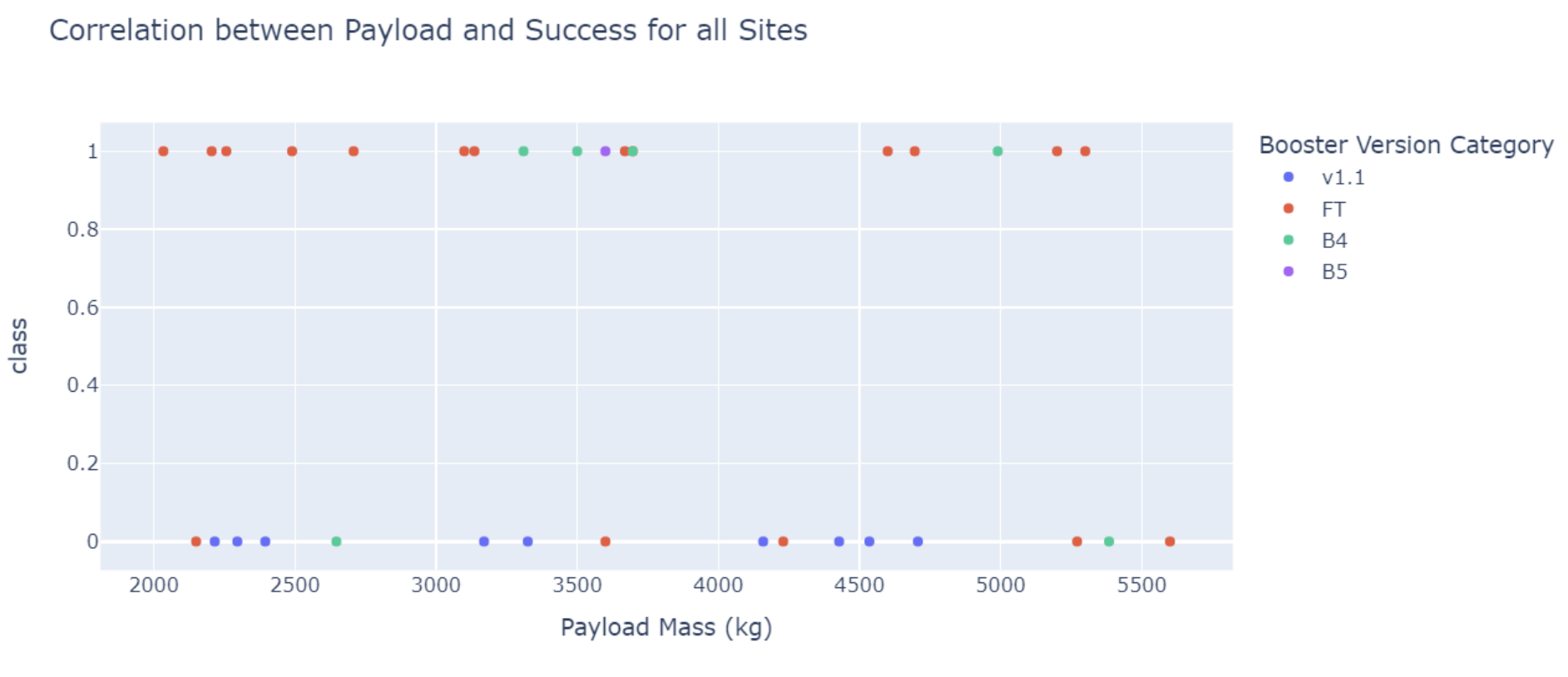
- Through the pie chart we can see the breakdown of which sites more successful launches have come from, with KSC LC-39A account for over 40% of the successful launches.

KSC LC-39A Successful Launch Percentage



- In this pie chart we can see that launches form KSC LC -39A are successful on over 75% of their launches. We can see this in the blue section that is the portion that has a successful outcome.

Correlation between Payload and Success



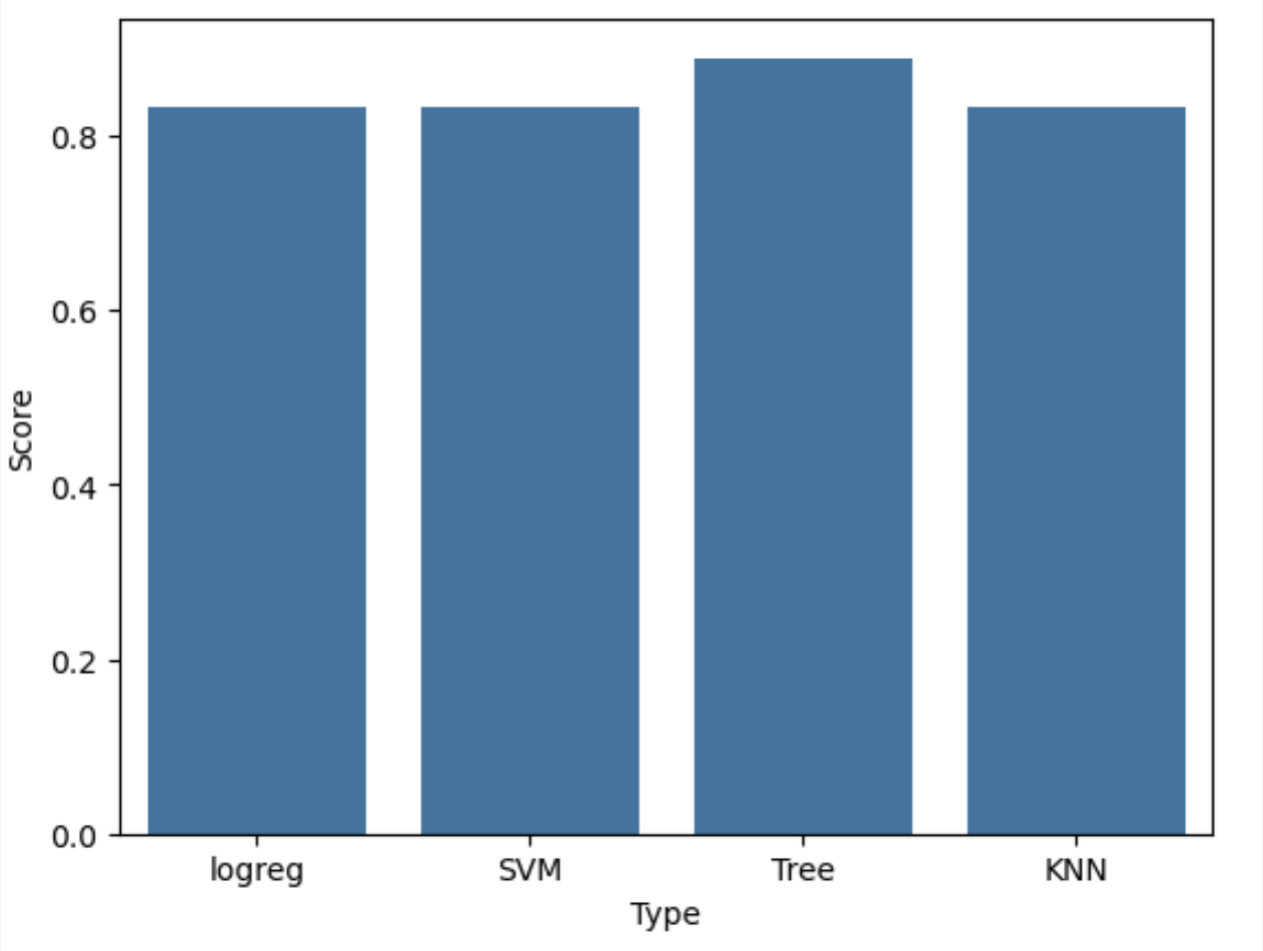
- Most of the launches were between 2000 and 6000kg so highlighting that here. You can also see in this range that both the FT and B4 boosters appear to have a high success rate with payloads under 5000kg being very successful.

Section 5

Predictive Analysis (Classification)

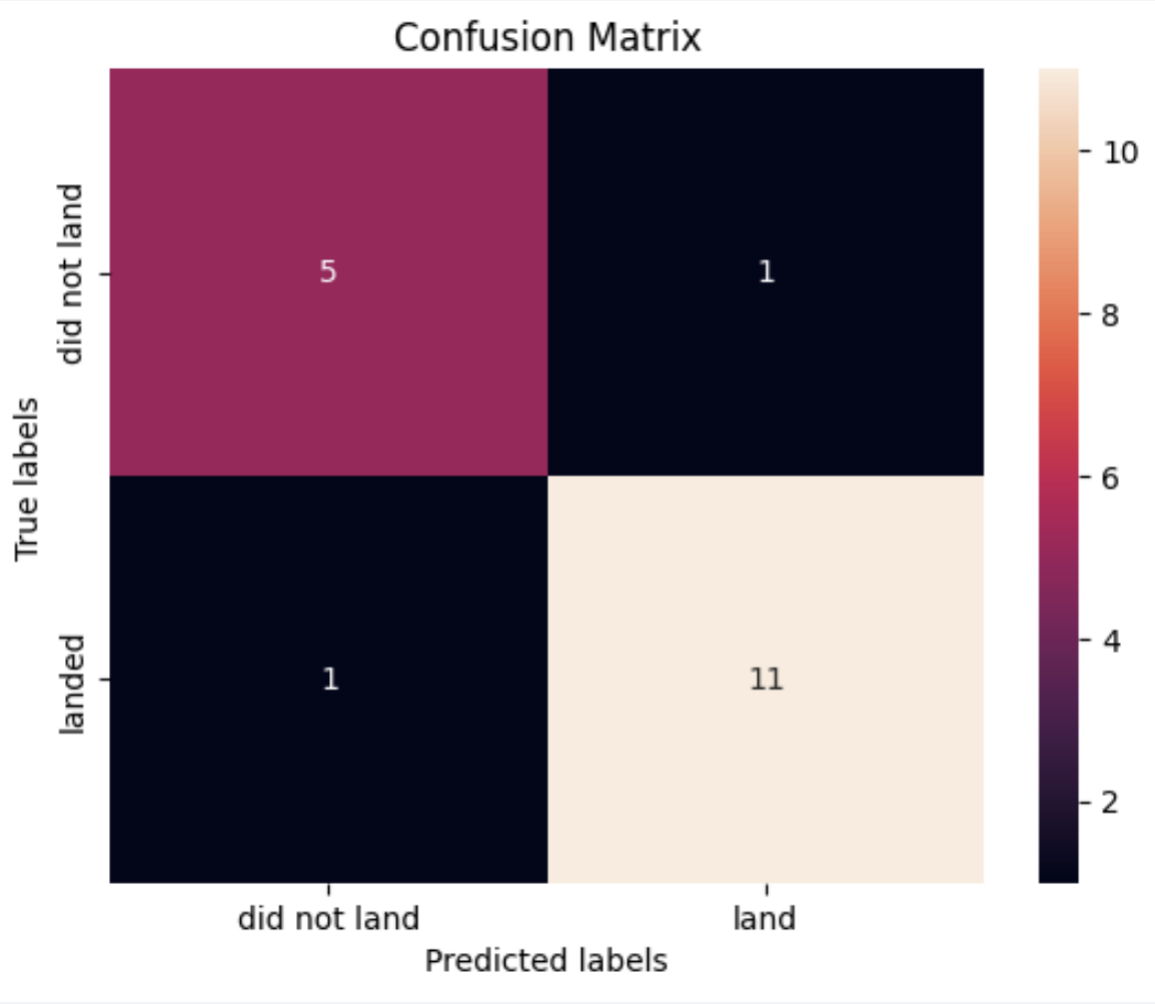
Classification Accuracy

- The Decision Tree model has the highest Accuracy Score.
- This will be the model we chose to use.



Confusion Matrix

- We can see that overall, the model predicted a success and failure correctly 16 times out of the 18.
- This shows good accuracy for the Decision Tree Model.



Conclusions

- Location of Launch Site is important.
- There is a range of payload mass where we can aim for to get the best results.
- For our predictive model we want to use the decision tree method.
- Initial failure as our own rocket is built should be calculated into the start-up for the company.
- Long term, following a similar model as SpaceX another company can be competitive as right now there is minimal competition in this market space.

Appendix

- All python code can be found on Github
- Plotly Dashboard Python Code Snippet:

```
@app.callback(Output(component_id='success-payload-scatter-chart', component_property='figure'),
              [Input(component_id='site-dropdown', component_property='value'),
               Input(component_id='payload-slider',component_property='value')])
def get_scatter_chart(entered_site,entered_payload):
    filtered_df = spacex_df[spacex_df['Payload Mass (kg)'] >= entered_payload[0]]
    filtered_df = filtered_df[filtered_df['Payload Mass (kg)'] <= entered_payload[1]]
    if entered_site == 'ALL':
        data = filtered_df
        fig = px.scatter(data, x ='Payload Mass (kg)', y = 'class',
        color='Booster Version Category',
        title='Correlation between Payload and Success for all Sites')
        return fig
    else:
        data = filtered_df[filtered_df['Launch Site'] == site_dict[entered_site]]
        fig = px.scatter(data, x ='Payload Mass (kg)', y = 'class',
        color='Booster Version Category',
        title=f'Correlation between Payload and Success for site {site_dict[entered_site]}')
        return fig    # return the outcomes piechart for a selected site
```

Thank you!