IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ray Abraham Thomas
April 18, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection
  - Data Wrangling
  - EDA with Data Visualization
  - EDA with SQL
  - Build an Interactive Map with Folium
  - Build a Dashboard with Plotly Dash
  - Predictive Analytics
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive analysis
  - Predictive analysis results

Introduction

- Project background and context:

  As competition grows in the space exploration industry, newly emerged company, Space Y, is leveraging predictive analytics to determine the cost of launching a space flight by determining whether the first stage will land safely. We will draw from data gathered from Space X rocket launches.

- Problems you want to find answers:
  - Can we determine whether the first stage will land successfully?
  - What factors contribute to the success of first stage landing?
  - Which machine learning algorithm yields the highest accuracy in predicting first stage landing success?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Gather data from SpaceX REST API and through web-scraping of the SpaceX Falcon 9 Wikipedia page.
- Perform data wrangling
  - Using Numpy and Pandas, first identify whether null values need to be corrected. Then convert the categorical labels to discrete numeric values to be usable with the machine learning algorithms.
- 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

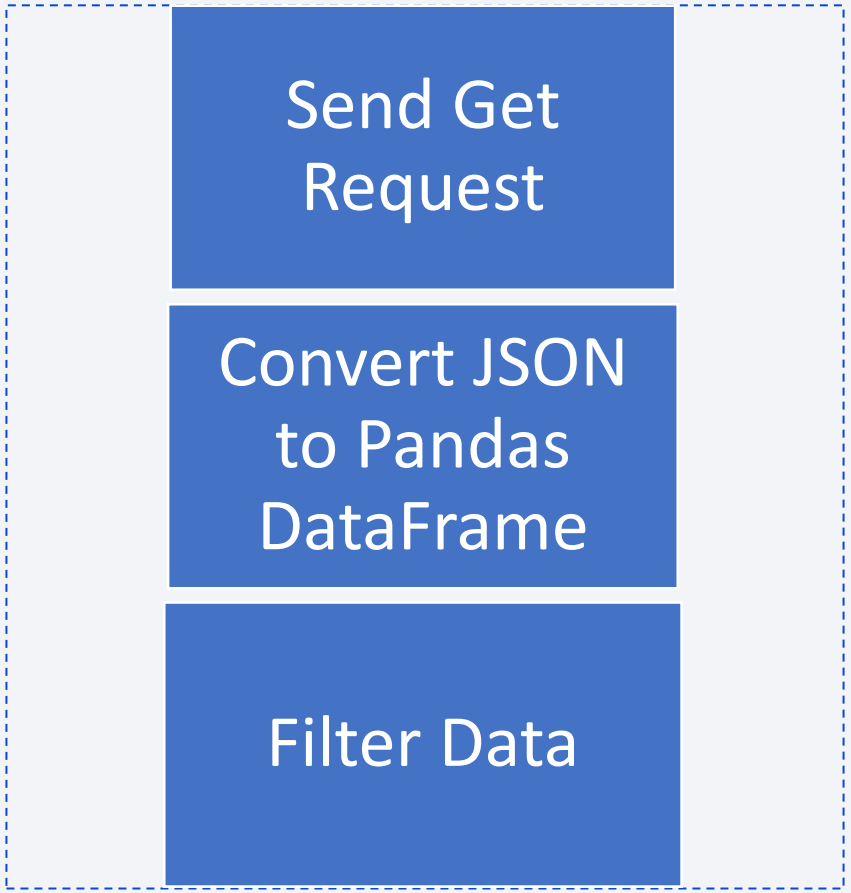
- How data sets were collected.
  1. We utilize SpaceX REST API to gather data about past Falcon 9 launches
  2. We also utilize Web Scraping of the Falcon 9 Wikipedia page to

Data Collection – SpaceX API

API Data Collection

- SpaceX REST API endpoints:
  - api.spacexdata.com/v4/launches/past
  - api.spacexdata.com/v4/rockets/
  - api.spacexdata.com/v4/launchpads/
  - api.spacexdata.com/v4/payloads/
  - api.spacexdata.com/v4/cores/
- Perform a GET request using the requests library to obtain launch data
- Convert the JSON response to a Pandas dataframe using json_normalize function

- 1. Hands-on Lab: Complete the Data Collection API Lab.ipynb

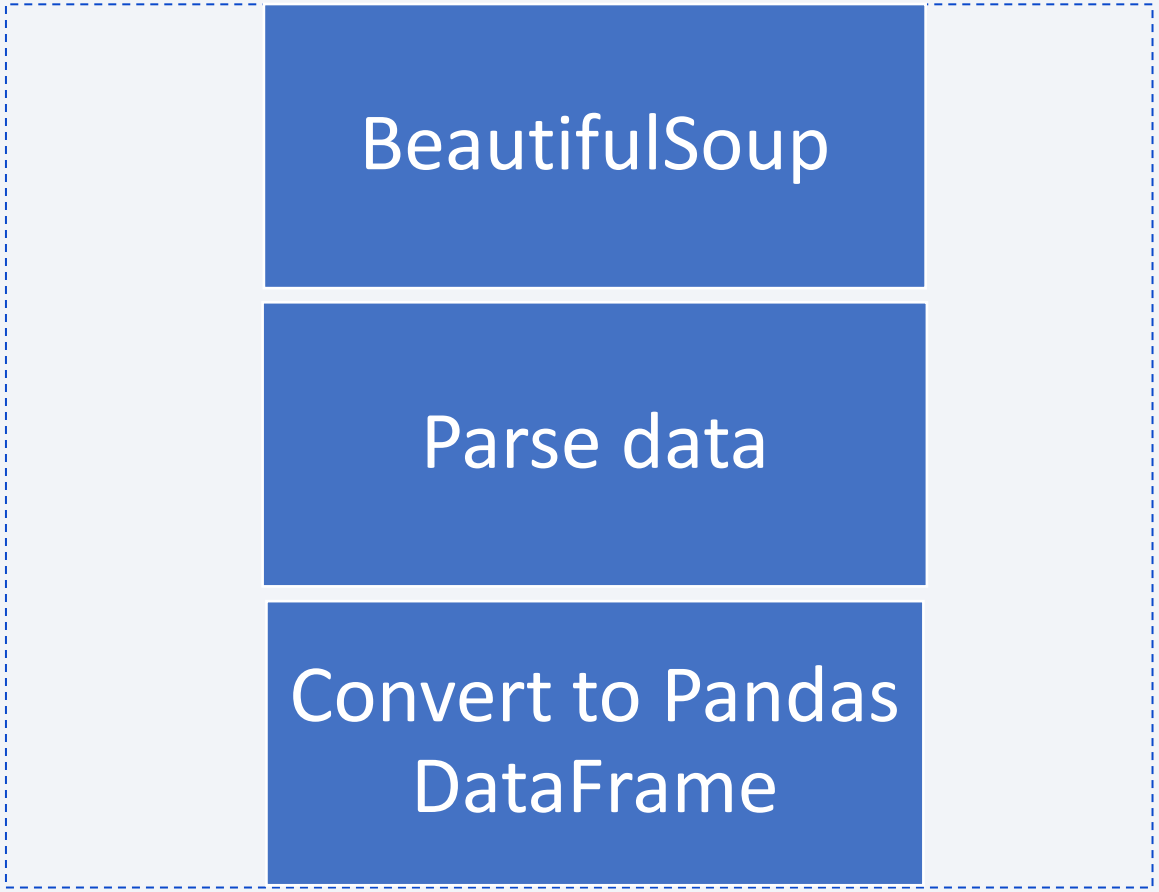


Data Collection - Scraping

Web Scraping Data Collection

- Use BeautifulSoup package to scrape HTML tables containing valuable Falcon 9 launch records
- Parse the data from those tables and convert them into a Pandas dataframe for further visualization and analysis

- 2. Hands-on Lab: Complete the Data Collection with Web Scraping Lab.ipynb



Data Wrangling

- The dataset from the previous section was loaded into the program using Pandas, followed by exploratory data analysis to identify patterns and determine training labels for supervised models. The completeness of the data was evaluated by calculating the percentage of missing values in each attribute. Finally, the mission outcomes were converted into training labels with "1" indicating a successful landing and "0" indicating an unsuccessful landing.
- 3. Hands-on Lab: Data Wrangling.ipynb

EDA with Data Visualization

- Different visualizations were used to explore the relationship between the Flight Number and Payload, Launch Site and Payload, and the success rate of each orbit type. The following charts were plotted:
  - Scatter plot of PayloadMass vs. FlightNumber with the outcome of the launch represented by different colors using the seaborn catplot.
  - Scatter plot of LaunchSite vs. FlightNumber with the outcome of the launch represented by different colors using the seaborn catplot.
  - Scatter plot of LaunchSite vs. PayloadMass with the outcome of the launch represented by different colors using the seaborn catplot.
  - Bar chart of success rate for each orbit type using the seaborn barplot.
- 5. Hands-on Lab: Complete the EDA with Visualization Lab.ipynb

EDA with SQL

- During the EDA phase, SQL queries were performed to identify the following:
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA'
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was acheived.
- 4. Hands-on Lab: Complete the EDA with SQL.ipynb

Build an Interactive Map with Folium

- The following map objects were created and added to a Folium map:
  - Markers were added to mark the locations of launch sites on the map. Each marker represents a launch site and is placed on the map based on its latitude and longitude coordinates. The markers help identify the specific locations of the launch sites.
  - Circles were added to create highlighted circle areas around the launch sites. These circles represent the proximity or coverage area of each launch site. The circles are centered at the latitude and longitude coordinates of the launch sites and have a specified radius. They provide a visual representation of the area around each launch site.
- These objects were added to the map to provide an interactive visual representation of the launch sites, their proximity to certain factors like the coast, and the success rates associated with each site. The map allows for easier analysis and exploration of geographical patterns related to the launch sites.
- 6. Hands-on Lab: Complete the Interactive Visual Analytics with Folium Lab.ipynb

Build a Dashboard with Plotly Dash

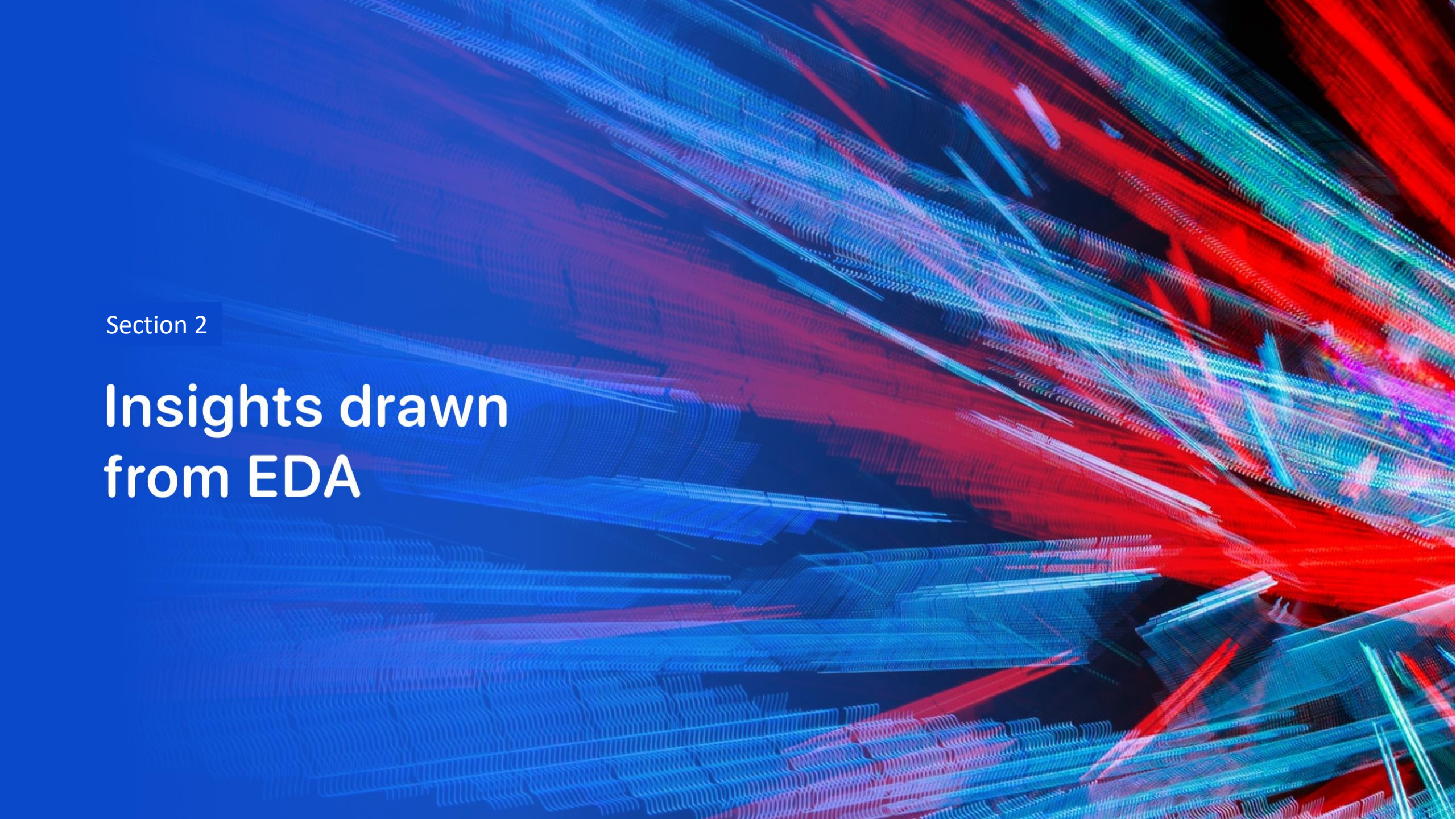
- The dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart. The tasks involved in building the dashboard include adding a launch site drop-down input component, adding a callback function to render success-pie-chart based on selected site dropdown, adding a range slider to select payload, and adding a callback function to render the success-payload-scatter-chart scatter plot.
- The purpose of adding these plots and interactions is to enable users to obtain some insights about the SpaceX launch data and answer questions such as which site has the largest successful launches, which site has the highest launch success rate, which payload range(s) has the highest launch success rate, which payload range(s) has the lowest launch success rate, and which F9 Booster version has the highest launch success rate.
- 7. Hands-on Lab: Build an Interactive Dashboard with Ploty Dash.ipynb

Predictive Analysis (Classification)

- The machine learning pipeline was developed following the steps below:
    1. Load the dataset using Pandas
    2. Standardize the data using preprocessing from scikit-learn.
    3. Split the data into training and test datasets using train_test_split from scikit-learn.
    4. Define the hyperparameters to test for each model using GridSearchCV from scikit-learn.
    5. Train and evaluate the performance of the Logistic Regression, SVM, Decision Tree, and K-Nearest Neighbors models on the training dataset.
    6. Select the best-performing model based on the evaluation results on the test dataset.
    7. Plot the confusion matrix to visualize the performance of the model.
- 8. Hands-on Lab: Complete the Machine Learning Prediction Lab.ipynb

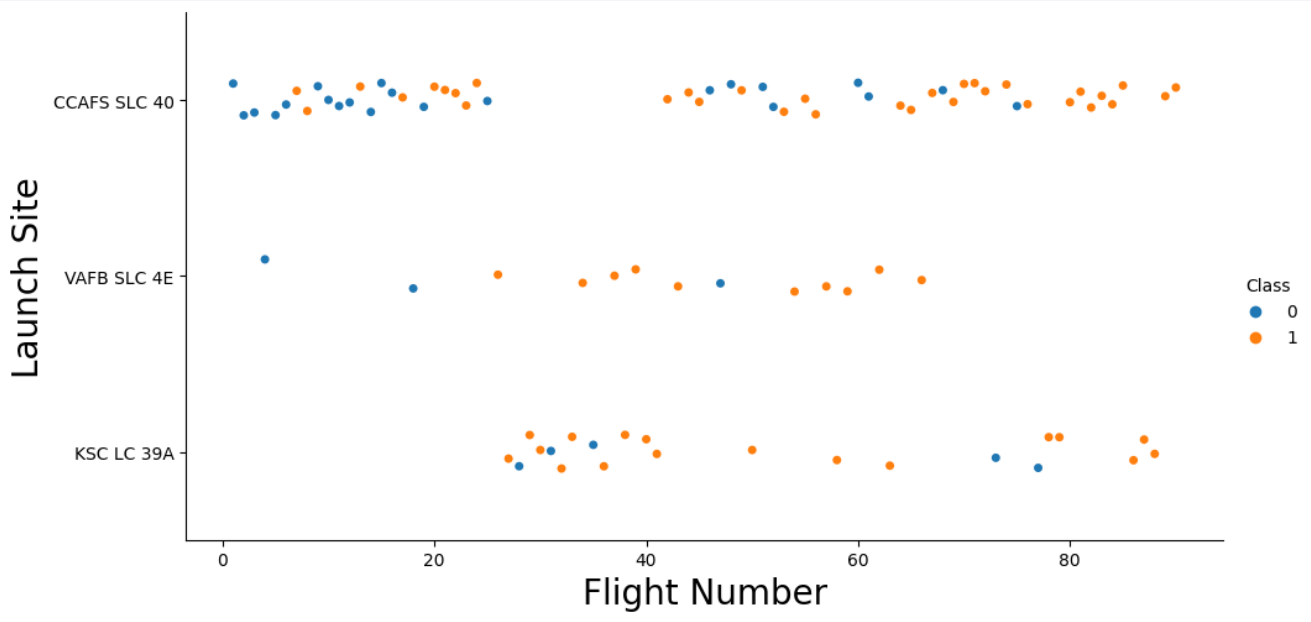
Results

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

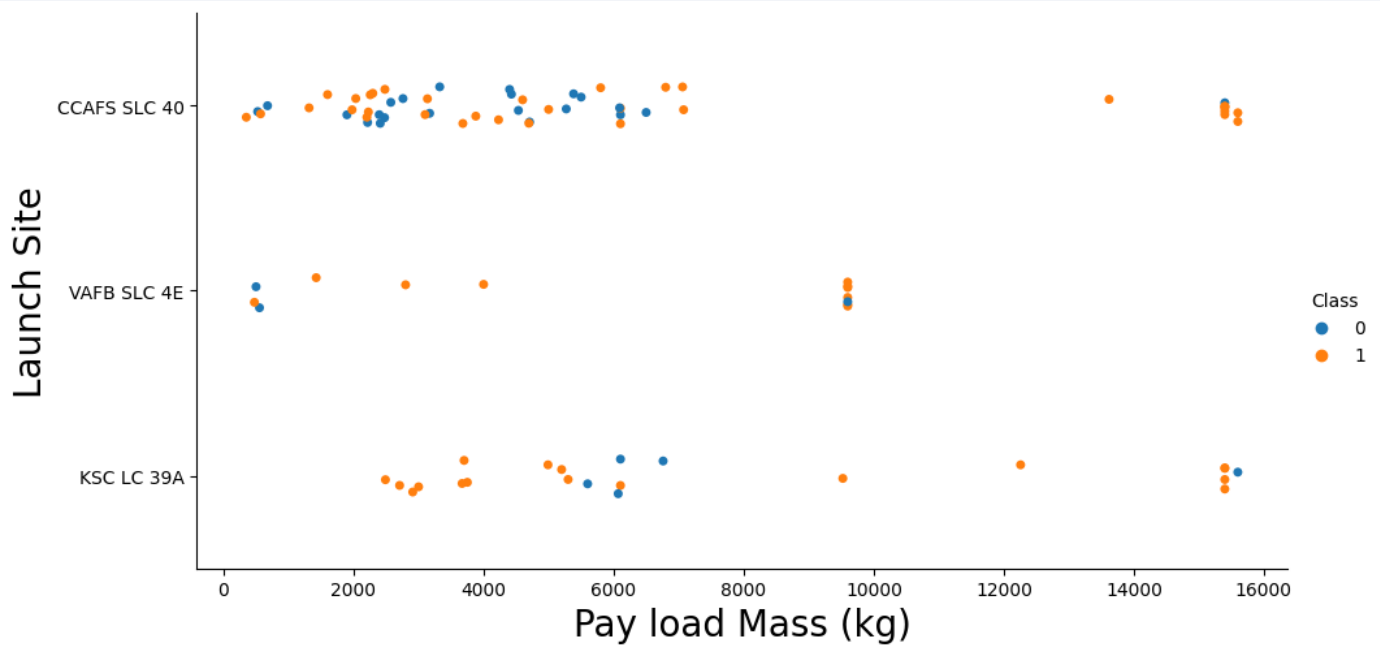

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

- From the plot, we can identify that a large set of the early flights were conducted at CCAFS SLC 40.
- We see many failures in those early flights.
- As the flight number increases, we see in all 3 launch sites an increase in a successful outcome.

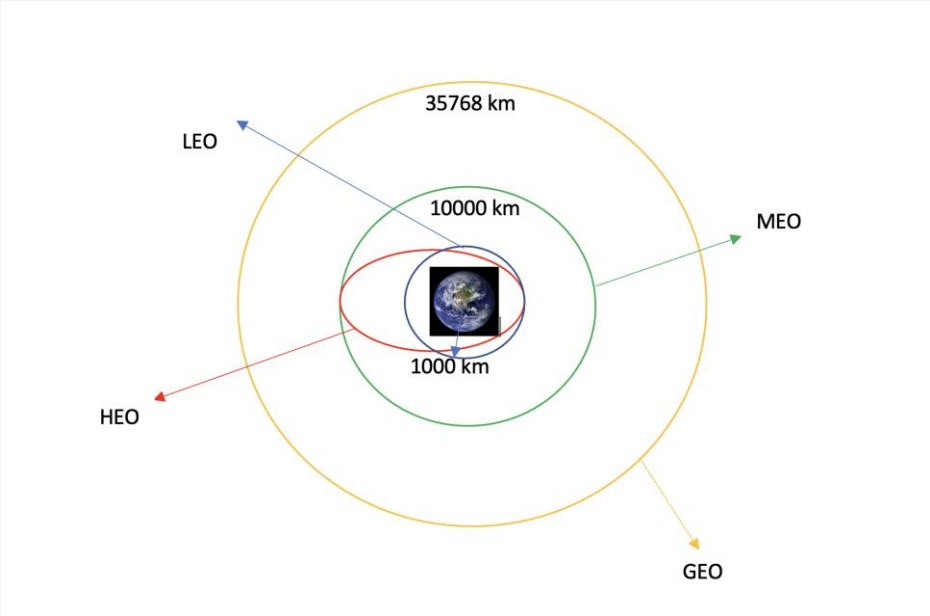


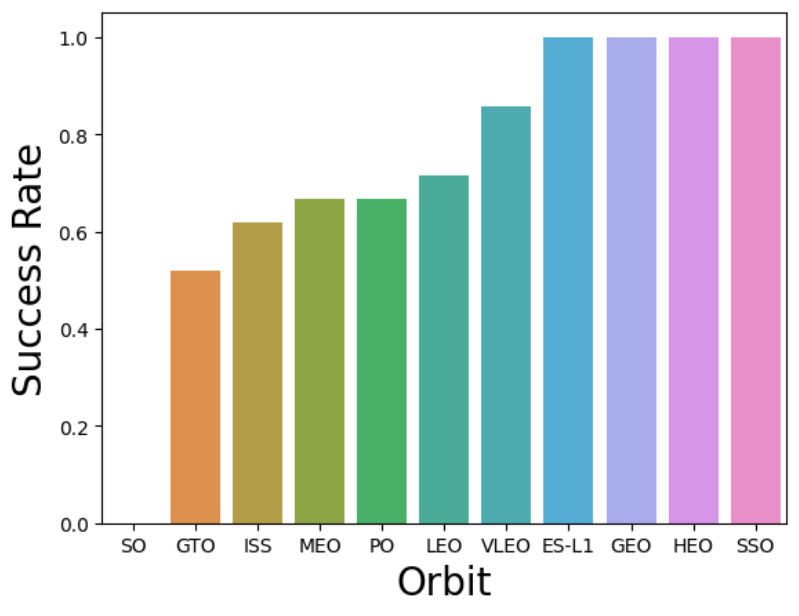
Payload vs. Launch Site



- We notice for CCAFS SLC 40, that as the pay load mass increases above 8000 kg, the probability of success is much higher.
- Note that VAFB-SLC does not contain any payloads of a mass greater than 10000 kg. But most of the flights are successful even though they are on the lower end of the payload mass.
- For KSC LC 39A, the payload mass ranges from greater than 2000 kg to less than 16000 kg. Of those all but 4 near 6000 kg and 1 near 15000 kg are successes.

Success Rate vs. Orbit Type

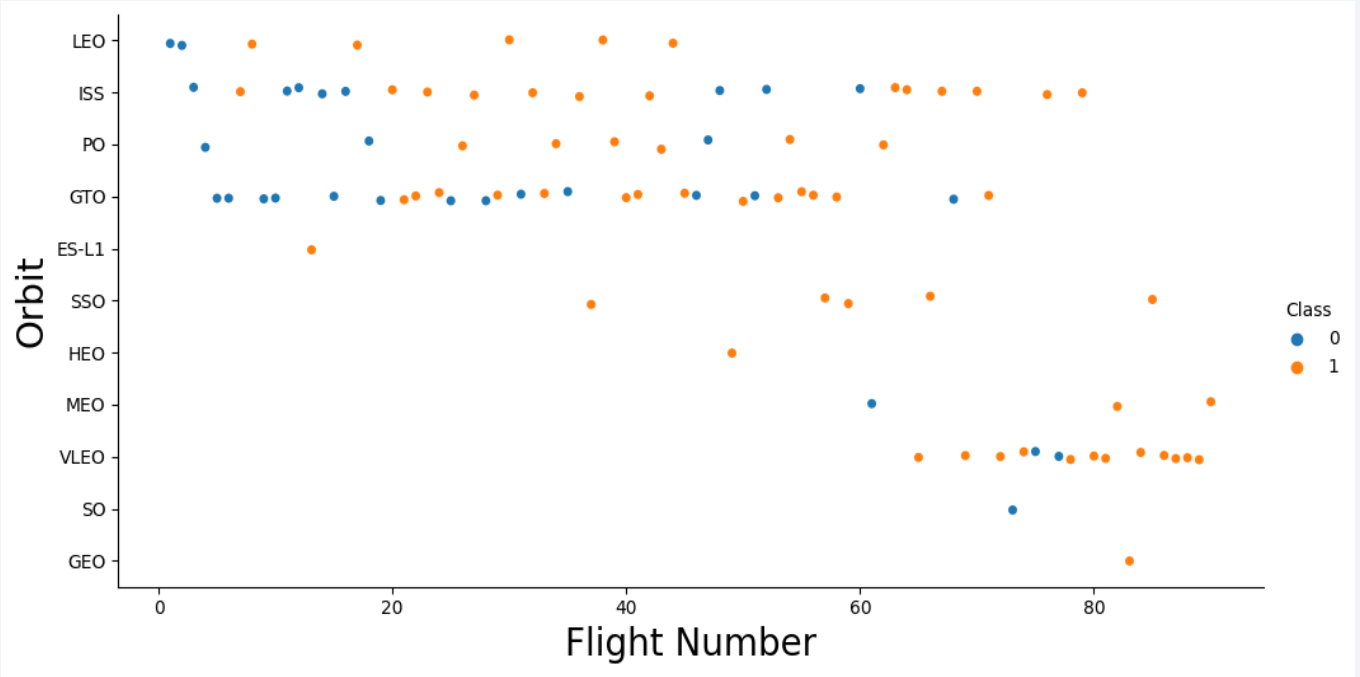




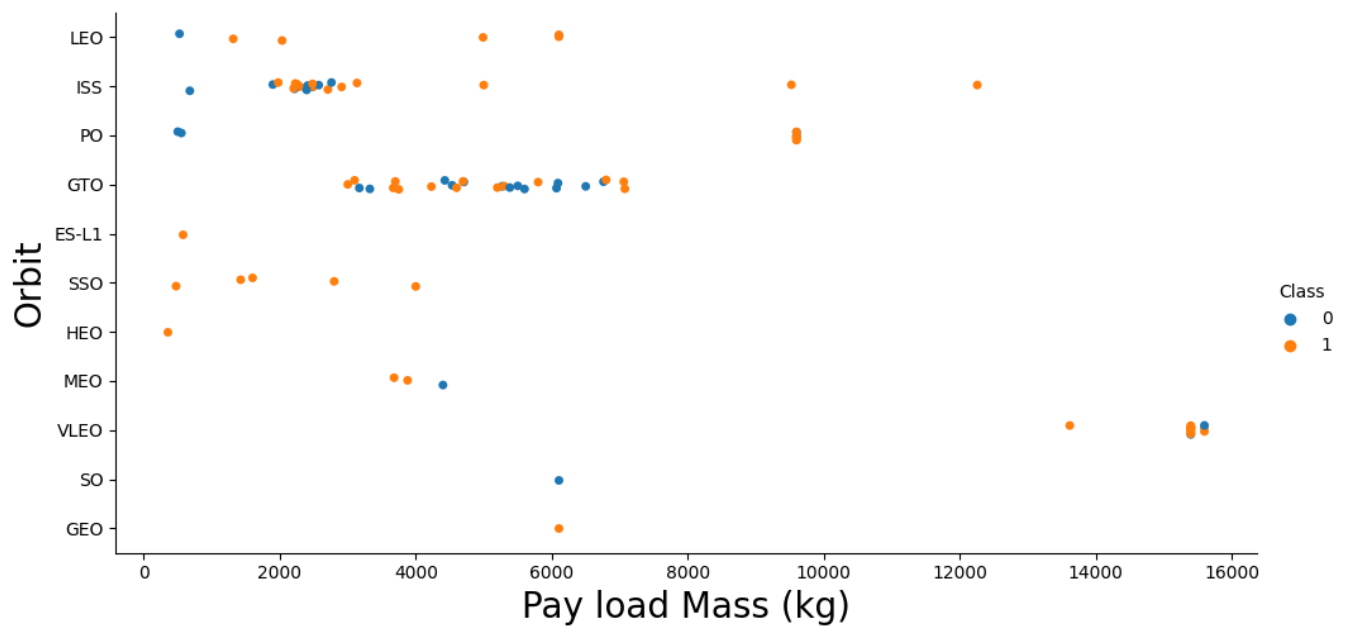
- We also see the ES-L1 lagrange point orbit, geosynchronous orbit (GEO), orbits beyond the geosynchronous orbit (HEO), and Sun-synchronous (SSO) have the highest success rates.
- Note that even though SO and SSO are identified to be equivalent, SO has a success rate of 0 while SSO has a success rate of 1. Further consideration need to be made regarding this discrepancy.

Flight Number vs. Orbit Type

- LEO, ISS, PO, GTO, and ES-L1 orbits contain a majority of the early flights. These orbits also have a wide range of flight numbers, up to nearly the 80th flight. They have a good mixture of successes and failures.
- SSO, HEO, MEO, VLEO, SO, and GEO were all orbits attempted after roughly 30 flights. These flights contain only 4 failures.



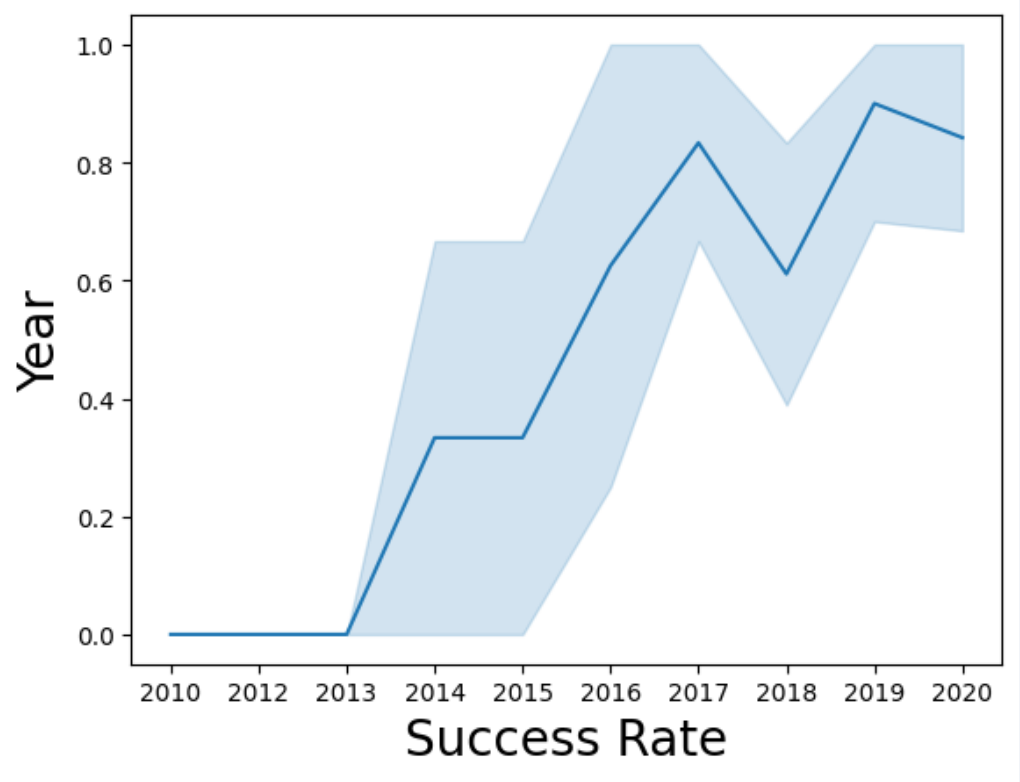
Payload vs. Orbit Type



- SO and GEO have only one flight each, both weighing 6000 kg.
- VLEO has a payload mass of > 13000 kg.
- SSO has payloads ranging from 0 to 4000 kg with total success.
- HEO only has one flight with near 0 kg. It was successful.
- LEO, ISS, and PO have pay loads ranging from 0 to 12000, with more successes in the flights with heavier payloads.
- GTO has a range of 3000 to 7000 kg with mixed success rates.

Launch Success Yearly Trend

- The early few years (from 2010 to 2013) had very low success rates.
- But since 2013 to2020, the success rate has steadily increased with a dip from 2017 to 2018 and a slightly smaller dip from 2019 to 2020.
- Need to research reasons behind the dips.



All Launch Site Names

- The SELECT DISTINCT statement identifies the unique values in the Launch_Site column.

```
In [ ]: %%sql
        SELECT DISTINCT("Launch_Site") FROM SPACEXTBL;
```

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

Out[ ]:

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

Launch Site Names Begin with 'CCA'

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

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

Out[ ]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- We ask for all columns where the Launch_Site value begins with ‘CCA’, limiting to just the first 5 records.

Total Payload Mass

```
In [ ]:
%%sql
SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL
WHERE "Customer" = "NASA (CRS)";

 * sqlite:///my_data1.db
Done.
Out[ ]:
SUM("PAYLOAD_MASS__KG_")
45596
```

- We add all the values in the Payload_Mass__KG_ column that correspond to the customer with the value equal to “NASA (CRS)”

Average Payload Mass by F9 v1.1

- We get the average value of all instances in the “Payload_Mass__Kg_” where the “Booster_Version” column is equal to “F9 v1.1”

In [ ]:
```
%%sql
SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL
WHERE "Booster_Version" LIKE "F9 v1.1%";
```

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

Out[ ]:

AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

First Successful Ground Landing Date

- This query finds the smallest date corresponding to the “Landing _Outcome” column equaling “Success (ground pad)”.

In [ ]:
```
%%sql
SELECT MIN("Date") FROM SPACEXTBL
WHERE "Landing _Outcome" = "Success (ground pad)";
```

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

Out[ ]:

MIN("Date")
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

In [ ]:

```
%%sql
SELECT "Booster_Version" FROM SPACEXTBL
WHERE "Landing _Outcome" = "Success (drone ship)"
AND "PAYLOAD_MASS__KG_" BETWEEN 4001 AND 5999;
```

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

Out[ ]:

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

- In this query, we identify the name of the “Booster_Version” where the “Landing _Outcome” equals “Success (drone ship)” and the Payload Mass is between 4000 and 6000 kg.

Total Number of Successful and Failure Mission Outcomes

- This query identifies the total number of each unique Mission Outcome by utilizing the COUNT statement and grouping by the unique Mission Outcomes.

In [ ]:
```
%%sql
SELECT "Mission_Outcome", COUNT("Mission_Outcome") FROM SPACEXTBL
GROUP BY "Mission_Outcome";
```

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

Out[ ]:

Mission_Outcome	COUNT("Mission_Outcome")
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

```
In [ ]: %%sql
        SELECT DISTINCT("Booster_Version") FROM SPACEXTBL
        WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTBL);

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

Out[ ]:

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

- This query utilizes the DISTINCT statement to find the Booster_Version names that have carried the MAX payload mass.

2015 Launch Records

- This query manually inputs month names by extracting the month number from the Date column. Then it identifies the month, booster version name, and launch site for all the failed drone ship landings which occurred in the year 2015.

In [ ]:

```
%%sql
SELECT CASE WHEN SUBSTR("Date", 4, 2) = '01' THEN 'January'
            WHEN SUBSTR("Date", 4, 2) = '02' THEN 'February'
            WHEN SUBSTR("Date", 4, 2) = '03' THEN 'March'
            WHEN SUBSTR("Date", 4, 2) = '04' THEN 'April'
            WHEN SUBSTR("Date", 4, 2) = '05' THEN 'May'
            WHEN SUBSTR("Date", 4, 2) = '06' THEN 'June'
            WHEN SUBSTR("Date", 4, 2) = '07' THEN 'July'
            WHEN SUBSTR("Date", 4, 2) = '08' THEN 'August'
            WHEN SUBSTR("Date", 4, 2) = '09' THEN 'September'
            WHEN SUBSTR("Date", 4, 2) = '10' THEN 'October'
            WHEN SUBSTR("Date", 4, 2) = '11' THEN 'November'
            WHEN SUBSTR("Date", 4, 2) = '12' THEN 'December'
        END AS Month,
"Landing _Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL
WHERE "Landing _Outcome" = "Failure (drone ship)"
AND SUBSTR("Date", 7, 4) = '2015';
```

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

Out[ ]:

Month	Landing _Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

In [ ]:

```
%%sql
SELECT "Landing _Outcome",
    COUNT("Landing _Outcome") AS Count,
    RANK() OVER (ORDER BY COUNT("Landing _Outcomes") DESC) AS Rank
FROM SPACEXTBL
WHERE "Landing _Outcome" Like "Success%"
AND "Date" BETWEEN "04-06-2010" AND "20-03-2017"
GROUP BY "Landing _Outcome"
ORDER BY COUNT("Landing _Outcome") DESC;
```

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

Out[ ]:

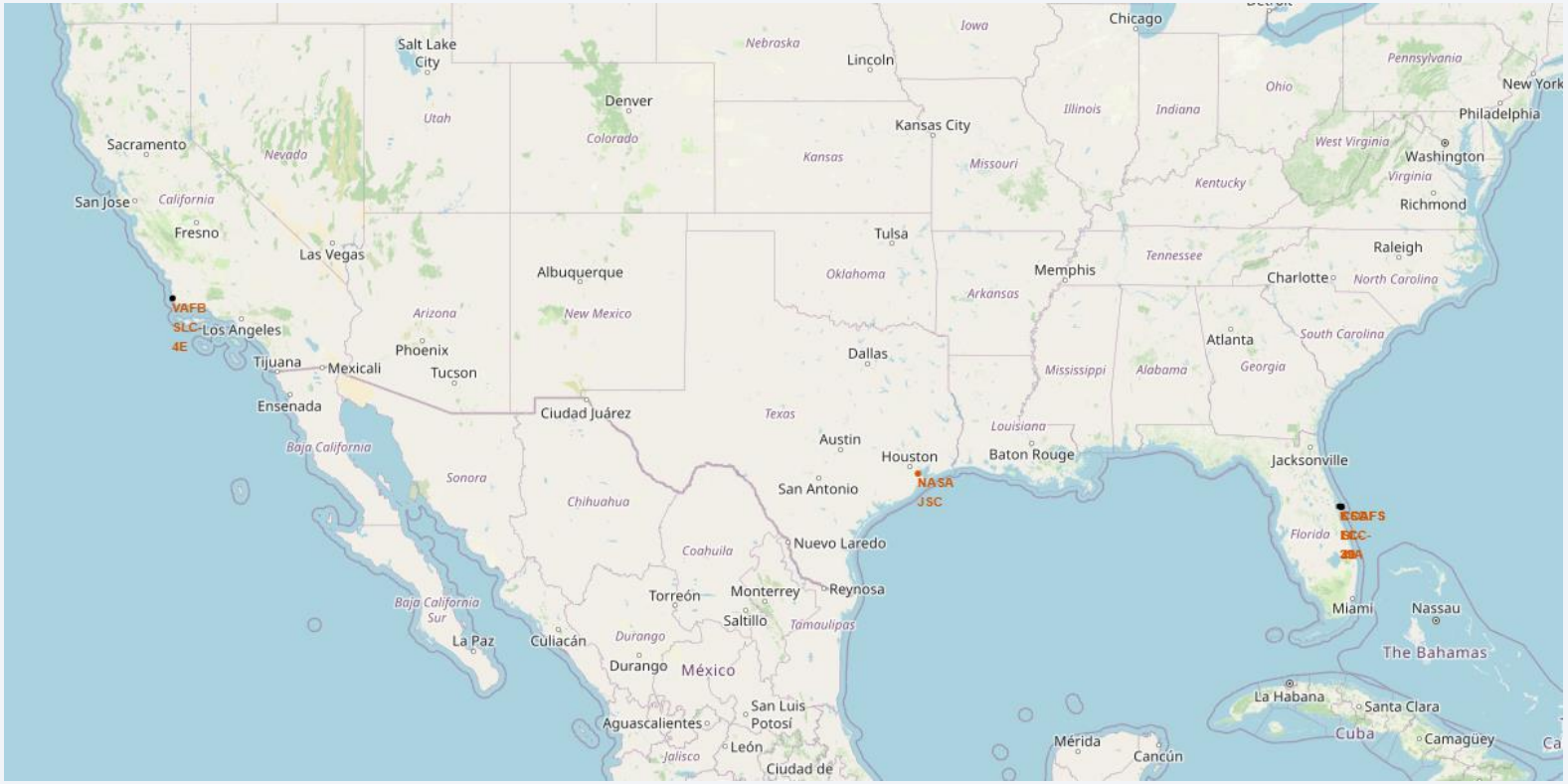
Landing _Outcome	Count	Rank
Success	20	1
Success (drone ship)	8	2
Success (ground pad)	6	3

- This query counts the unique type of success landing outcomes and ranks them from largest to smallest.

Section 3

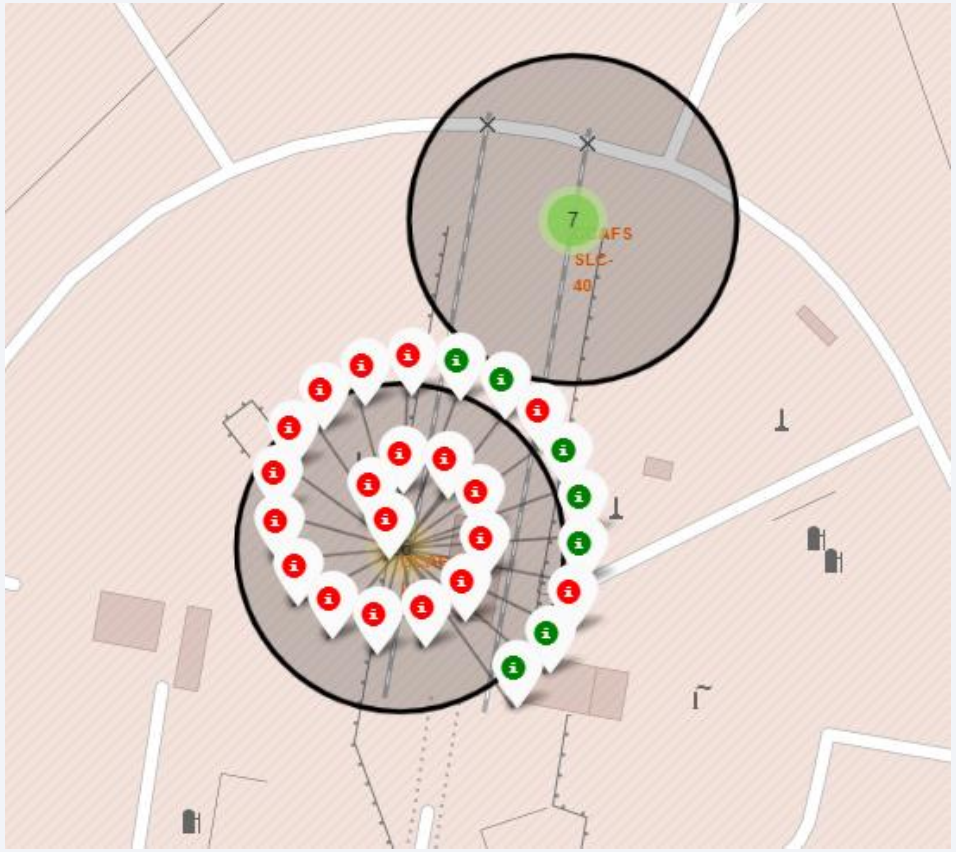
Launch Sites Proximities Analysis

Map of all Launch Sites



- In this map, we have marked and labeled each of the launch sites.

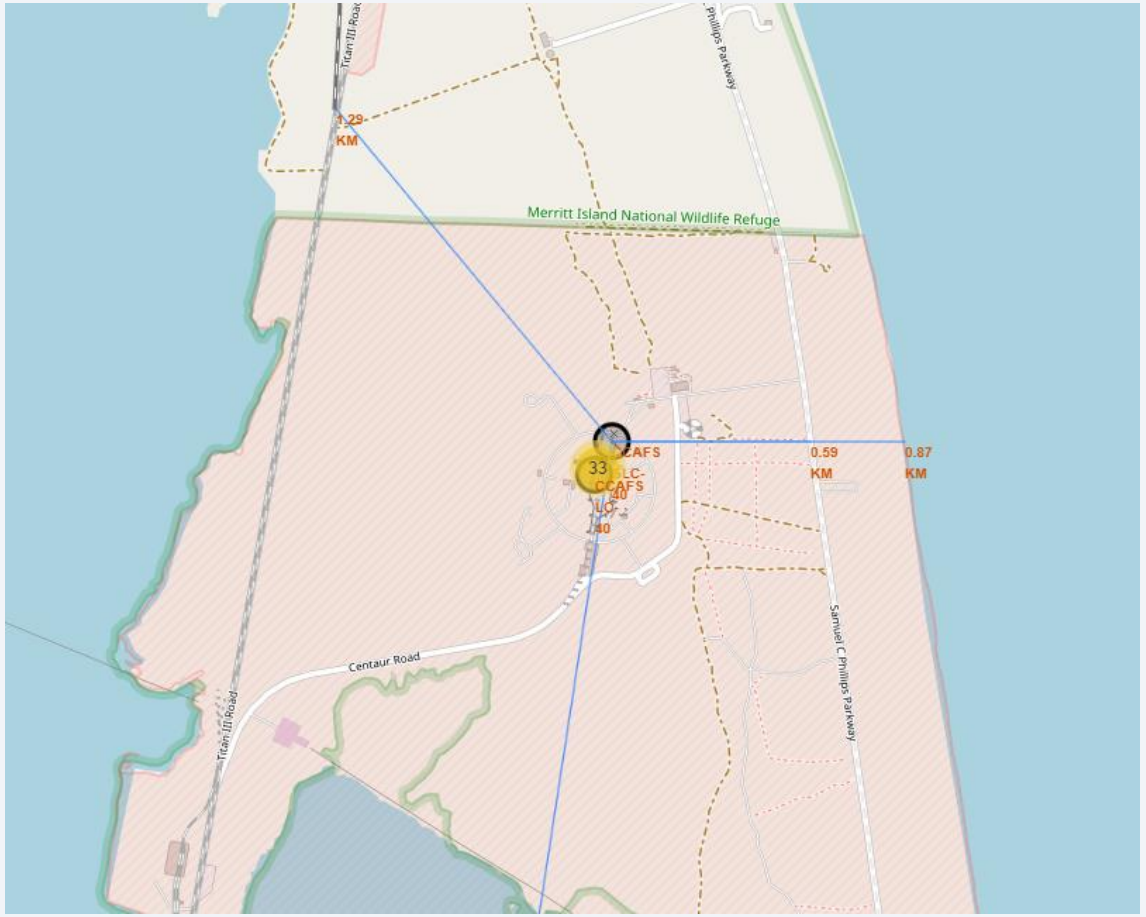
Markers indicating Success/Failure by Color

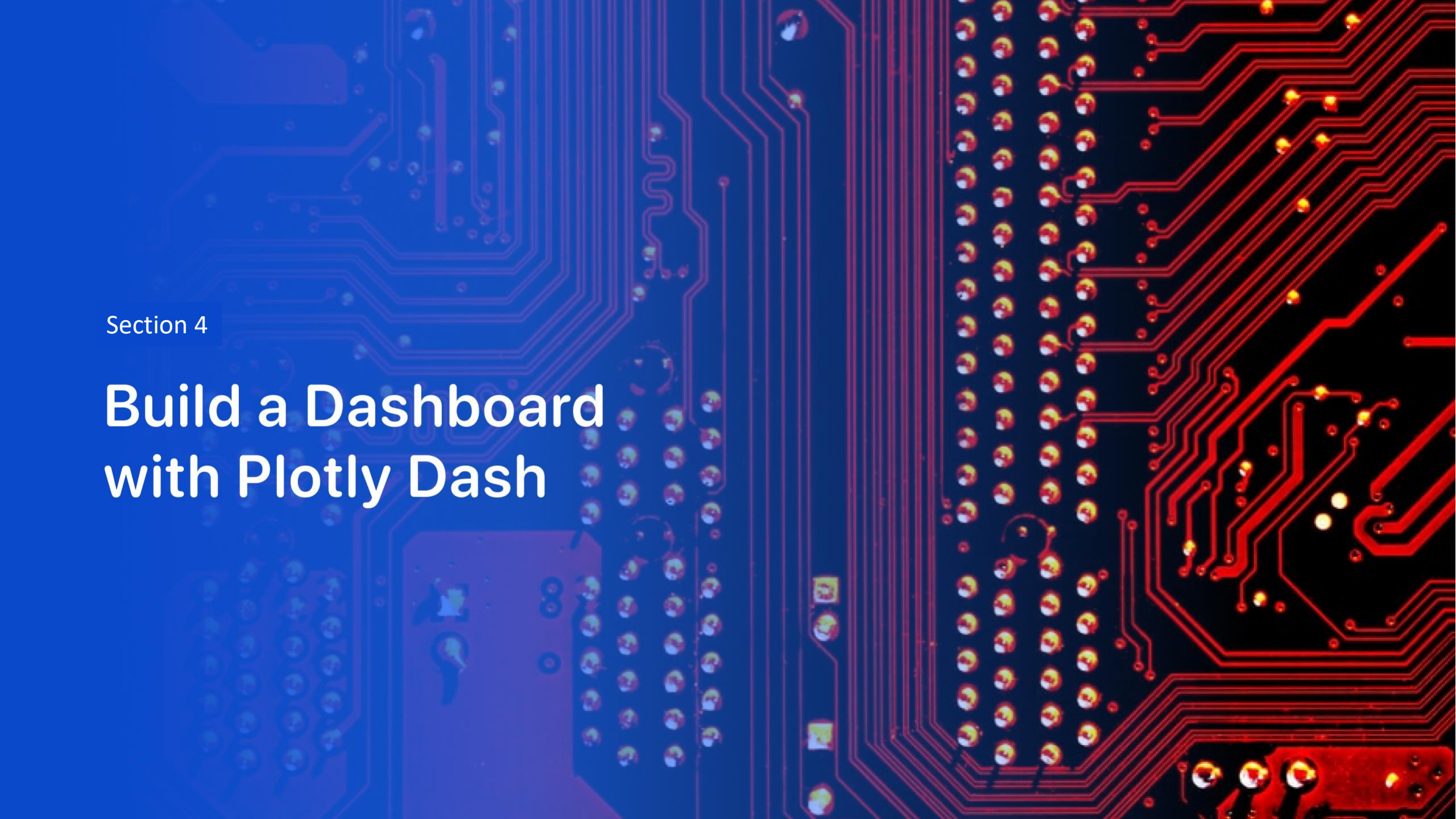


- For each site, we have placed GREEN markers for each successful landing and RED markers for each failed landing.

Proximities to Nearest Landmarks

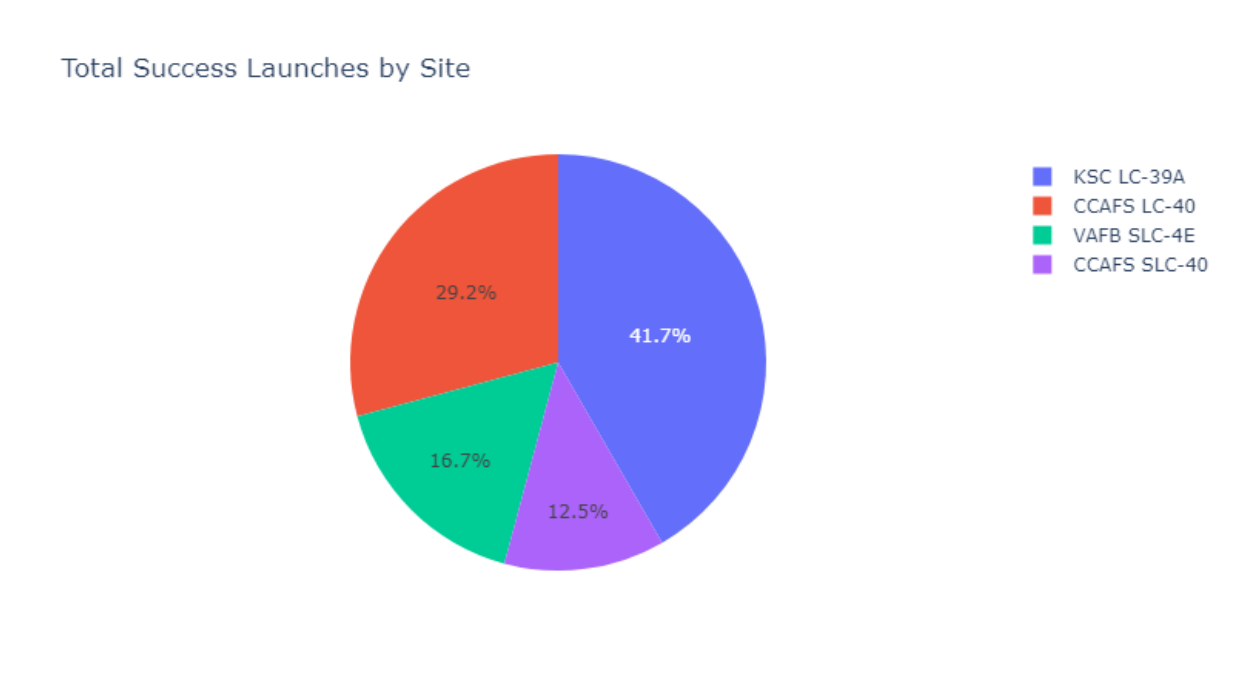
- The distances to the nearest coastline, highway, and railway are labeled with a line showing the direction in which these landmarks lie in relation to the launch site.



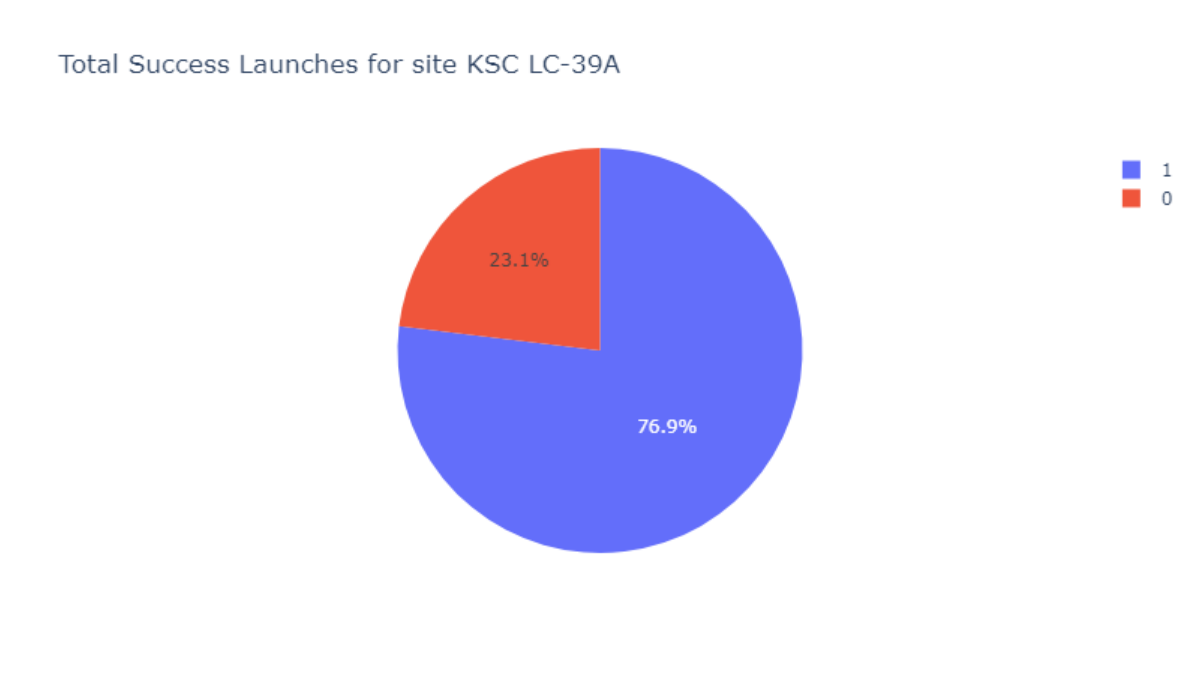

Section 4
Build a Dashboard with Plotly Dash

Pie Chart of Launch Success Counts for All Sites

- KSC LC-39A has the largest overall success rate at 41.7%.

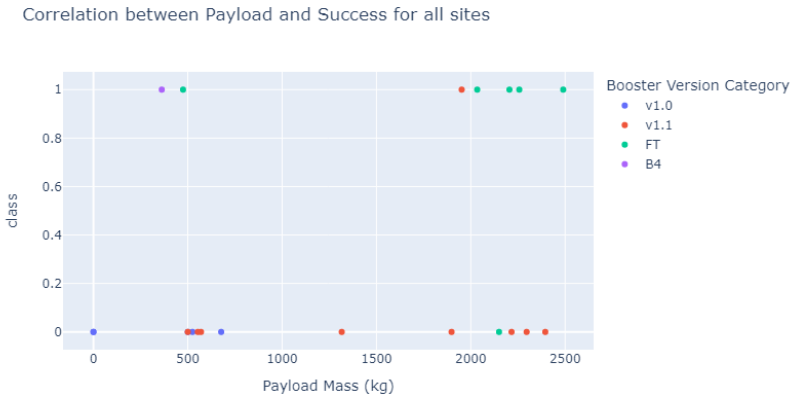


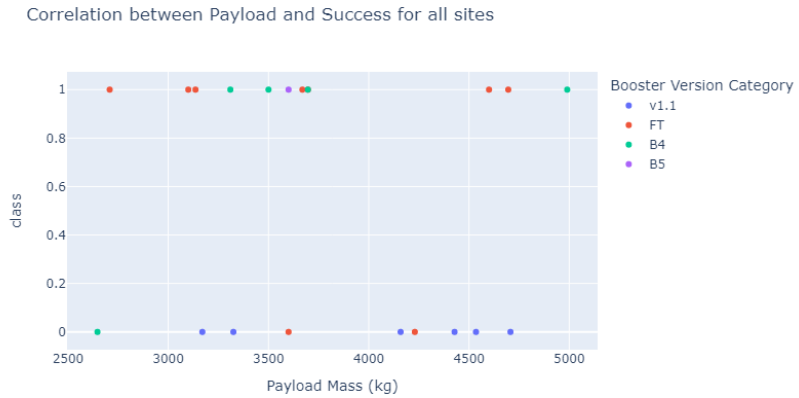
Pie Chart of Launch Site with Highest Success Ratio

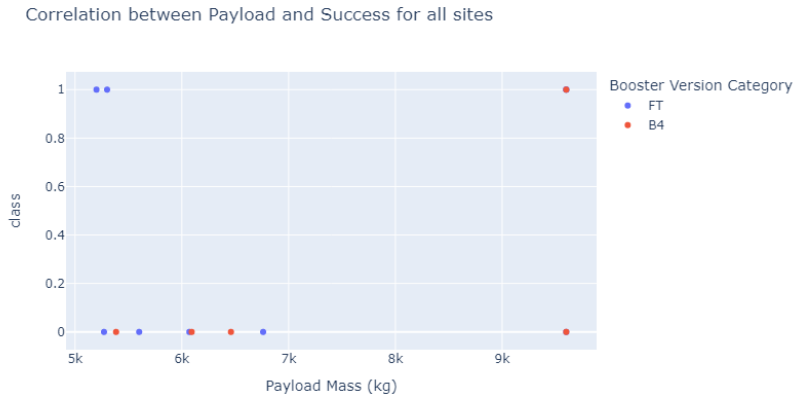


- The success rate among all the launches conducted at KSC LC-39A is 76.9%.

Scatter Plot of Payload vs. Launch Outcomes for all Sites







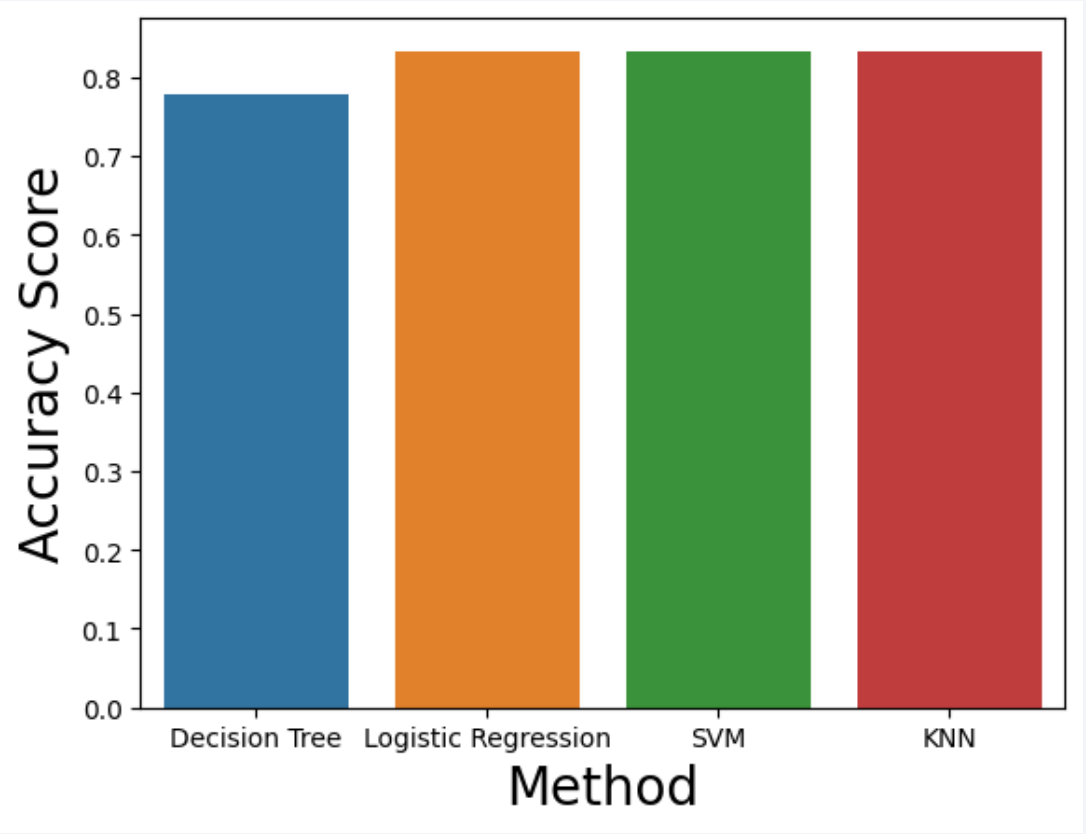
- The above 3 charts show the correlation between payload and success for the payload mass range of 0 – 2500 kg (left), 2500 – 5000 kg (middle), and 5000 – 10000 kg (right).
- In the first and second plot, we see that the FT, B4, and B5 boosters have the highest success rate.
- In the third plot, we see that after 5000 kg, the FT booster's success rate also begins to diminish.

Section 5

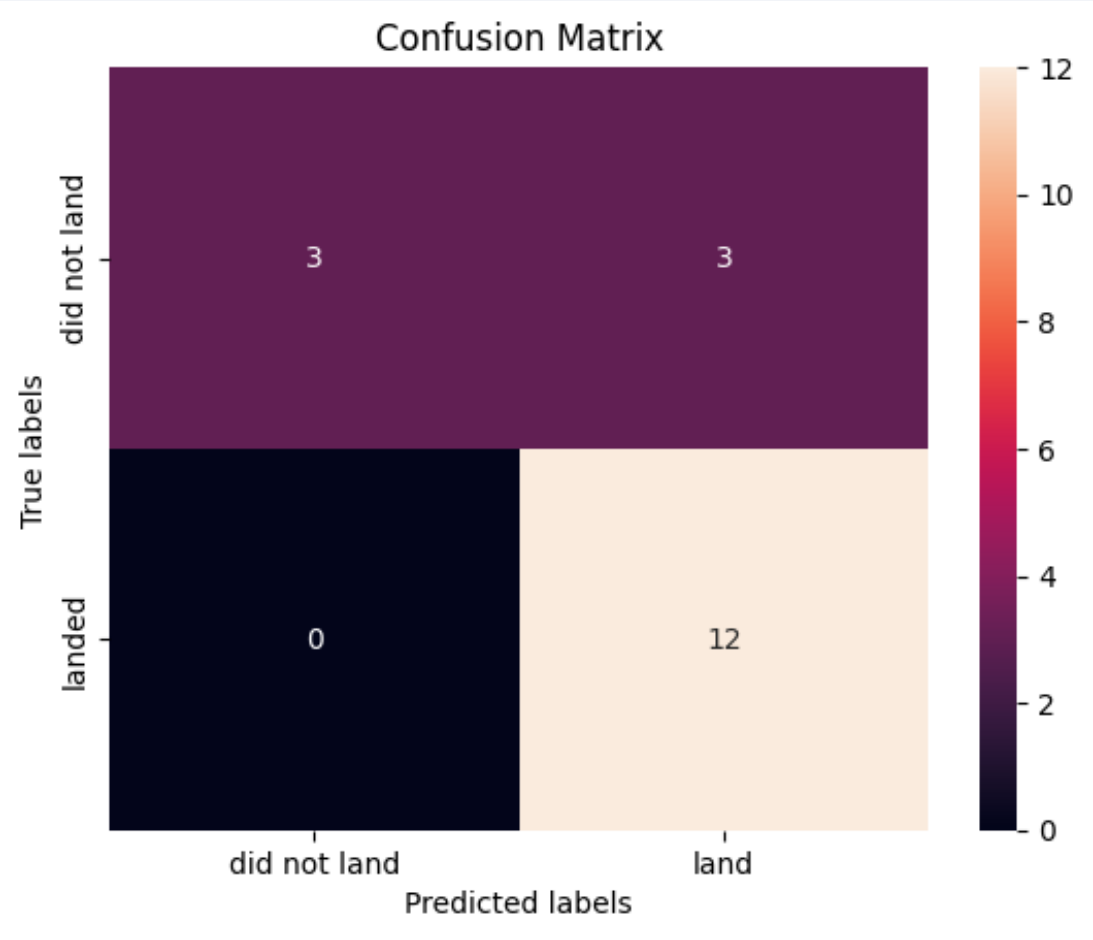
Predictive Analysis (Classification)

Classification Accuracy

- We see that all of the methods except for Decision Tree yield an accuracy of 83.33%
- This indicates that Logistic Regression, SVM, and KNN are all equally viable algorithmic options to choose from. Therefore, we should choose from the algorithm with the least computational cost.
- Note that a larger dataset with different training/testing split may yield differing results. This needs to be further explored.



Confusion Matrix

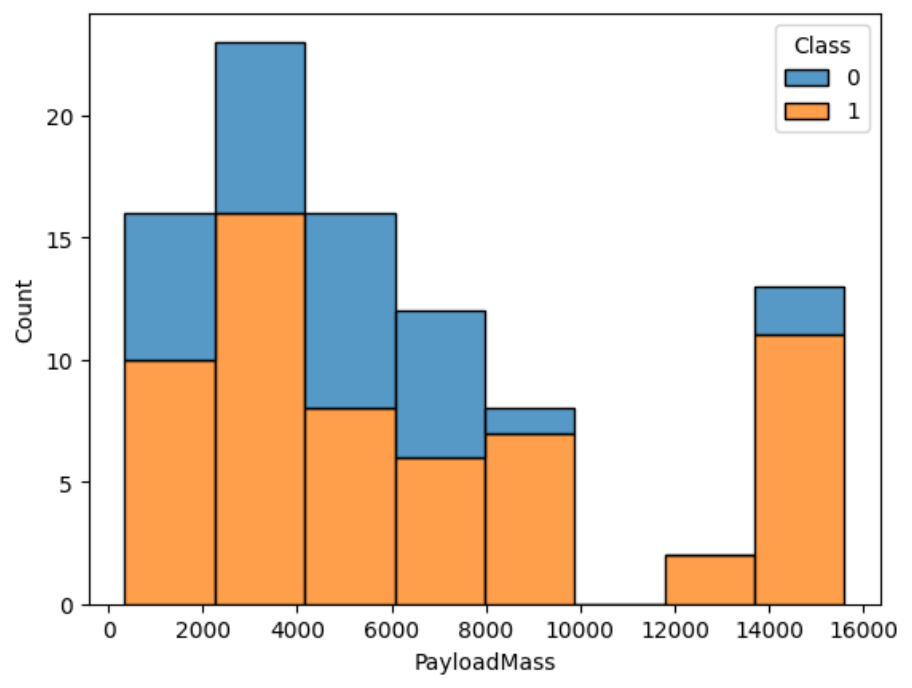


- In this confusion matrix, we see that the biggest concern is the case when the false positive rate being 1/6 (16.67%).
- More data will likely increase the accuracy rate for this model.

Conclusions

- As the number of flights increased, the success rate also began to increase.
- The site with the best success rate is KSC LC-39A.
- Excluding decision tree, the remaining algorithms all similar results. It is recommended to choose the model with the lowest computational cost.
- Refining the models with more data will yield more accurate results.

Appendix



- The above histogram identifies the distribution of flights by payload mass.
- You can access the full project repository in GitHub through this link.

Thank you!