IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Chandrika Shenoy>
<29-11-2024>

Outline

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

Executive Summary

- **Summary of methodologies**
  - Data Collection via API, Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis (EDA) with Data Visualization
  - EDA with SQL
  - Interactive Visual Analytics with Folium
  - Dashboards with Plotly Dash
  - Predictive Analysis

- **Summary of All Results**
  - Exploratory Data Analysis Results
  - Interactive Maps and Dashboards
  - Predictive Analysis Results

Introduction

- **Project background and context**

  The commercial space industry is rapidly evolving, with companies like Virgin Galactic, Rocket Lab, and SpaceX leading the way. .SpaceX has achieved significant milestones, including sending spacecraft to the International Space Station and launching the Starlink satellite internet constellation. A key factor in SpaceX's success is the ability to reuse the first stage of their Falcon 9 rockets, which significantly reduces launch costs. SpaceX mentions on its website that the Falcon 9 rocket launch cost 62 million dollars while other providers cost upward of 165 million dollars each. The price difference is due to the fact that SpaceX can reuse the first stage. By determining if the stage will land, we can determine the cost of a launch. This information is key to any other company that plans to compete with SpaceX for a rocket launch.

- **Problems you want to find answers**
  - What can be termed as a successful or a failed landing?
  - How do the rocket variables determine the success or failure of a landing?
  - Which conditions allow SpaceX to get the best landing success rate?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API
  - Webscraping from SpaceX (Falcon 9) Wikipidea page
- Perform data wrangling
  - Binary Classification for Outcome column in the dataset
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardize Data
  - Split into training and test data
  - Find hyperparameter for classification methods (SVM, Decision Trees, Logistic Regression)
  - Find the best method using test data

Data Collection

Datasets were collected in the following two ways:

Rest Space X API	Webscraping
Rocket Specifications: Booster Name Payload Mass Type of Landing Launch Details: **Launch Site** Latitude Longitude Orbit **Flight Outcomes:** Landing Outcome Number of Flights	Falcon 9 launch records, landing and payload information

Data Collection – SpaceX API

[Data Collection SpaceX API Code]

Data Collection - Scraping

Import libraries → Define Helper functions → HTTP request to SpaceX Wikipidea page → Parse HTML response using Beautiful soup → Extract Falcon 9 launch records from the HTML table → Convert Data to Pandas dataframe → Export dataframe to CSV.

[Data Collection Webscraping Code]

Data Wrangling

1. Import Libraries and Define Functions.
2. Load SpaceX Dataset.
3. Handle missing values.
4. Identify columns that are numerical and categorical.
5. Calculate the number of launches at each launch site.
6. Calculate the number and occurrence of each orbit.
7. Determine the number of landing_outcomes.
8. Create a landing outcome label from Outcome column and using the value (0 or 1) determine if the landing was a success or failure.
9. Calculate Success Rate.
10. Export dataframe to CSV.

[Data Wrangling Code]

EDA with Data Visualization

Scatter Plots	Bar Chart	Line Chart
• Flight Number vs Payload Mass • Flight Number vs Launch Site • Payload Mass vs Launch Site • Orbit vs Flight Number • Orbit vs Payload Mass	• Success Rate vs Orbit	• Success Rate vs Date

[EDA with Data Visualisation Code]

EDA with SQL

The following SQL queries were performed to understand data from the dataset:

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string 'CCA'
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome in ground pad was achieved.
6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
7. List the total number of successful and failure mission outcomes
8. List the names of the booster versions which have carried the maximum payload mass.
9. List the records which will display the month names, failure landing outcomes in drone ship ,booster versions, launch site for the months in year 2015.
10. Rank the count of landing outcomes between the date 04/06/2010 and 20/03/2017 in descending order.

[EDA with SQL Code]

Build an Interactive Map with Folium

Interactive Map with Folium Summary

- **Folium Map:** Used to create a map centered at NASA Johnson Space Center, Houston, Texas.
- **Highlighted Circle:** Added using folium.Circle to highlight specific areas.
- **Text Labels:** Used folium.Marker to add labels at specific coordinates.
- **MarkerCluster:** Displayed multiple types of information at the same coordinates.
- **Marker Colors:** Added a marker_color column in the dataframe to show:
    - Green for successful launches (class=1).
    - Red for failed launches (class=0).
- **Mouse Position:** Used MousePosition to display coordinates as the mouse moves over the map.
- **Distance Marker:** Used DistanceMarker to measure distances between launch sites and key locations (e.g., railways, coastlines).
- **Polyline:** Used to draw lines connecting launch sites to key locations.

Interactive Map with Folium (untrusted) Code

Interactive Map with Folium (trusted) Code

Build a Dashboard with Plotly Dash

The following components were added:

- **Dropdown:** Enables users to select a specific launch site or all launch sites (dcc.Dropdown).
- **Pie Chart**: Displays successful vs. failed launches for the selected site. Callback function (get_pie_chart) links the dropdown input to the pie chart output.
- **Range Slider:** Allows users to filter by payload range (dcc.RangeSlider).
- **Scatter Plot:** Shows correlation between payload and launch success. Callback function (get_scatter_plot) links dropdown and payload slider inputs to the scatter plot output.

Dashboard with Plotly Dash Code

Predictive Analysis (Classification)

Data Preparation

- Load the dataset
- Normalize Data
- Split data into training and test sets

Model Preparation and Evaluation

- Set parameters to GridSearchCV
- Train the model with training dataset.
- Get best parameters and accuracy on the validation data
- Compute the accuracy of the test data using the score method
- Plot the confusion matrix

Model Comparison

- Compare the models based on their accuracy scores
- Choose the model that performs the best

Models used:
- Logistic Regression
- SVM
- Decision Tree
- KNN

Predictive Analysis Code

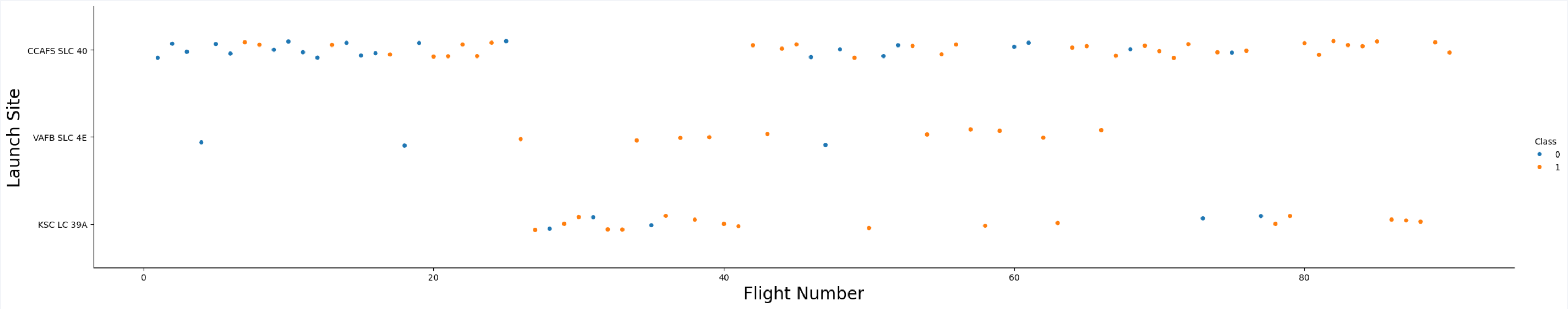
Results

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

Section 2

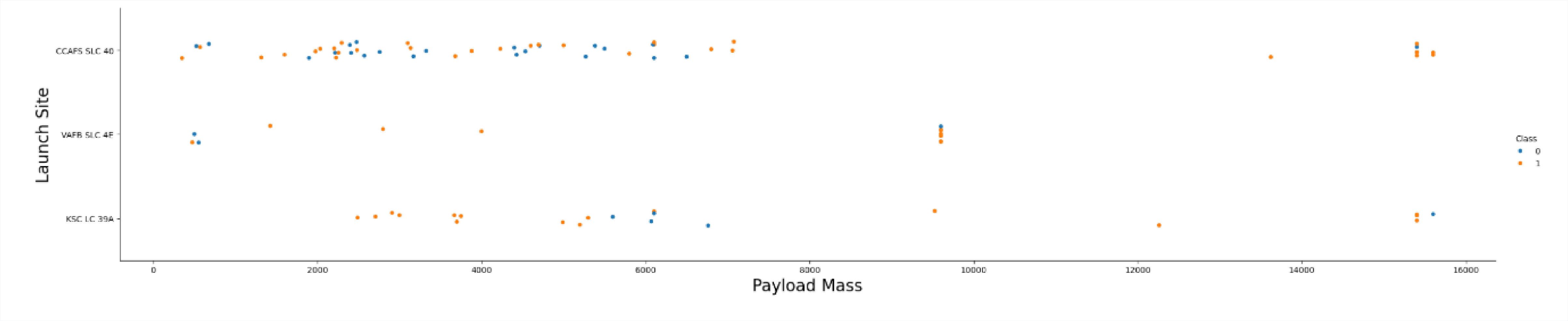
Insights drawn from EDA

Flight Number vs. Launch Site



Success rate of launches (Class = 1) varies by Launch Site. CCAFS SLC 40 and KSC LC 39A appear to have a higher frequency of successful launches compared to VAFB SLC 4E.

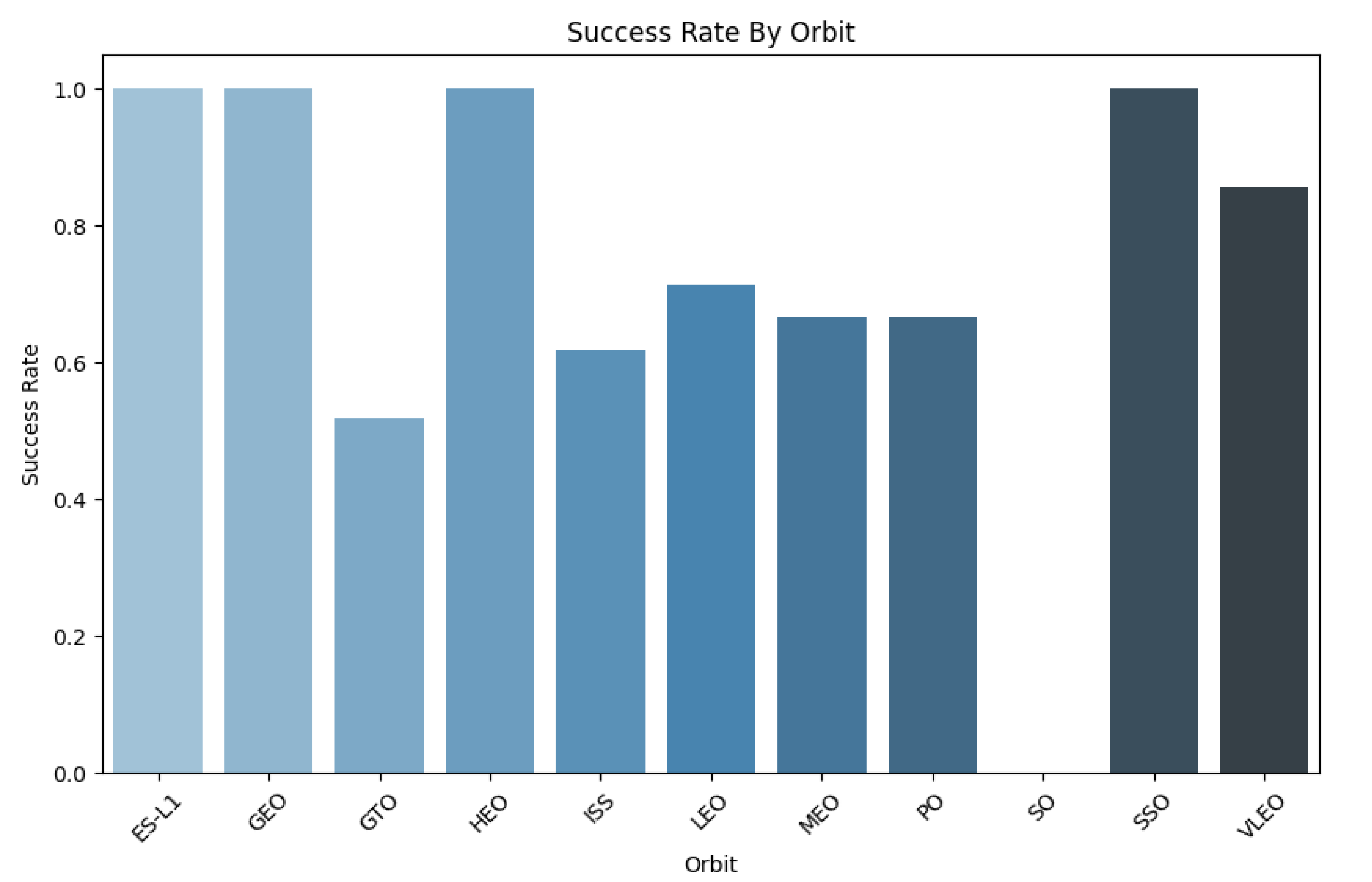
Payload vs. Launch Site



Successful launches (Class = 1) occur across various payload masses at all launch sites. However too heavy payloads can make a landing fail.

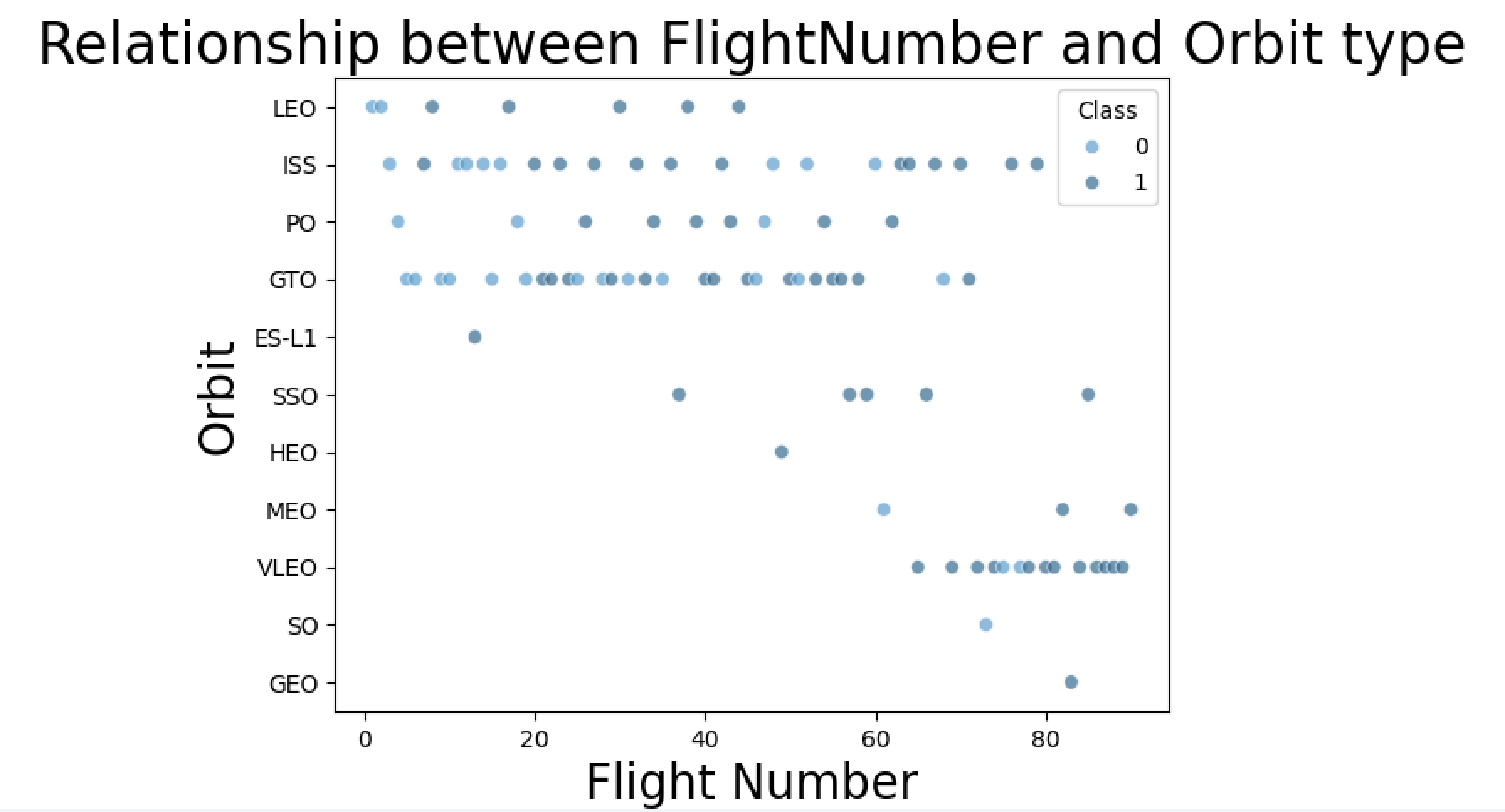
Success Rate vs. Orbit Type

Some orbits, like ES-L1, GEO, and SSO, have a near 100% success rate, while GTO shows a significantly lower success rate.



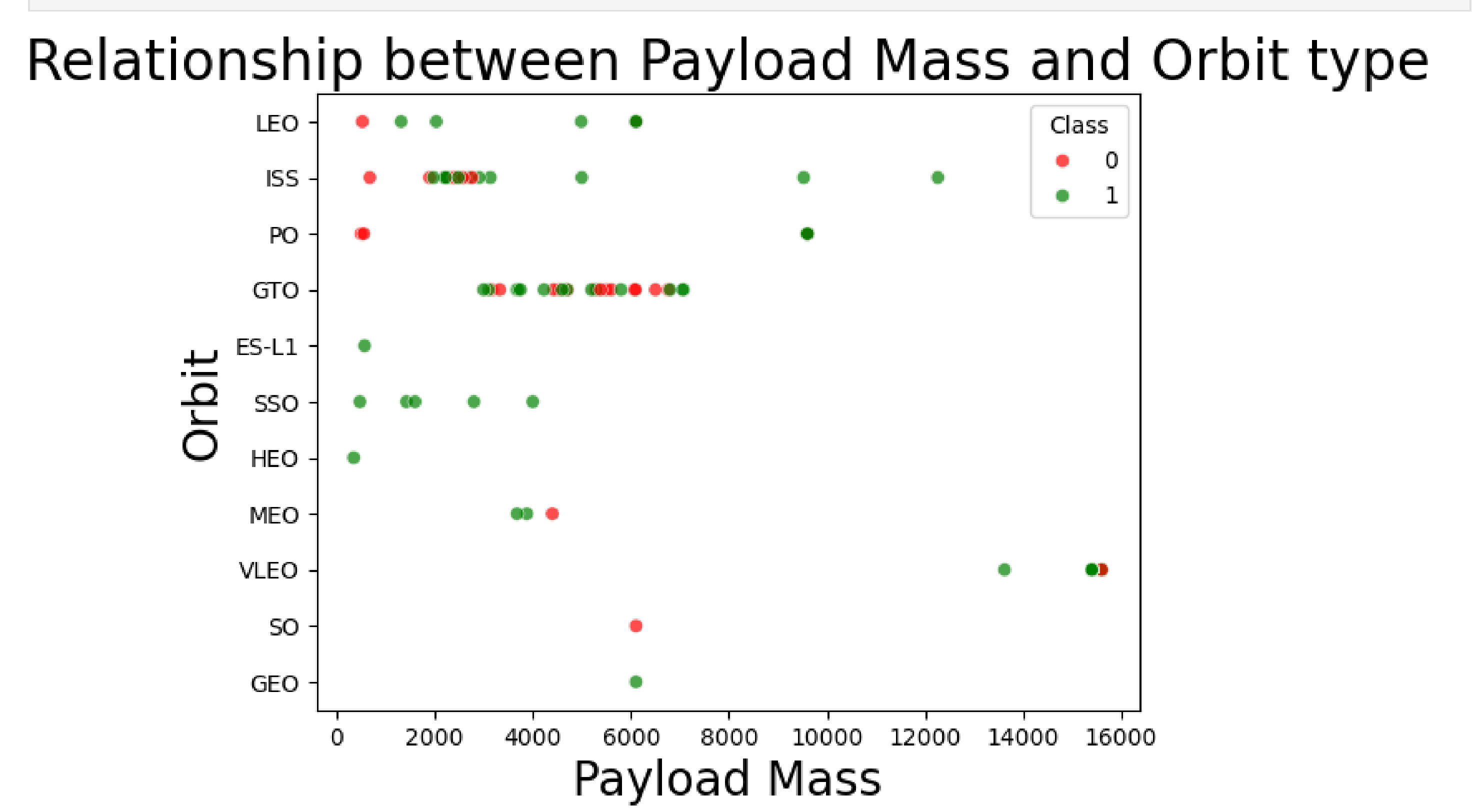
Flight Number vs. Orbit Type

As the number of flights increases, the success rate improves for LEO orbit. For GTO orbit, there appears to be no relationship between flight number and success.

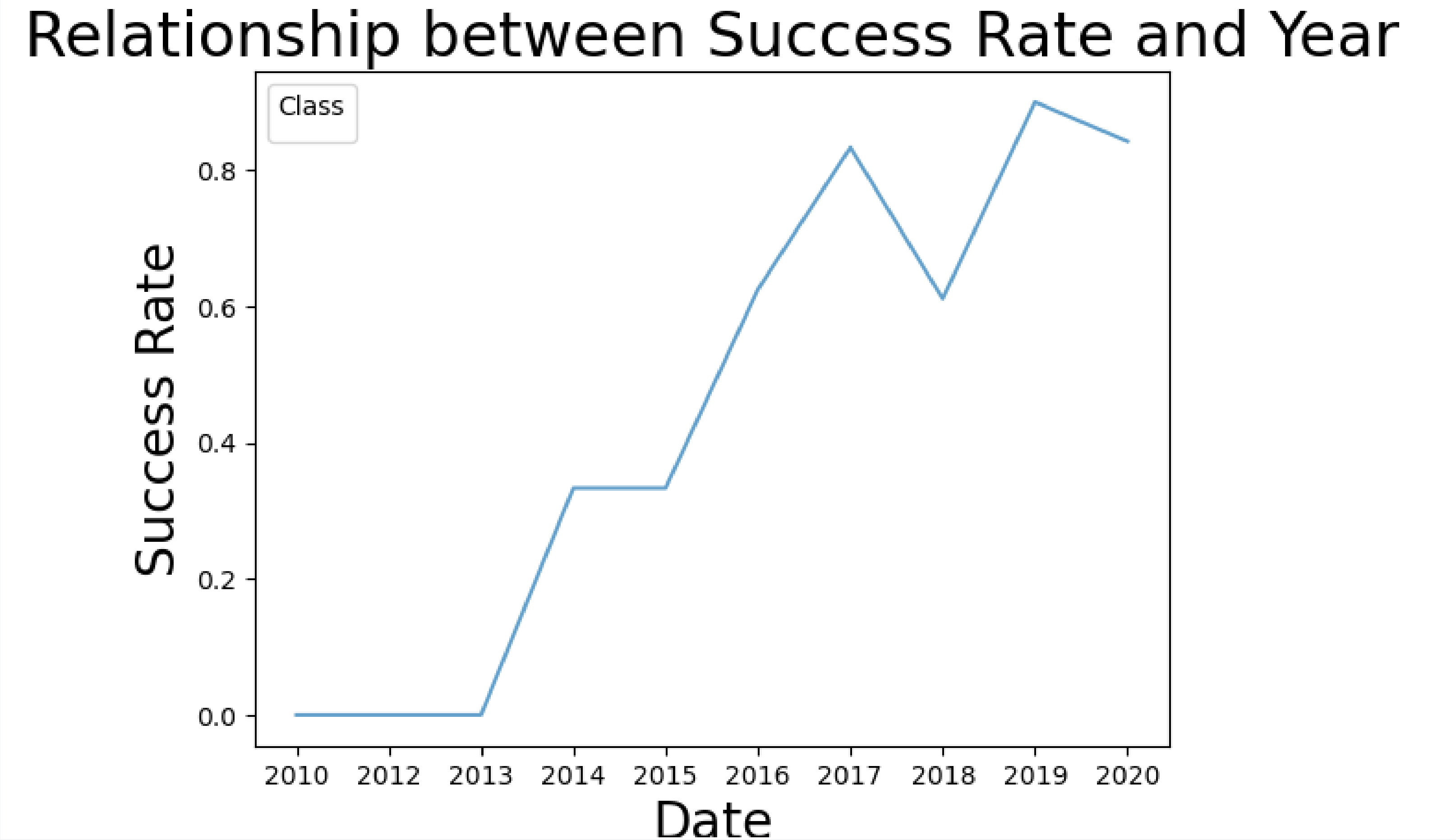


Payload vs. Orbit Type

With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



Launch Success Yearly Trend



SpaceX Rocket success rate has been increasing since 2013 till 2020.

All Launch Site Names

SQL Query	Explanation	Result
`%sql select distinct Launch_Site from SPACEXTABLE`	Distinct was used to remove any duplicates.	**Launch_Site** CCAFS LC-40 VAFB SLC-4E KSC LC-39A CCAFS SLC-40

Launch Site Names Begin with 'CCA'

SQL Query	Result

SQL Query

```
%sql select * from
SPACEXTABLE where
Launch_Site like 'CCA%'
limit 5
```

Explanation

Where clause used to select names specificcally starting with CCA.
Limit used to give only 5 records.

Result

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

SQL Query	Explanation	Result
`%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer = 'NASA (CRS)'`	Sum was used to calculate the total payload mass.	sum(PAYLOAD_MASS__KG_) 45596

Average Payload Mass by F9 v1.1

SQL Query	Explanation	Result
`%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like 'F9 v1.1%'`	AVG was used to calculate the average of the total payload mass. Like was used to obtain the result only for Booster version `F9 v1.1%.`	avg(PAYLOAD_MASS__KG_) 2534.6666666666665

First Successful Ground Landing Date

SQL Query	Explanation	Result
`%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'`	Min function was used to find the oldest date of successful landing.	**min(Date)** 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query	Explanation	Result
`%sql select Booster_Version from SPACEXTABLE where Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000`	Where and And clauses alongwith < and > give the list of booster versions where landing outcome was successful within the specified payload mass range.	**Booster_Version** F9 FT B1022 F9 FT B1026 F9 FT B1021.2 F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

<table>
<tr><th>SQL Query</th><th>Explanation</th><th>Result</th></tr>
<tr>
<td>%sql select Landing_Outcome,count(Landing_Outcome) from SPACEXTABLE where Landing_Outcome in ('Failure', 'Failure (drone ship)','Failure (parachute)','Success','Success (drone ship)','Success (ground pad)') group by Landing_Outcome</td>
<td>The IN clause was used to fetch landing outcome from a list. Group By clause was used to display the count for every outcome.</td>
<td>
<table>
<tr><th>Landing_Outcome</th><th>count(Landing_Outcome)</th></tr>
<tr><td>Failure</td><td>3</td></tr>
<tr><td>Failure (drone ship)</td><td>5</td></tr>
<tr><td>Failure (parachute)</td><td>2</td></tr>
<tr><td>Success</td><td>38</td></tr>
<tr><td>Success (drone ship)</td><td>14</td></tr>
<tr><td>Success (ground pad)</td><td>9</td></tr>
</table>
</td>
</tr>
</table>

Boosters Carried Maximum Payload

SQL Query

```
%sql select Booster_Version,PAYLOAD_MASS__KG_ from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

Explanation

Subquery was used alongwith Max function to select the maximum payload. The subquery results were used in main query to return the booster version with the heaviest payload.

Result

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

2015 Launch Records

<table>
<tr><th>SQL Query</th><th>Explanation</th><th>Result</th></tr>
<tr>
<td><code>%sql select substr(Date,6,2) as month, Landing_Outcome,Booster_Version,Launch_Site from SPACEXTABLE where Landing_Outcome in ('Failure (drone ship)') and substr(Date,1,4) = '2015'</code></td>
<td>Substr is used to fetch the month from a date. The query returns the month, landing outcome, booster version, launch site.</td>
<td>
<table>
<tr><th>month</th><th>Landing_Outcome</th><th>Booster_Version</th><th>Launch_Site</th></tr>
<tr><td>01</td><td>Failure (drone ship)</td><td>F9 v1.1 B1012</td><td>CCAFS LC-40</td></tr>
<tr><td>04</td><td>Failure (drone ship)</td><td>F9 v1.1 B1015</td><td>CCAFS LC-40</td></tr>
</table>
</td>
</tr>
</table>

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

SQL Query

```
%sql select Landing_Outcome,count(Landing_Outcome) as Outcome_count from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by Outcome_count DESC
```

Explanation

Count is used to select the count of landing outcomes between specified dates.
Group By shows results by landing outcome.
Order By Count Desc shows the results in the descending order.

Result

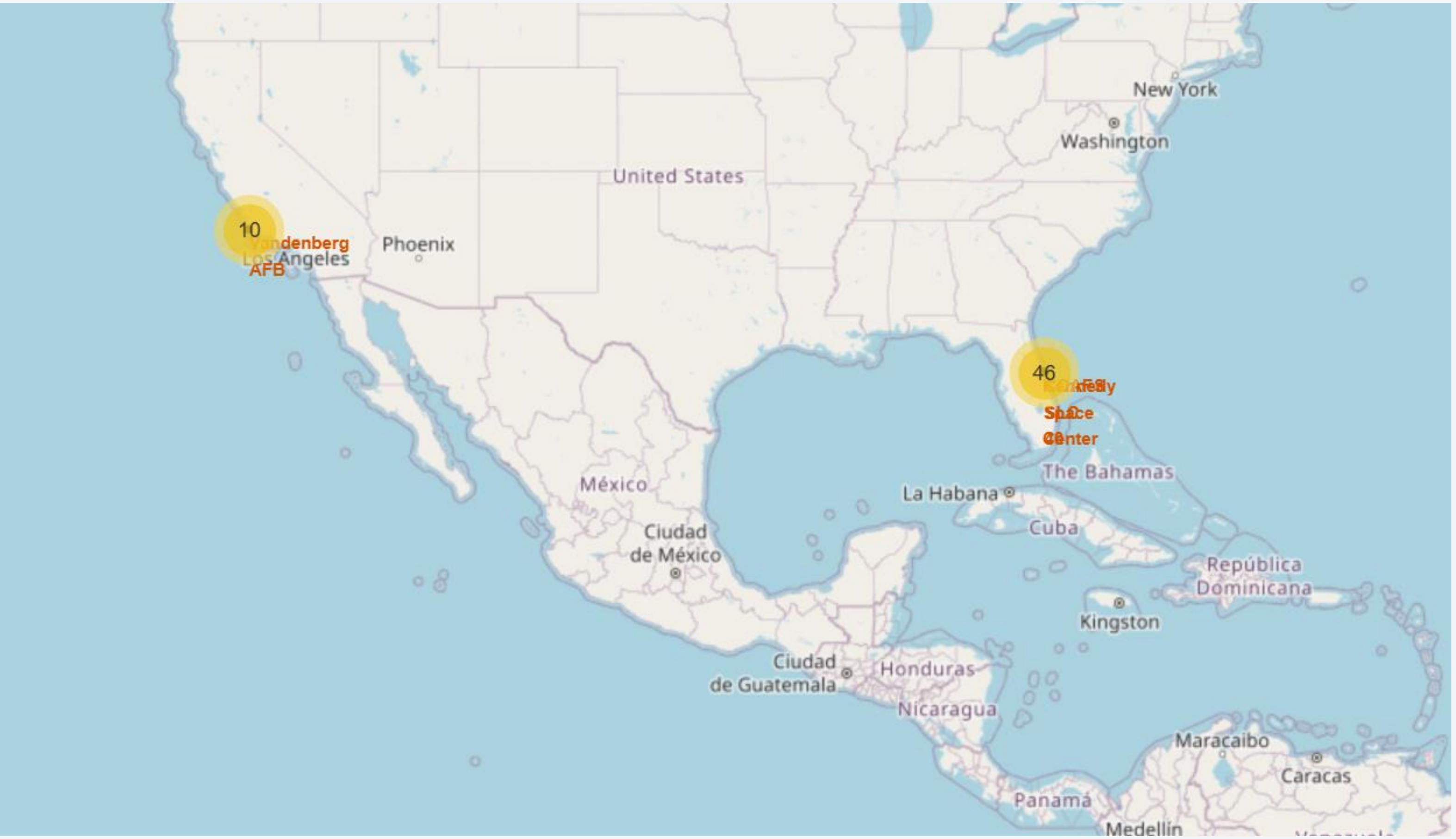
Landing_Outcome	Outcome_count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

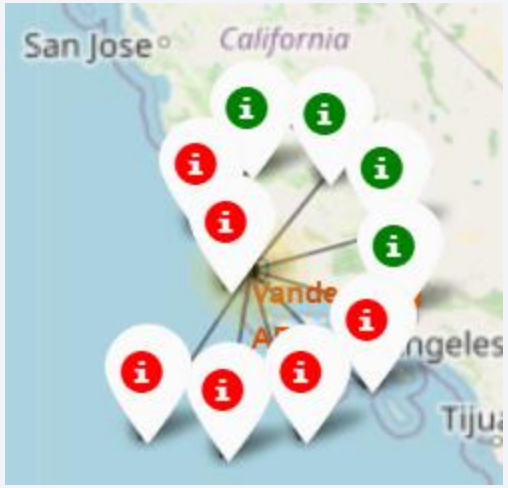
Launch Sites Proximities Analysis

Folium Map: All Launch Sites

SpaceX launch sites are located on the eastern and western coasts of the united States.

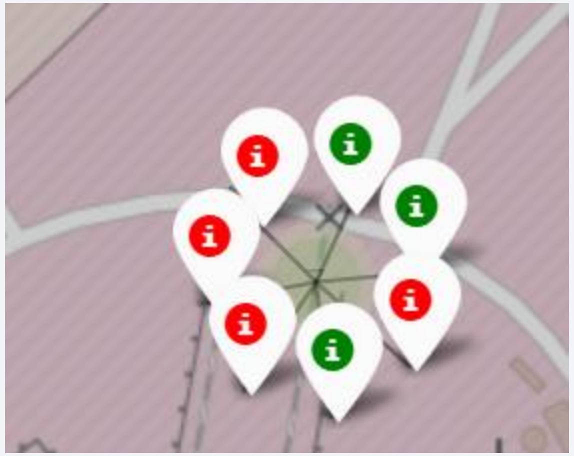


Folium Map: Color Labeled Launch Outcomes

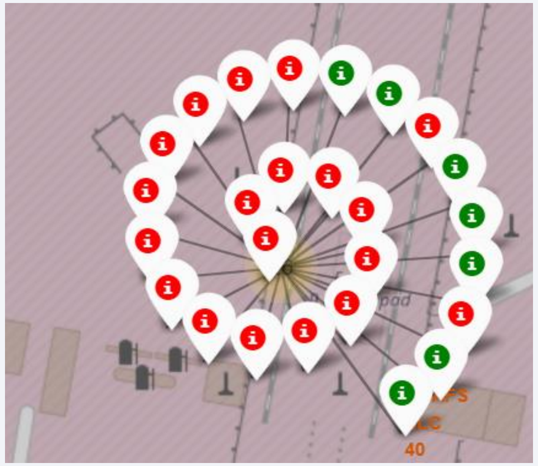


VAFB SLC-4E

KSC LC-39A



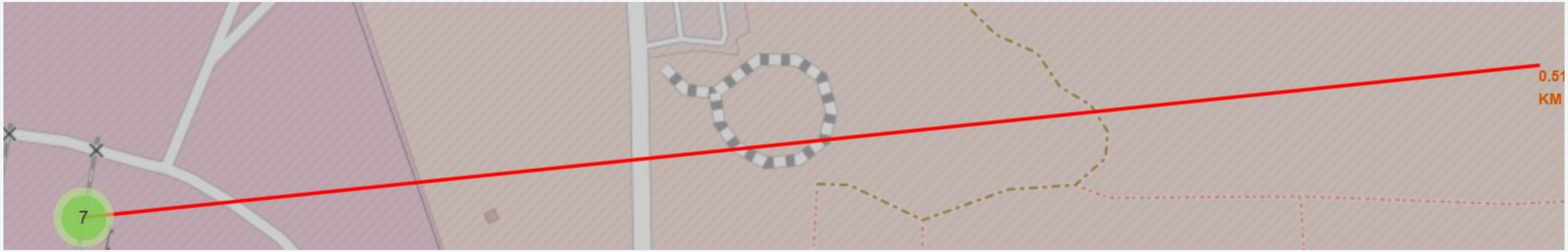
CCAFS SLC-40



CCAFS LC-40

- Screenshots show the launch outcomes for the different launch sites.
- Green indicates the number of successful launches
- Red indicates the number of failed launches.
- KSC LC-39A have the maximum number of successful launches

Folium Map: Distances between Launch Sites and Proximities



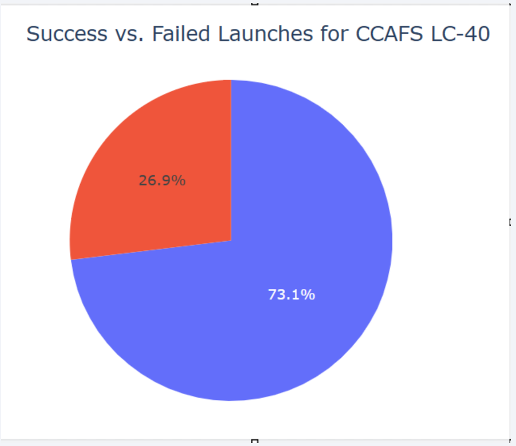
Distance between CCAFS SLC-40 and costline is 0.5 km

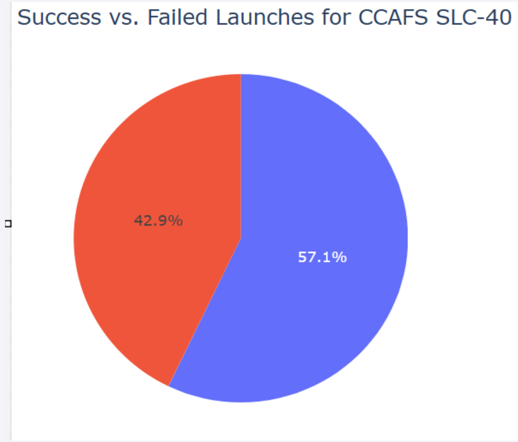
Distance between CCAFS SLC-40 and NASA Causeway is 19.24 km

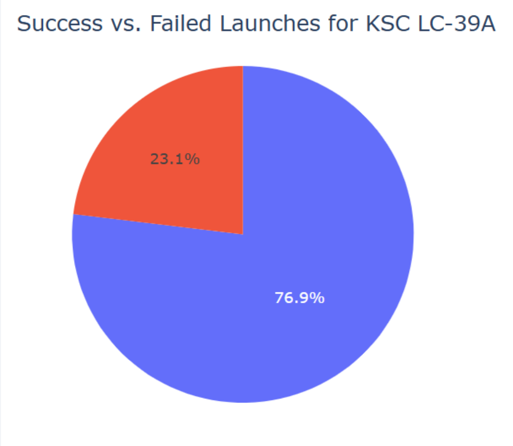
Section 4

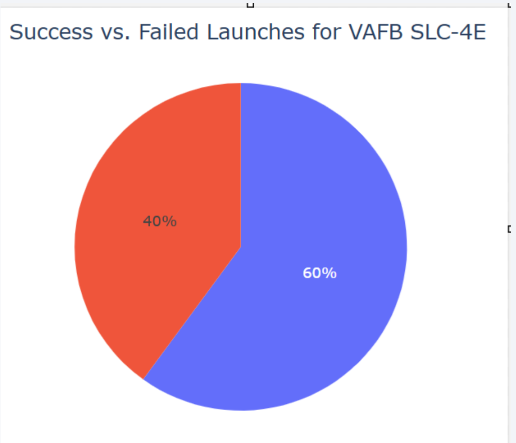
Build a Dashboard with Plotly Dash

Dashboard Screenshot: Total Success By Site



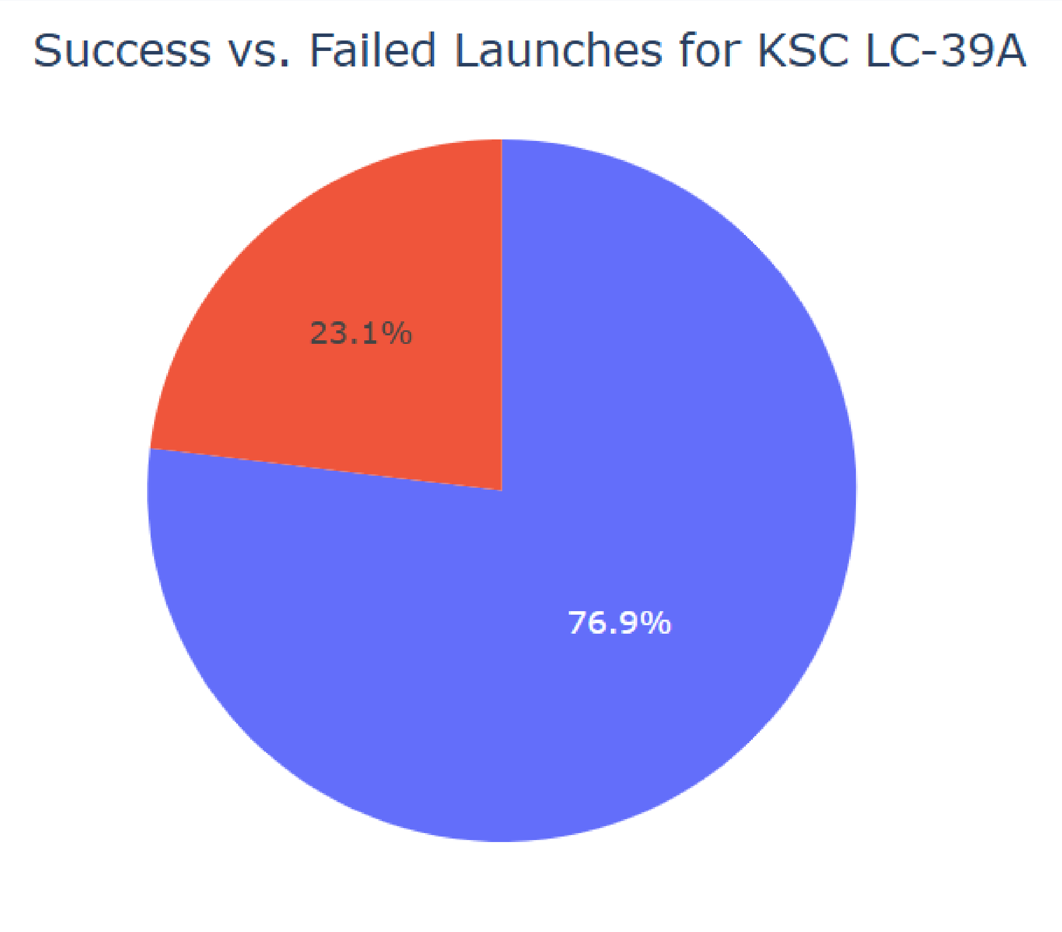






KSLC-39A has the highest success rate of all launches.

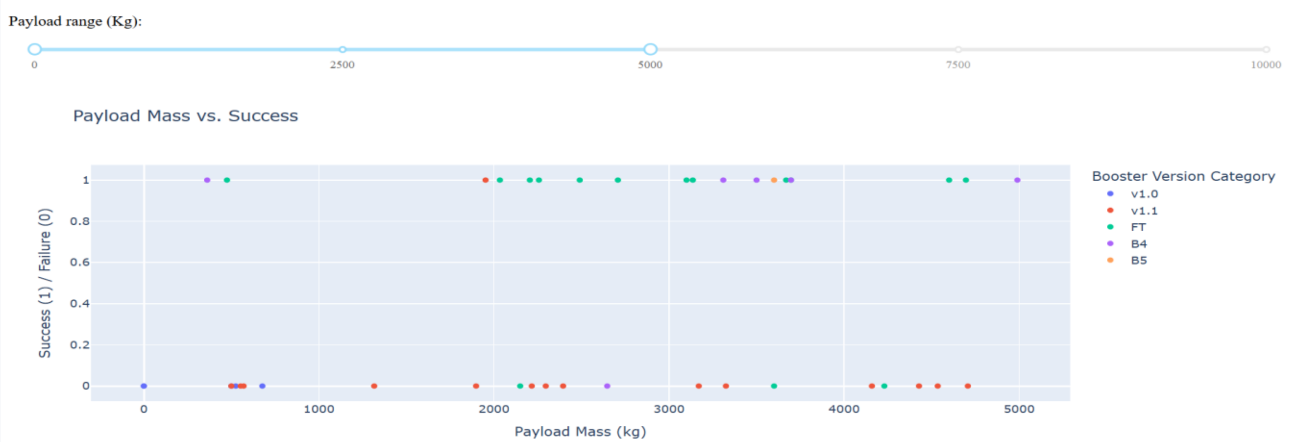
Dashboard Screenshot: Launch Site with Highest Success Rate



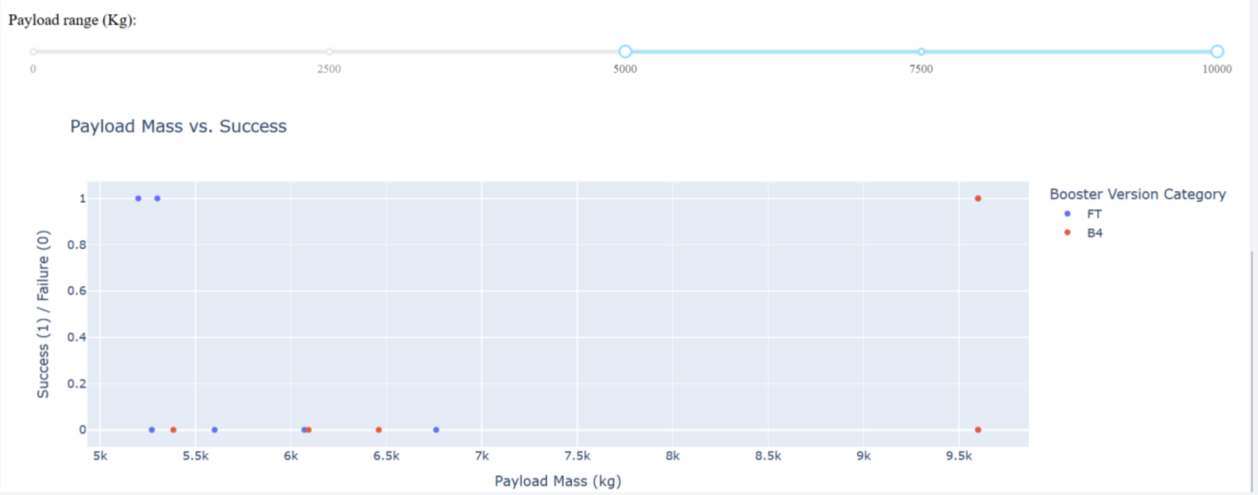
KSLC-39A has the highest success rate of 76.9% of all launches.

Dashboard Screenshot: Payload Mass vs Success Rate

Payload Mass between 0 and 5000 kg



Payload Mass between 5000 and 10000 kg



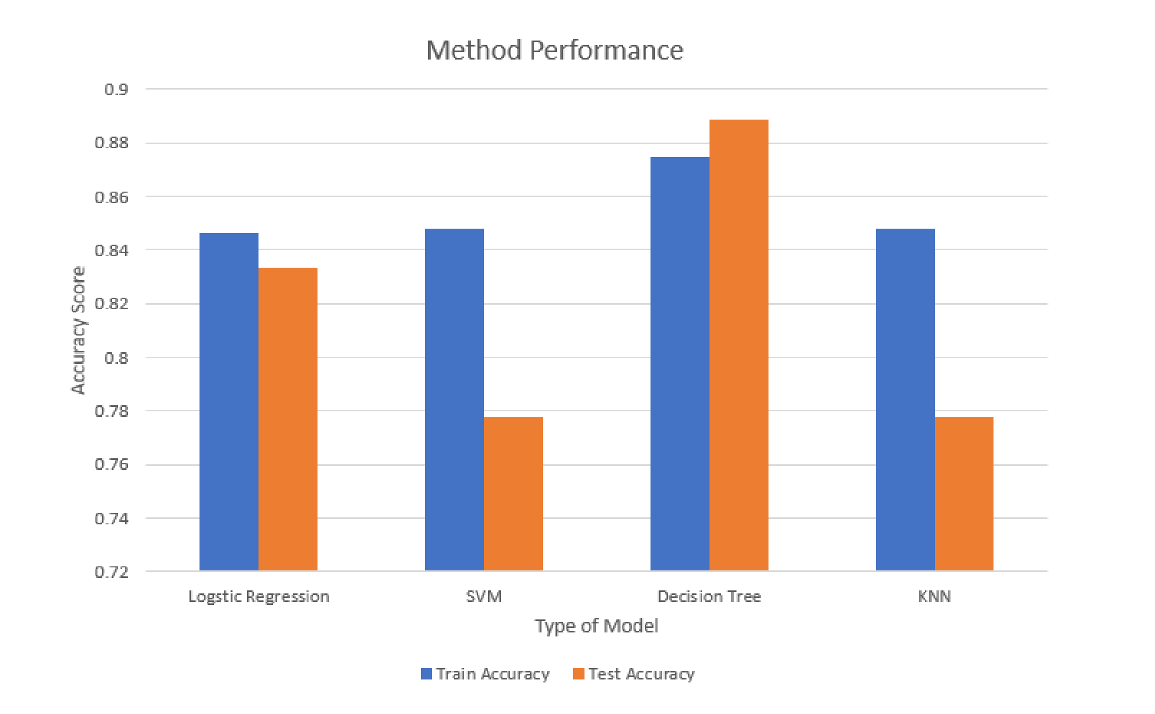
Low weighted payloads have a better success rate.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

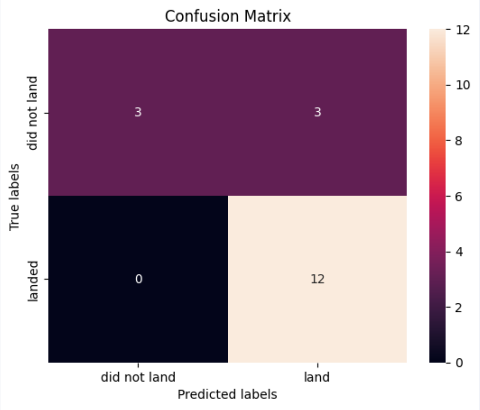
- The accuracy scores are very close for every model.
- However, for now, decision tree performs the best.



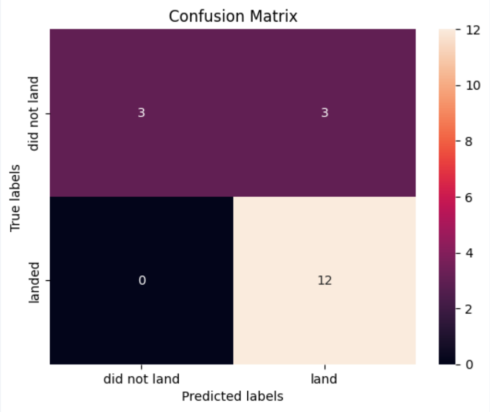
Decision Tree Hyperparameters

```
tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_leaf':
1, 'min_samples_split': 5, 'splitter': 'best'}
accuracy : 0.875
```

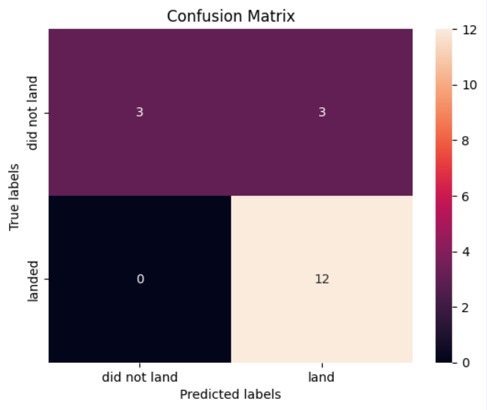
Confusion Matrix



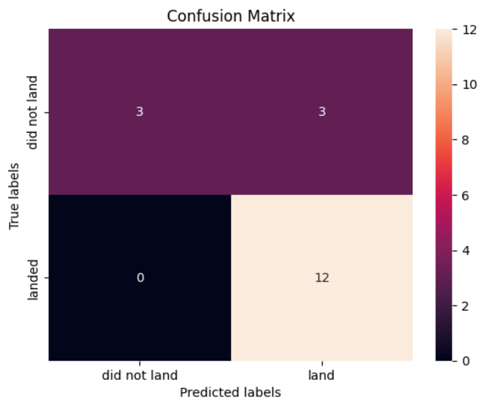
Logistic Regression



SVM



Decision Tree



KNN

Confusion Matrix for each model are quite similar as test accuracy scores are almost equal.

Conclusions

- Mission success depends on factors like launch site, orbit, and the number of previous launches, with experience improving outcomes over time.
- Orbits with the highest success rates are GEO, HEO, SSO, and ES L1.
- Payload mass impacts mission success, with lighter payloads generally performing better,specific orbits may require lighter or heavier payloads.
- The reason for differences in launch site performance (e.g., KSC LC 39A being the best) remains unclear. More data needed for better analysis.
- The Decision Tree Algorithm is chosen as the best model due to its higher training accuracy, despite identical test accuracy across models.

Thank you!