Coursera Applied Data Science Capstone

Data-Driven Insights

Created By Tahereh Nayerifard

Executive Summary

<u>Summary</u>

- Data collection
- Data wrangling
- Exploratory Data Analysis with Data Visualization
- Exploratory Data Analysis with SQL
- Building an interactive map with Folium
- Building a Dashboard with Plotly Dash
- Predictive analysis (Classification)

Introduction

- SpaceX, leading the commercial space industry, has revolutionized space travel by offering cost-effective solutions. The company prominently features Falcon 9 rocket launches on its website at a mere $62 million per launch, a stark contrast to competitors whose launches can cost over $165 million each. The significant savings SpaceX achieves stem largely from its groundbreaking reusability of the first stage. Thus, the ability to predict the successful landing of this crucial stage directly impacts the overall launch cost.

- By leveraging both public data and sophisticated machine learning models, our objective is to forecast whether SpaceX will opt to reuse the first stage. In this pursuit, we aim to address several critical questions:

  - Impact of Variables: How do factors like payload mass, launch site, number of flights, and orbits influence the success rate of first stage landings?

  - Trend Analysis: Does the frequency of successful landings exhibit an upward trajectory over the years?

  - Optimal Algorithm: Which binary classification algorithm stands out as the most effective for this specific predictive task?

- Through rigorous analysis and model deployment, we endeavor to unravel the intricate interplay of these variables and pave the way for more accurate predictions regarding the reusability of SpaceX's first stage.

Data Collection Methodology (SpaceX API)

Overall:

- ✓ Using SpaceX Rest API
- ✓ Using Web Scrapping from Wikipedia

Steps:

1. Fetching rocket launch data from the SpaceX API
2. Parsing the response content using .json() and converting it into a dataframe with .json_normalize()
3. Extracting essential launch details from the SpaceX API by employing custom functions
4. Organizing the acquired data into a structured dictionary
5. Transforming the dictionary into a dataframe
6. Filtering the dataframe to retain records corresponding to Falcon 9 launches
7. Imputing missing values in the Payload Mass column with the calculated mean using .mean() for this specific column
8. Saving the processed data to a CSV file

Data Collection Methodology (Web scraping)

Overall:

- ✓ Filtering the data
- ✓ Dealing with missing values
- ✓ Using One Hot Encoding to prepare the data to a binary classification

Steps:

1. Retrieving Falcon 9 launch data from Wikipedia
2. Generating a BeautifulSoup object from the HTML response
3. Capturing all column names from the HTML table header
4. Parsing HTML tables to gather the required data
5. Structuring the acquired data into a dictionary
6. Converting the dictionary into a data frame
7. Saving the data to a CSV file

Data wrangling Methodology

1. Conducting Exploratory Data Analysis to establish Training Labels
2. Computing the count of launches at each site
3. Determining the quantity and frequency of each orbit
4. Analyzing the number and frequency of mission outcomes based on orbit type
5. Generating a landing outcome label derived from the Outcome column
6. Saving the processed data to a CSV file

EDA with data visualization Methodology

- Exploratory Data Analysis (EDA) was conducted with data visualization, where the following charts were generated:
- Scatter plots were plotted for Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit Type vs. Success Rate, Flight Number vs. Orbit Type, Payload Mass vs. Orbit Type, and the Yearly Trend in Success Rate.
- Scatter plots illustrate the relationships between variables. If discernible patterns or correlations exist, they could potentially be utilized in a machine learning model.
- Bar charts were used to compare data among discrete categories. The objective was to highlight the relationships between specific categories under comparison and their corresponding measured values.
- Line charts were employed to depict trends in the data over time, particularly focusing on time series analysis.

EDA with SQL Methodology

Executed SQL queries:

- Displayed the names of the unique launch sites involved in the space missions.
- Retrieved 5 records where the launch sites start with the string 'CCA'.
- Calculated the total payload mass carried by boosters launched by NASA (CRS).
- Determined the average payload mass carried by booster version F9 v1.1.
- Listed the date when the first successful landing outcome on a ground pad was achieved.
- Identified the names of boosters that successfully landed on a drone ship, carrying a payload mass between 4000 and 6000.
- Provided the total count of successful and failed mission outcomes.
- Listed the booster versions that carried the maximum payload mass.
- Identified the failed landing outcomes on drone ships, along with their booster versions and launch site names, specifically for the months in the year 2015.
- Ranked the count of landing outcomes (e.g., Failure (drone ship) or Success (ground pad)) between the dates 2010-06-04 and 2017-03-20 in descending order.

Interactive Visual Analytics with Folium

To create an interactive map using Folium, follow these steps:

- ✓ Markers for All Launch Sites:
  - Added a marker with a circle, popup label, and text label for NASA Johnson Space Center using its latitude and longitude coordinates as the start location.
  - Added markers with circles, popup labels, and text labels for all launch sites, displaying their geographical locations and proximity to the Equator and coastlines.
- ✓ Colored Markers for Launch Outcomes:
  - Incorporated colored markers for success (Green) and failed (Red) launches using Marker Cluster to highlight launch sites with high success rates.
- ✓ Distances between Launch Sites and Proximities:
  - Included colored lines to illustrate distances between the launch site KSC LC-39A (as an example) and its proximities like railways, highways, coastlines, and the closest city.
  - This setup will provide an engaging and informative map displaying essential information about launch sites, outcomes, and proximity details.

SpaceX Dash App

To create a dashboard using Plotly Dash, follow these steps:

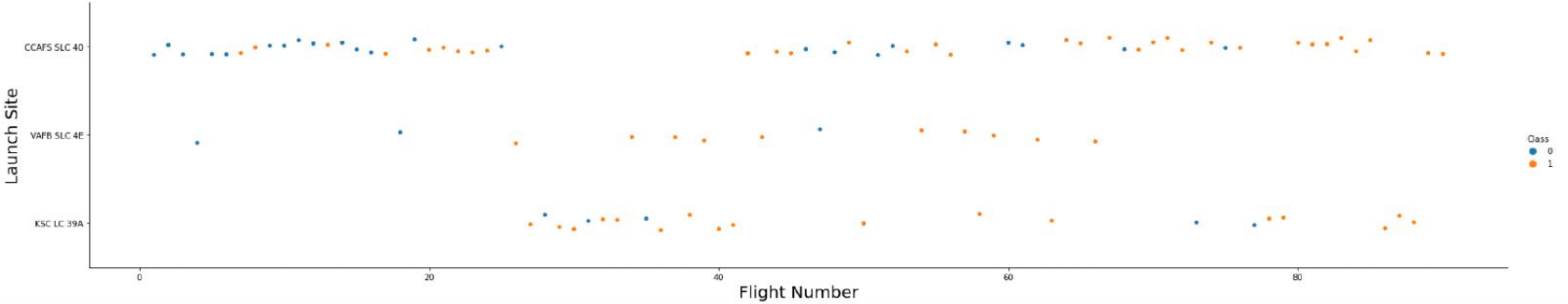
- ✓ Launch Sites Dropdown List:
  - Integrated a dropdown list to facilitate Launch Site selection.
- ✓ Pie Chart Displaying Success Launches:
  - Included a pie chart to present the total count of successful launches across all sites and to compare Success vs. Failed counts for a specific Launch Site, if selected.
- ✓ Slider for Payload Mass Range:
  - Implemented a slider to choose the Payload range.
- ✓ Scatter Chart of Payload Mass vs. Success Rate by Booster Versions:
  - Added a scatter chart to illustrate the relationship between Payload and Launch Success for various Booster Versions.

Predictive analysis (Classification)

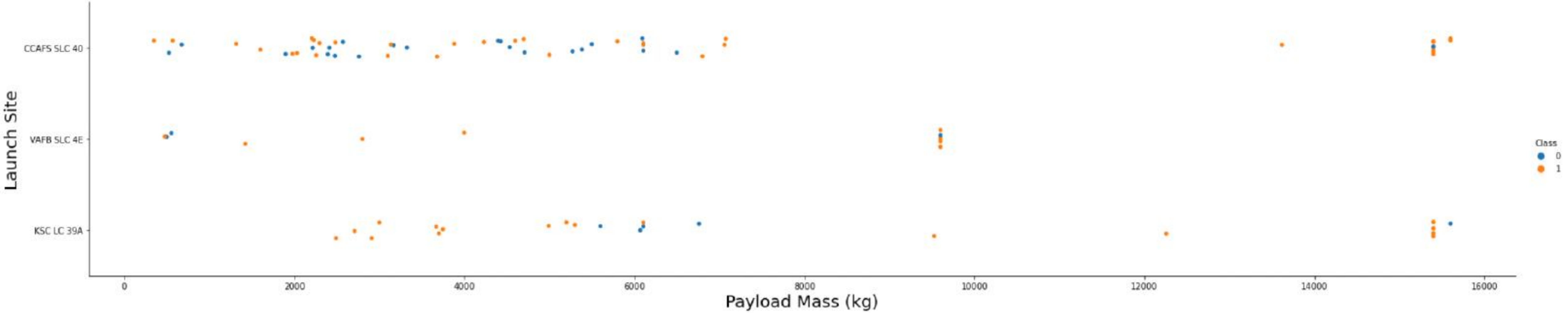
Performing the following tasks in a machine learning workflow:

- Extracting a NumPy array from the column "Class" in the dataset
- Standardizing the data using StandardScaler, followed by fitting and transforming it
- Dividing the data into training and testing sets using the train_test_split function
- Setting up a GridSearchCV object with cross-validation folds (cv = 10) to determine optimal parameters
- Evaluating the best-performing method by analyzing Jaccard score and F1 score metrics
- Reviewing the confusion matrix for all models
- Calculating the accuracy on the test data using the .score() method for each model
- Applying GridSearchCV on Logistic Regression, SVM, Decision Tree, and KNN models

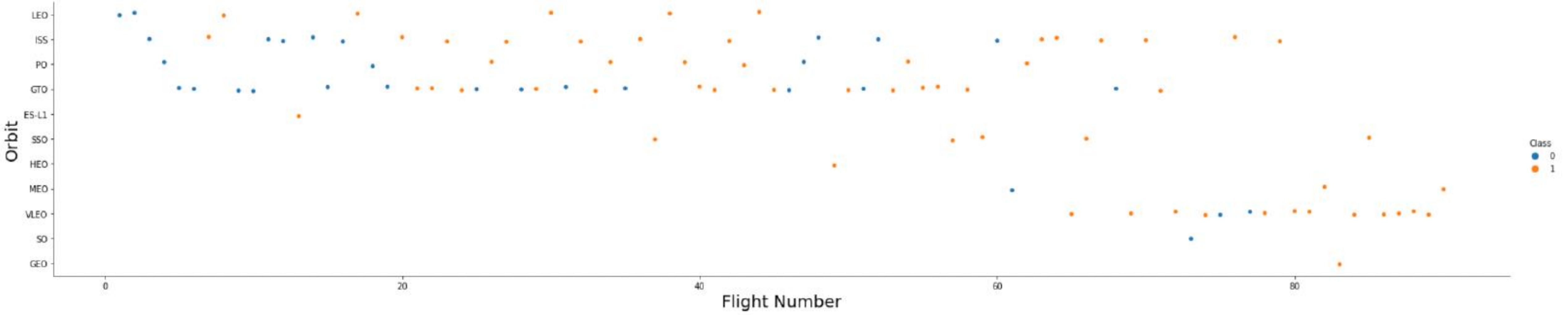
EDA with visualization results
(Flight Number vs. Launch Site)



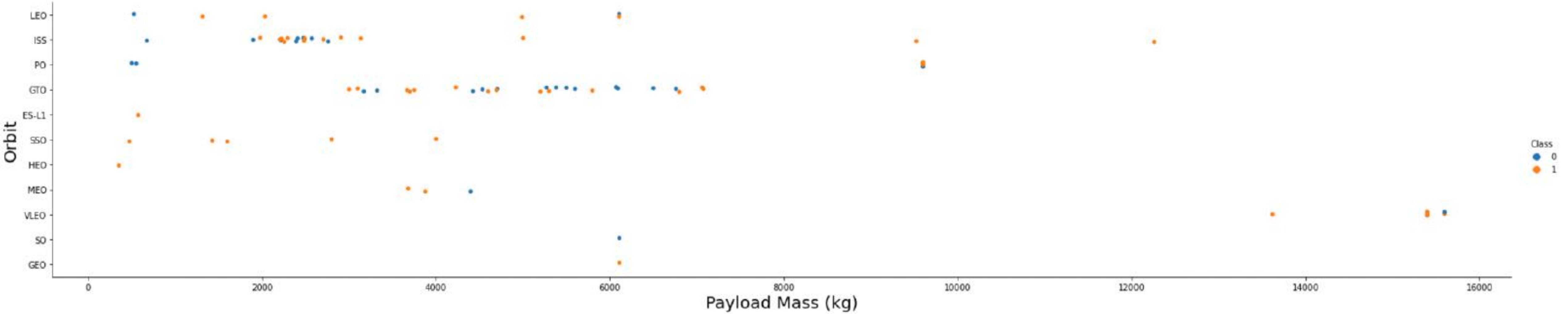
(Payload vs. Launch Site)



EDA with visualization results
(Flight Number vs. Orbit type)

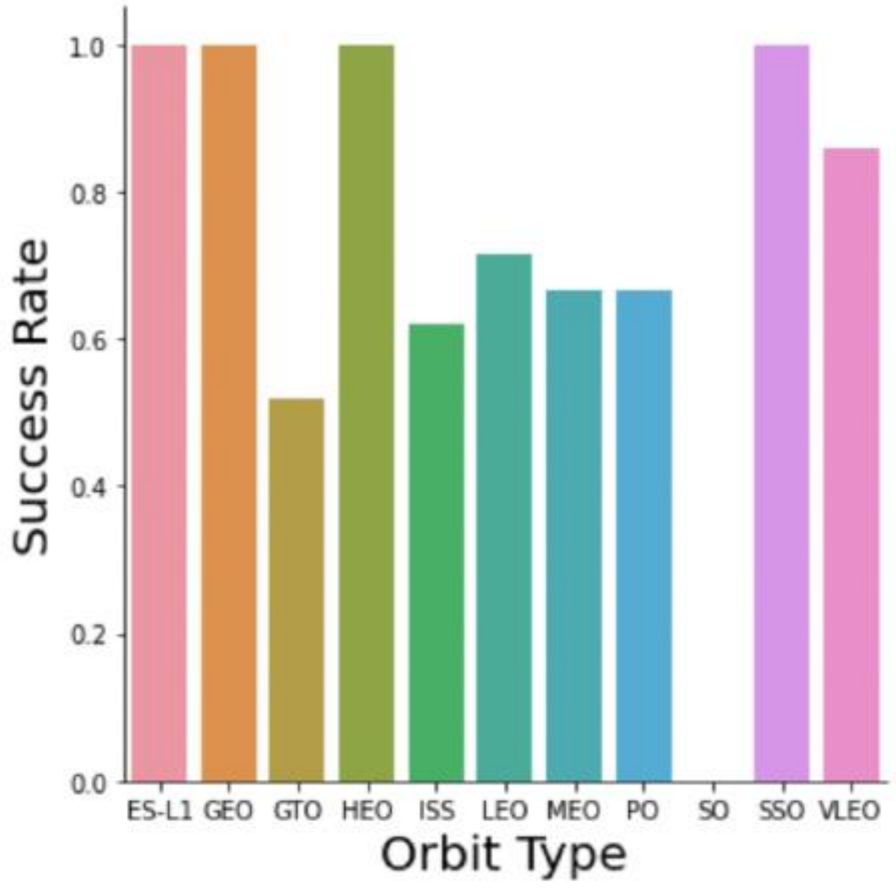


(Payload Mass vs. Orbit type)

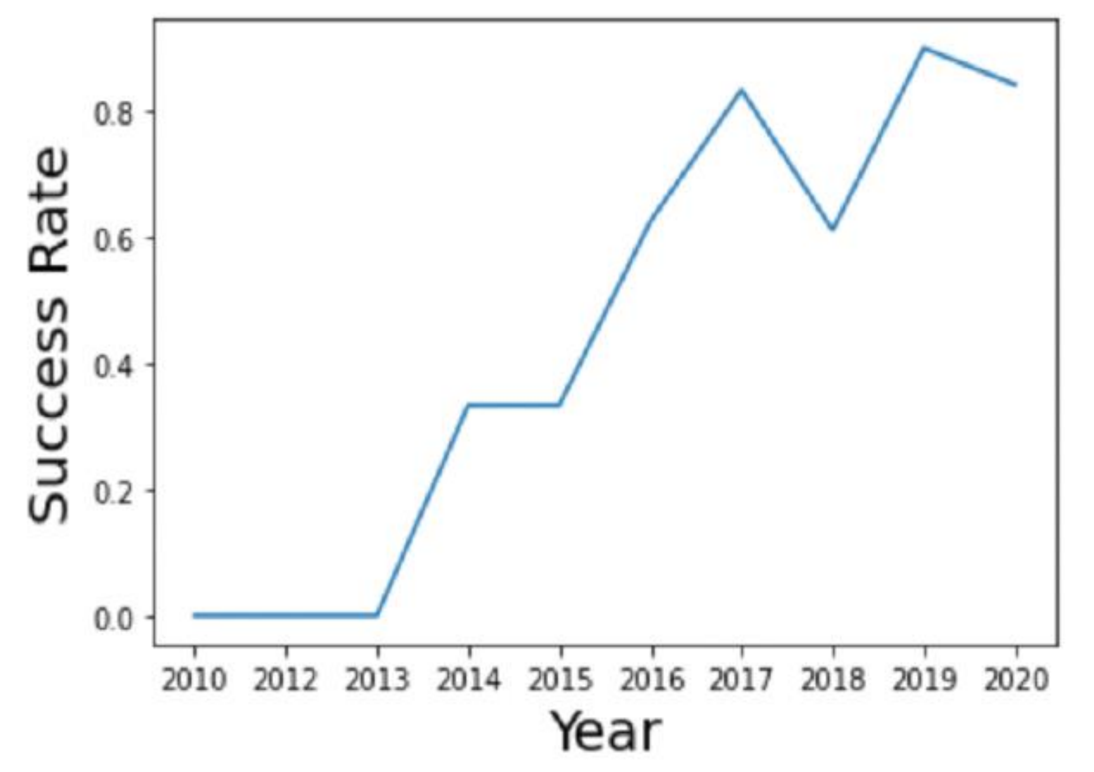


EDA with visualization results

(Success rate vs. Orbit type)



(Launch success yearly trend)



Created By TN

EDA with SQL results
(All launch site names)

```
In [4]: %sql select distinct launch_site from SPACEXDATASET;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[4]:

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

EDA with SQL results
(Names Begin with `CCA`)

```
In [5]: %sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

EDA with SQL results
(Total payload mass)

```
In [6]: %sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXDATASET where customer = 'NASA (CRS)';
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[6]:
```

total_payload_mass
45596

(Average payload mass by F9 v1.1)

```
In [7]: %sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXDATASET where booster_version like '%F9 v1.1%';
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[7]:
```

average_payload_mass
2534

EDA with SQL results
(Date of the initial successful ground landing)

```
In [8]: %sql select min(date) as first_successful_landing from SPACEXDATASET where landing__outcome = 'Success (ground pad)';

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[8]:
```

first_successful_landing
2015-12-22

(Successful landing on a drone ship with a payload ranging from 4000 to 6000)

```
In [9]: %sql select booster_version from SPACEXDATASET where landing__outcome = 'Success (drone ship)' and payload_mass__kg_ between 4
000 and 6000;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[9]:
```

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

EDA with SQL results

(Total count of successful and failed mission outcomes)

```
In [10]: %sql select mission_outcome, count(*) as total_number from SPACEXDATASET group by mission_outcome;
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[10]:
```

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

(Boosters carried maximum payload)

```
In [11]: %sql select booster_version from SPACEXDATASET where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXDATASET);
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[11]:
```

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

EDA with SQL results
(2015 launch records)

```
In [12]: %%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXDATASET
         where landing__outcome = 'Failure (drone ship)' and year(date)=2015;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[12]:
```

MONTH	DATE	booster_version	launch_site	landing__outcome
January	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

(Rank success count: 2010-06-04 until 2017-03-20)

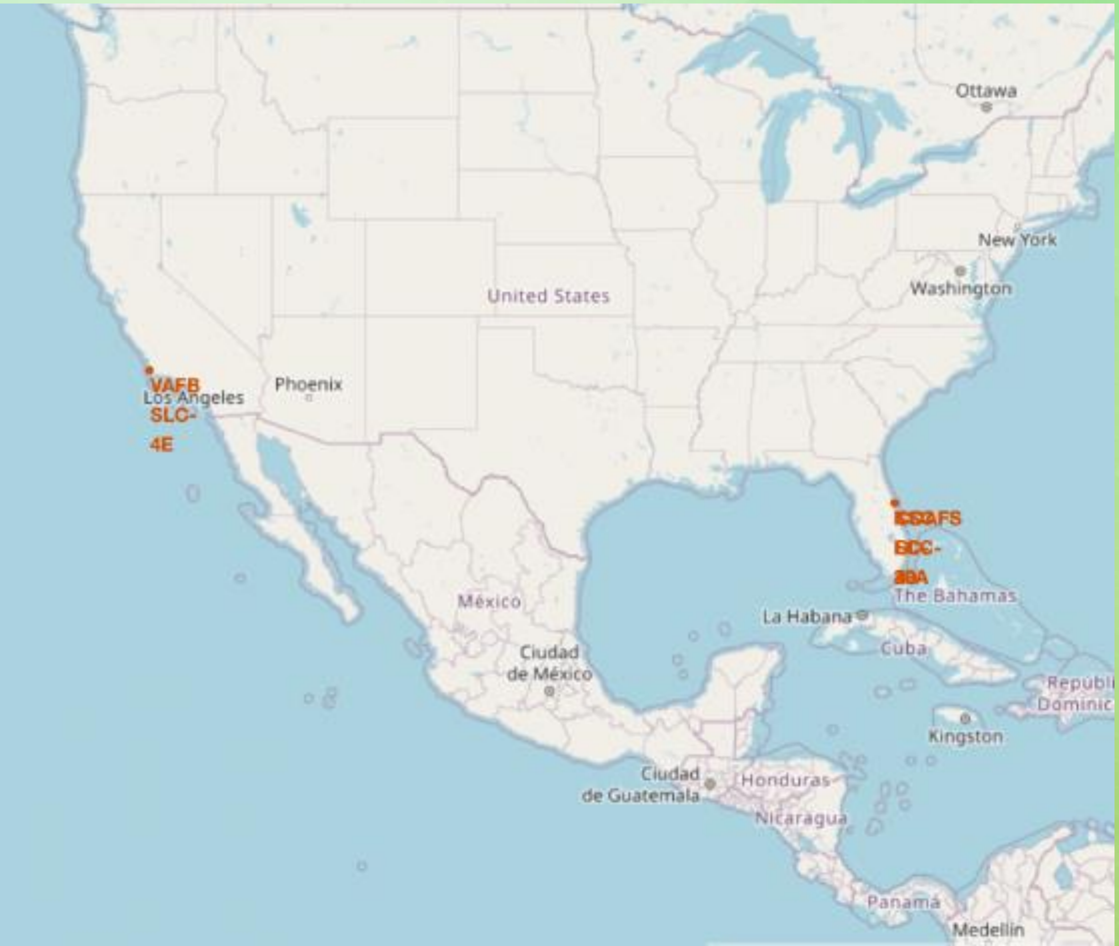
```
In [13]: %%sql select landing__outcome, count(*) as count_outcomes from SPACEXDATASET
         where date between '2010-06-04' and '2017-03-20'
         group by landing__outcome
         order by count_outcomes desc;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[13]:
```

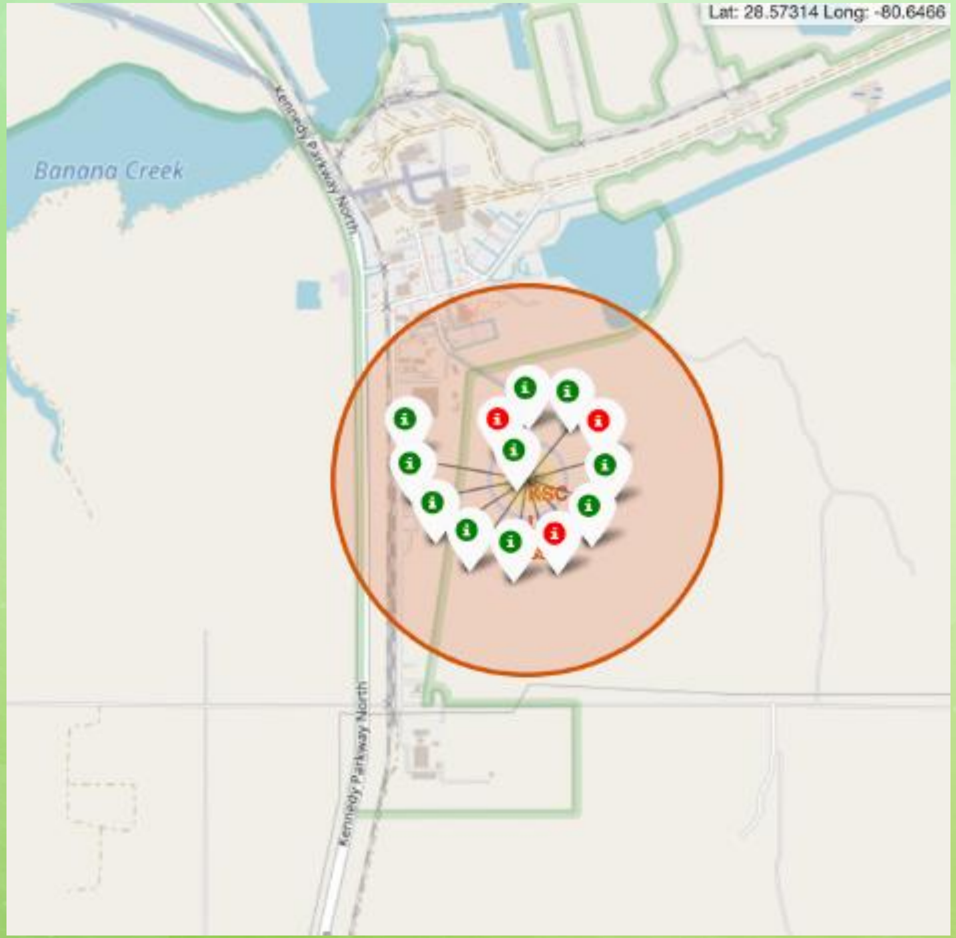
landing__outcome	count_outcomes
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Interactive map with Folium

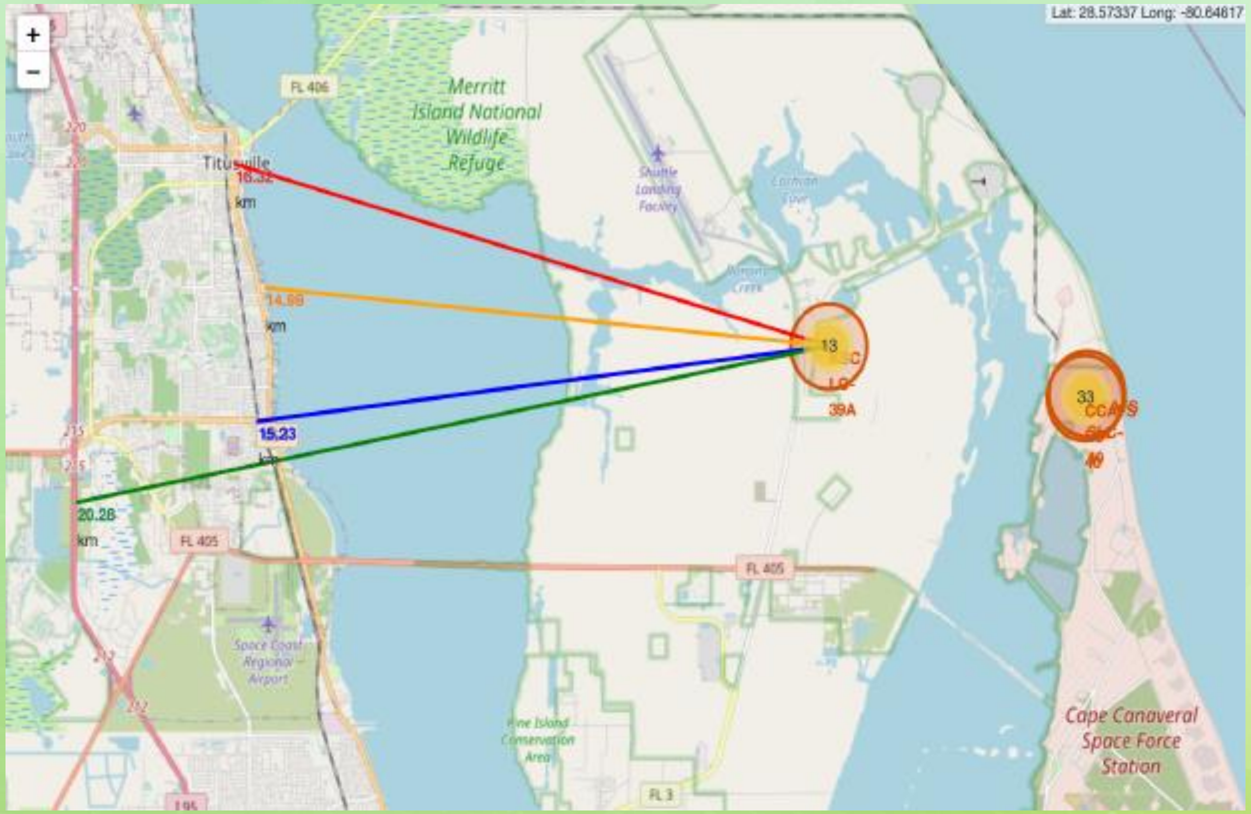
Launch sites' location markers on a global map



Color-labeled launch records on the map

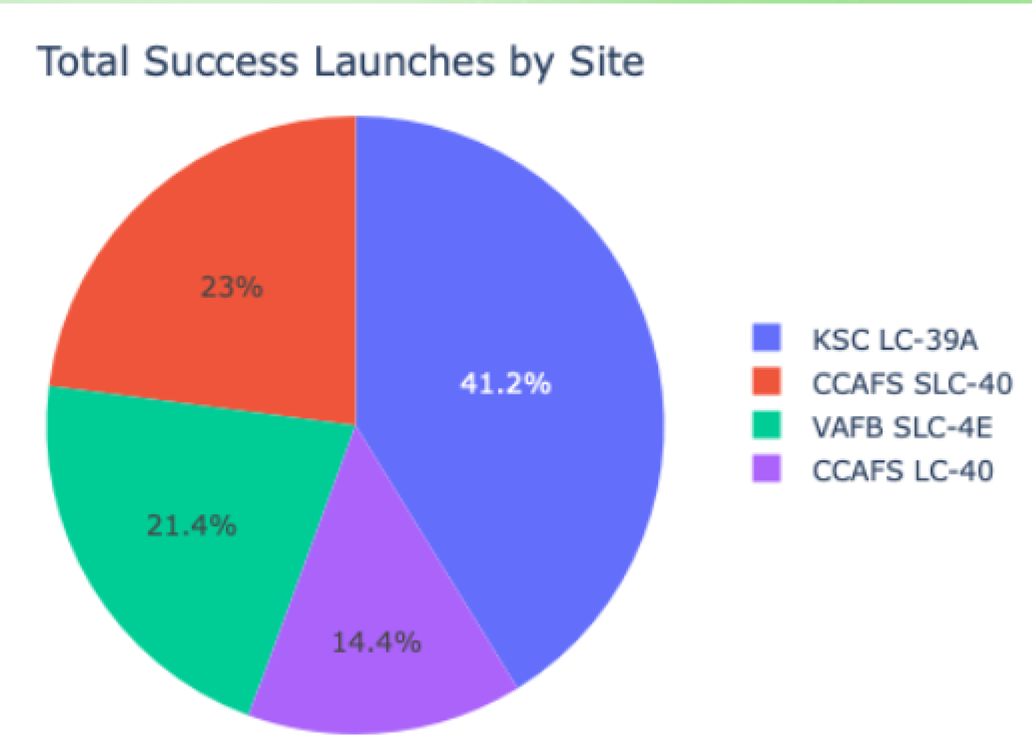


Interactive map with Folium
(Distance from the launch site KSC LC-39A to its proximities)



Build a Dashboard with Plotly Dash

Launch success count for all sites

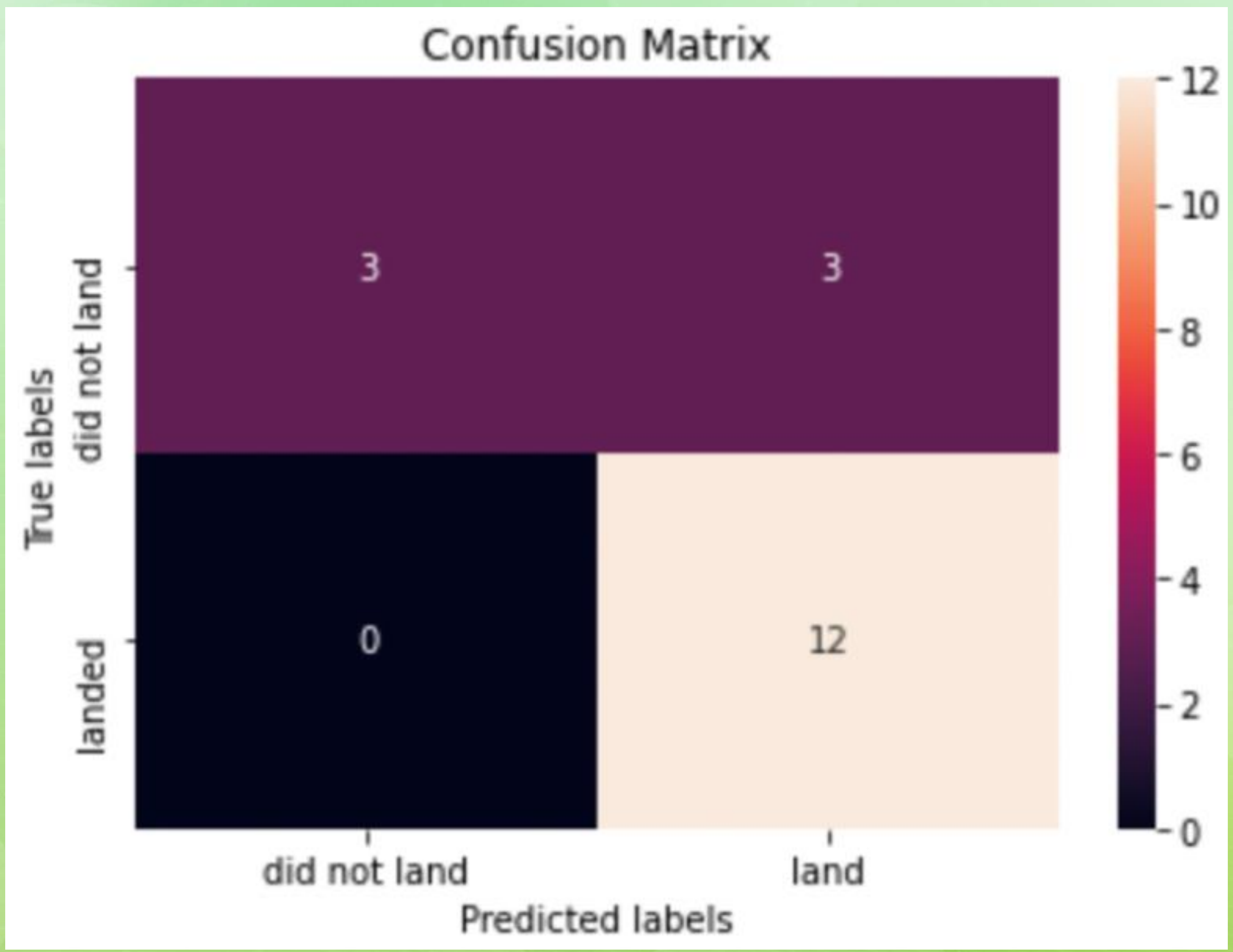


Launch site with highest launch success ratio

Build a Dashboard with Plotly Dash
(Payload Mass vs. Launch Outcome for all sites)

Build a Dashboard with Plotly Dash
(Payload Mass vs. Launch Outcome for all sites)

Predictive analysis (Classification)
(Confusion Matrix)



Conclusion

- ✓ Missions with lower payload masses demonstrate superior performance compared to those with heavier payloads.
- ✓ The majority of launch sites are situated near the Equator line, with all sites being in close proximity to coastlines.
- ✓ The Decision Tree Model stands out as the optimal algorithm for this dataset.
- ✓ Over time, the success rate of launches exhibits an upward trend.
- ✓ KSC LC-39A boasts the highest success rate among all launch sites.
- ✓ Orbits ES-L1, GEO, HEO, and SSO achieve a flawless 100% success rate.