IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Muhammad Shahnoor
February 17, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection with Web Scraping and API
  - Data Wrangling and Cleaning
  - Exploratory Data Analysis with SQL and Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning
- Summary of all results
  - EDA (Exploratory Data Analysis)
  - Interactive analytics
  - Predictive Analytics

Introduction

- Project background and context

  Space X's website markets Falcon 9 rocket launches at a cost of $62 million, while competitors charge up to $165 million, owing to the former's ability to reuse the first stage. By predicting the probability of a successful landing, we can ascertain the cost of a launch, a valuable asset for alternate companies competing for rocket launch services. This project aims to create a machine learning pipeline to forecast the success rate of the first stage landing.

  ## Problems you want to find answers

  - What factors determine if the rocket will land successfully?
  - The interaction amongst various features that determine the success rate of a successful landing.
  - What operating conditions needs to be in place to ensure a successful landing program.

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - Data was collected using SpaceX API and web scraping.
- Perform data wrangling
  - Replacing missing data with mean. One-hot encoding
- Perform exploratory data analysis using SQL and Data Visualization
- Interactive dashboard using Folium and Plotly Dash
- Machine Learning - classification models

Data Collection

- The data was collected using various methods
  - Diverse data collection methods were utilized.
  - SpaceX API was leveraged for data acquisition through get requests.
  - The response content was decoded into Json format using .json() function call and transformed into a pandas dataframe using .json_normalize().
  - Data cleaning procedures were carried out, encompassing missing value identification and filling.
  - Web scraping was performed from Wikipedia using BeautifulSoup to retrieve Falcon 9 launch records.
  - The HTML table containing the launch records was extracted, parsed, and converted into a pandas dataframe for future analysis.

Data Collection – SpaceX API

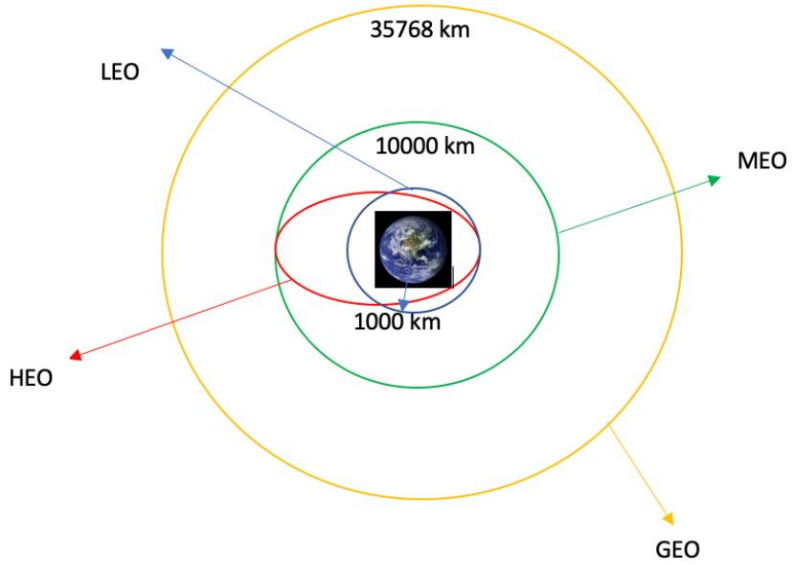
- Data was collected using the get request method to SpaceX API, followed by preliminary data cleaning, basic data wrangling, and formatting procedures.
- The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_data_collection_api.ipynb

Data Collection - Scraping

- Web scraping techniques were employed to extract Falcon 9 launch records using BeautifulSoup.
- The obtained HTML table was parsed and subsequently converted into a pandas dataframe for further analysis.The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_data_collection_api.ipynb

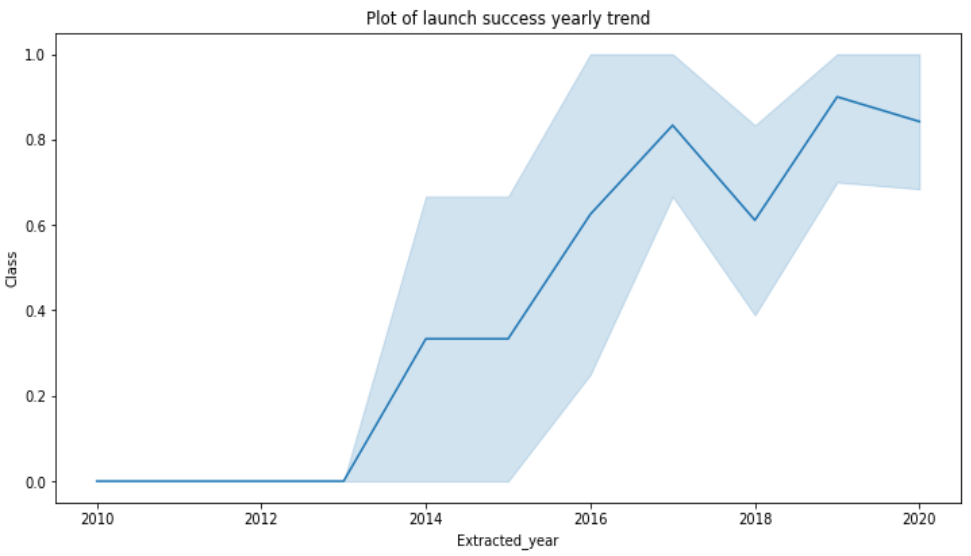
Data Wrangling

- Exploratory data analysis was conducted to determine training labels.
- The number of launches at each site, as well as the frequency and occurrence of each orbit, were computed.
- A landing outcome label was generated from the outcome column.
- The resulting analysis was exported to csv for further use.
- The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_Data_wrangling.ipynb



EDA with Data Visualization

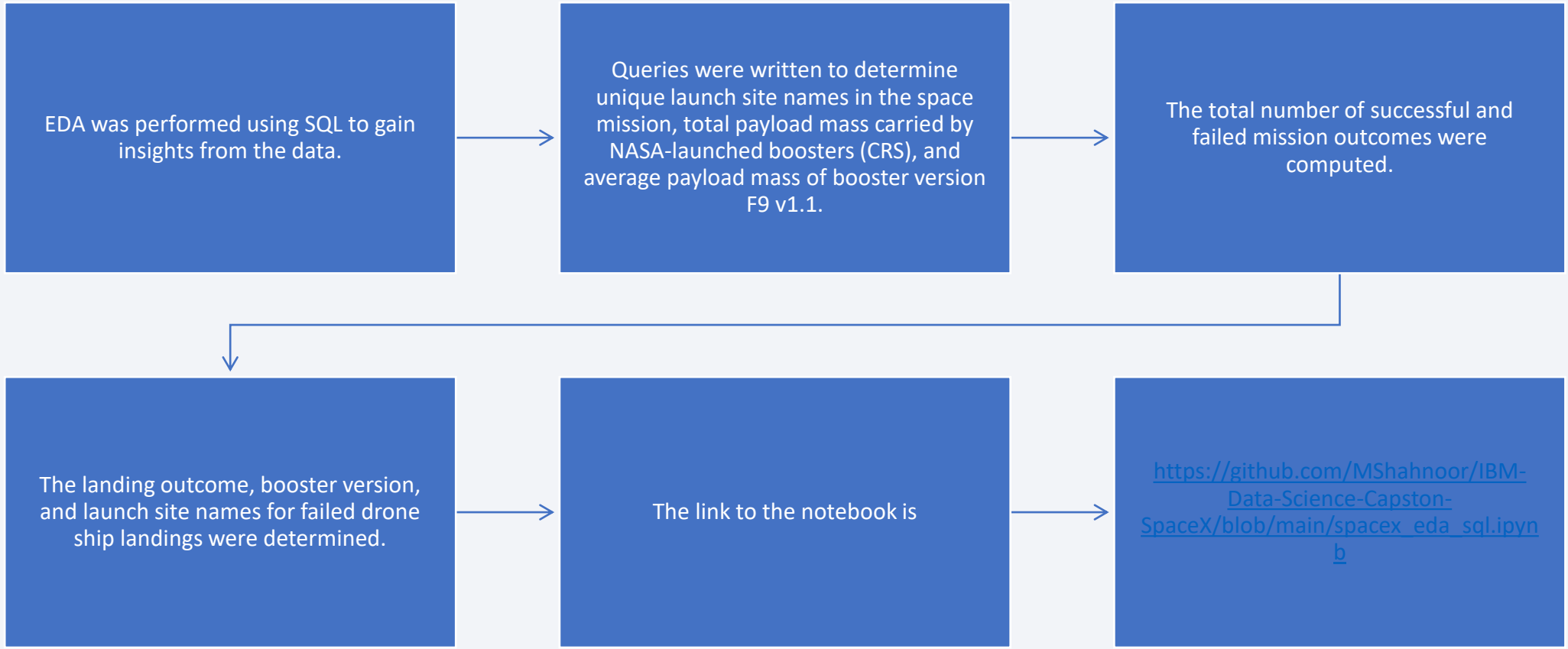
- The data was explored through visualizations of the relationship between flight number and launch site, payload and launch site, success rate of each orbit type, flight number and orbit type, and the yearly trend of launch success.



The link to the notebook is

https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_eda_with_visualization.ipynb

EDA with SQL


EDA was performed using SQL to gain insights from the data.
Queries were written to determine unique launch site names in the space mission, total payload mass carried by NASA-launched boosters (CRS), and average payload mass of booster version F9 v1.1.
The total number of successful and failed mission outcomes were computed.
The landing outcome, booster version, and launch site names for failed drone ship landings were determined.
The link to the notebook is
https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_eda_sql.ipynb

Build an Interactive Map with Folium

- Launch sites were marked and map objects, such as markers, circles, and lines, were added to the folium map to indicate the success or failure of launches for each site.
- Launch outcomes were assigned to class 0 for failure and 1 for success.
- Using color-labeled marker clusters, we identified launch sites with relatively high success rates.
- Distances between launch sites and nearby features, such as railways, highways, coastlines, and cities, were calculated and analyzed to answer specific questions.
- The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_folium_visualization.ipynb

Build a Dashboard with Plotly Dash

- An interactive dashboard was created using Plotly Dash.
- Pie charts were plotted to display the total launches by certain sites.
- Scatter graphs were plotted to show the relationship between outcome and payload mass (kg) for different booster versions.
- The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_interactive_dashboard_code.ipynb

Predictive Analysis (Classification)

- The data was loaded using NumPy and Pandas, transformed, and split into training and testing sets.
- Different machine learning models were built and their hyperparameters were tuned using GridSearchCV.
- The model's performance was evaluated using the accuracy metric and improved via feature engineering and algorithm tuning.
- The best performing classification model was identified.
- The link to the notebook is
- https://github.com/MShahnoor/IBM-Data-Science-Capston-SpaceX/blob/main/spacex_machine_Learning.ipynb

Results

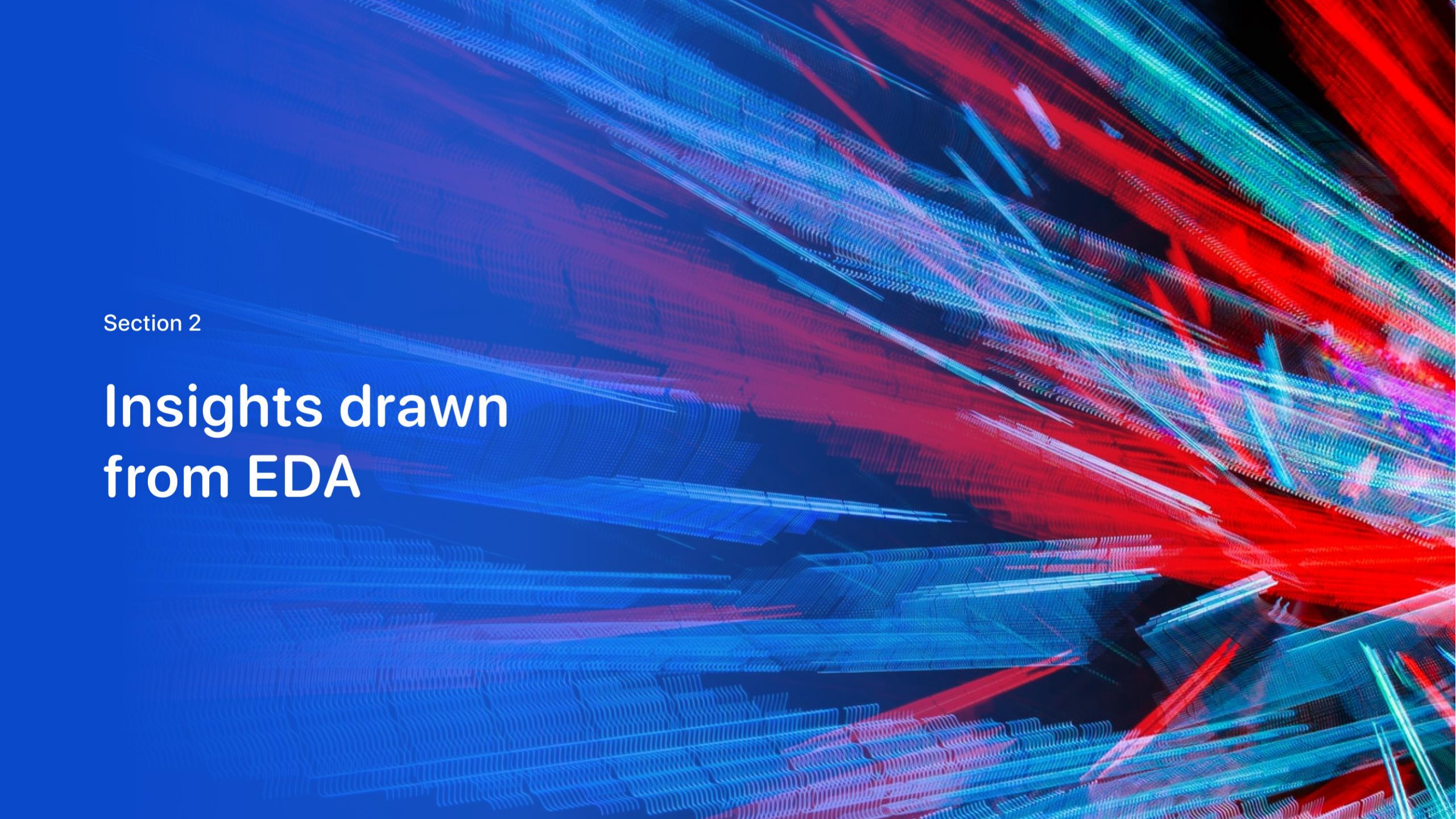
EXPLORATORY DATA ANALYSIS RESULTS

INTERACTIVE ANALYTICS DEMO IN SCREENSHOTS

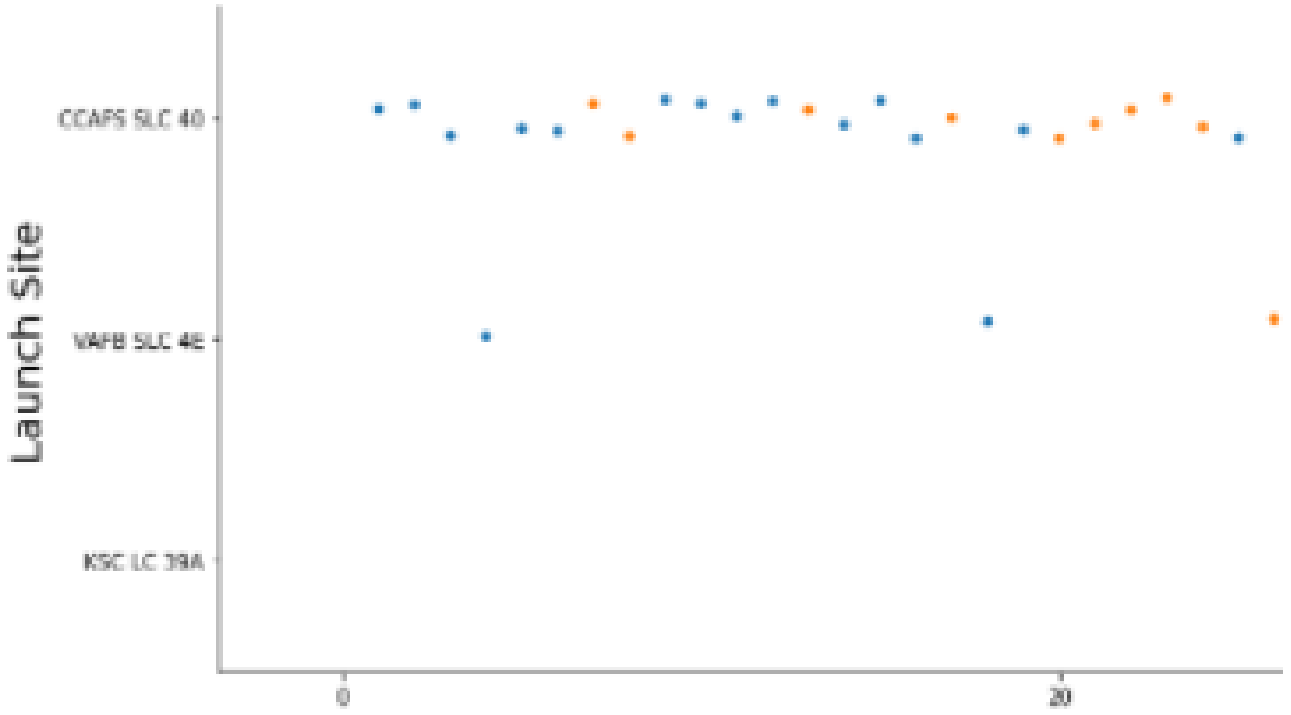
PREDICTIVE ANALYSIS RESULTS

Section 2

Insights drawn from EDA



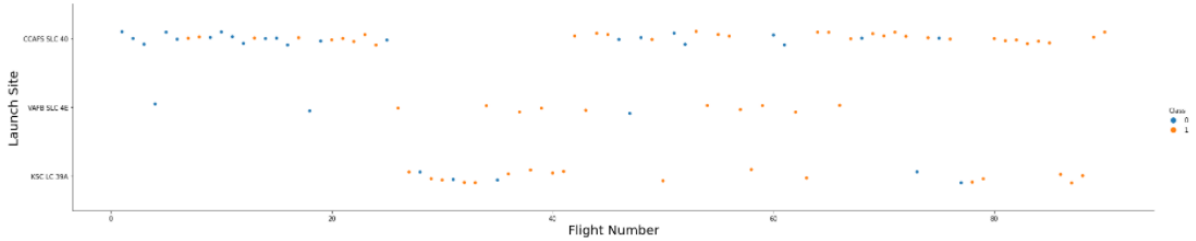
Flight Number vs. Launch Site



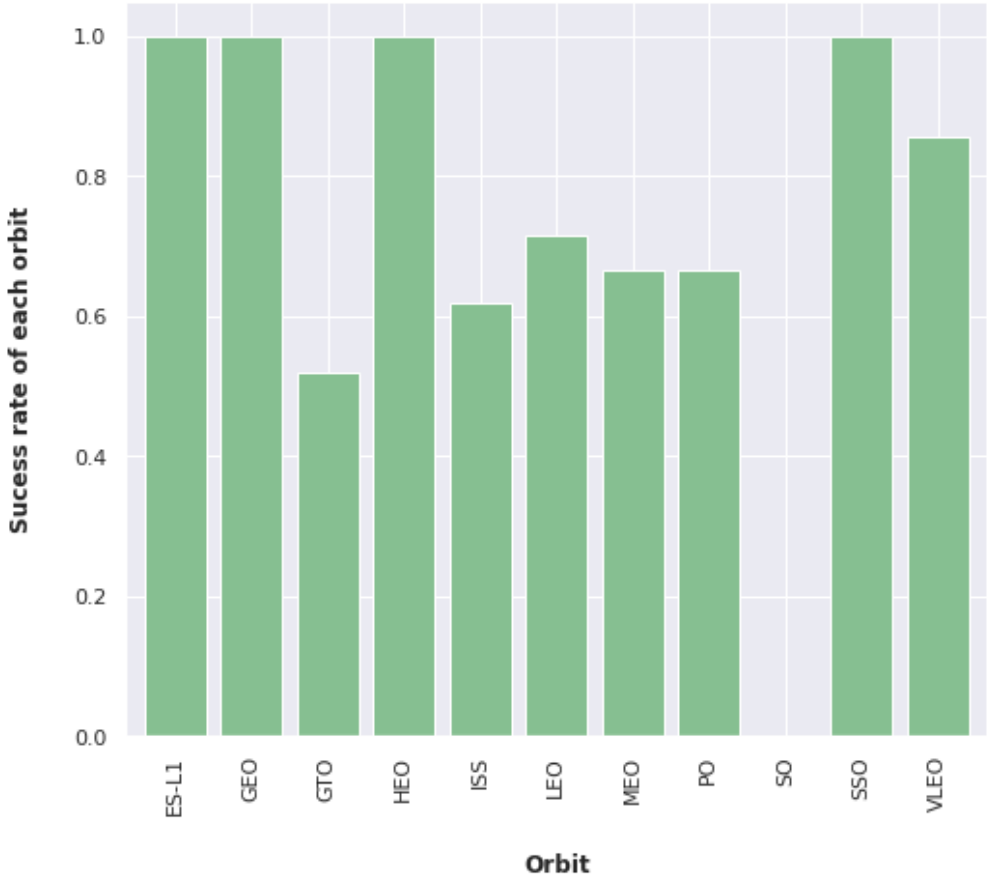
- The plot indicates a positive correlation between the number of flights and success rate at a launch site. As the flight count increases, the success rate also tends to increase.

Payload vs. Launch Site

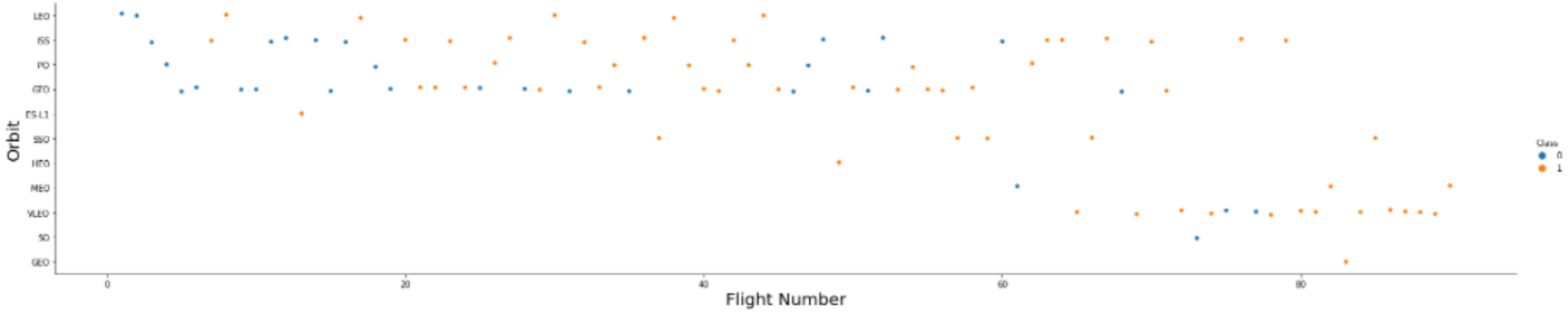
The greater the payload mass for launch site CCAFS SLC 40 the higher the success rate for the rocket.



Success Rate vs. Orbit Type

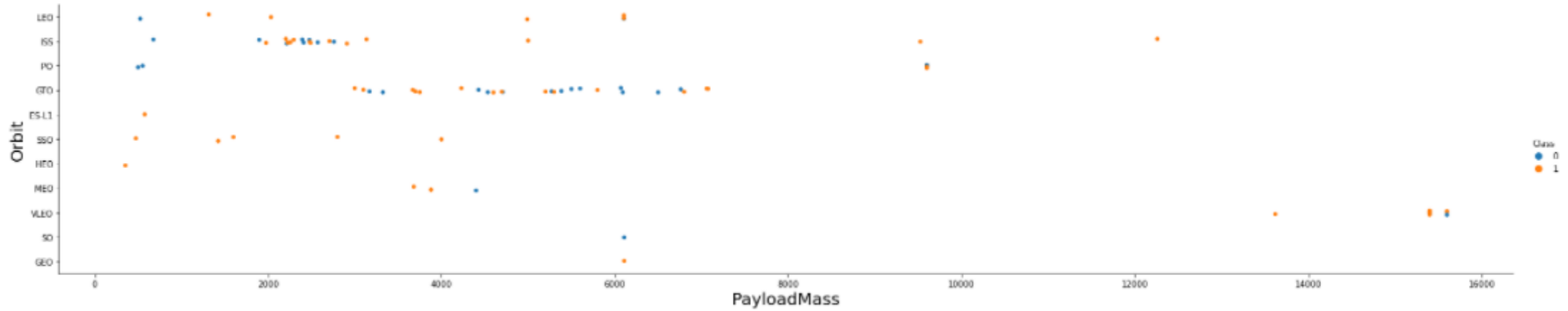


- The plot indicates that the following orbit types had the highest success rates: ES-L1, GEO, HEO, SSO, and VLEO.



Flight Number vs. Orbit Type

- The Flight Number vs. Orbit type plot reveals an interesting observation. In the LEO orbit, there appears to be a correlation between success rate and the number of flights, whereas in the GTO orbit, no such correlation exists.

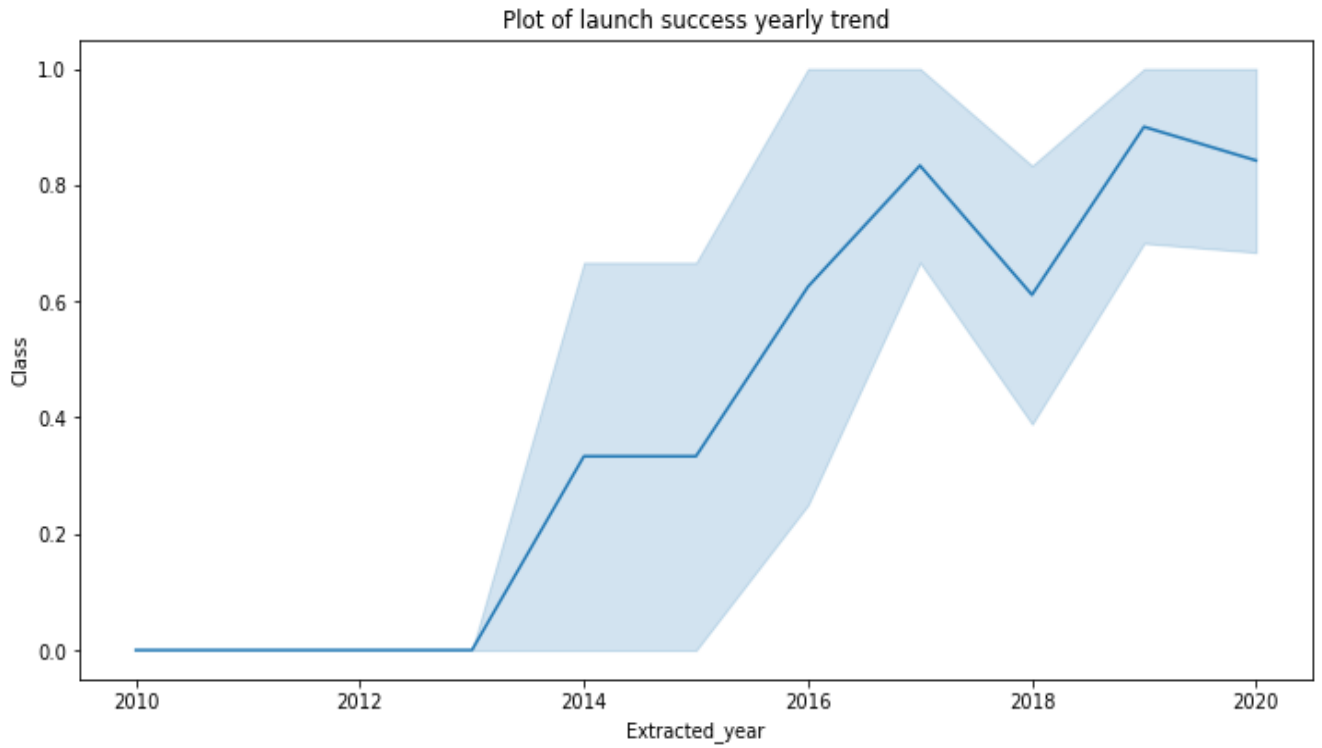


Payload vs. Orbit Type

- The data suggests that successful landings are more frequent in PO, LEO, and ISS orbits with heavy payloads.

Launch Success Yearly Trend

- The plot indicates a gradual increase in the success rate from 2013 to 2020.



All Launch Site Names

- To display only unique launch sites from the SpaceX data, the DISTINCT keyword was utilized.

```
[ ]  1  %sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEX;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

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

Task 2

Display 5 records where launch sites begin with the string 'CCA'

```
[ ]  1  %sql SELECT * FROM SPACEX WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

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

Launch Site Names Begin with 'CCA'

Total Payload Mass

- The total payload carried by NASA boosters was calculated to be 45596 using the following query.

Display the total payload mass carried by boosters launched by NASA (CRS)

```
[ ]  1  %sql SELECT SUM(PAYLOAD_MASS__KG_) AS "Total Payload Mass by NASA (CRS)" FROM SPACEX WHERE CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Total Payload Mass by NASA (CRS)

45596

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 was calculated to be 2928.4.

Display average payload mass carried by booster version F9 v1.1

```
[ ]  1  %sql SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload Mass by Booster Version F9 v1.1" FROM SPACEX \
     2  WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Average Payload Mass by Booster Version F9 v1.1

2928

First Successful Ground Landing Date

- The first successful landing outcome on a ground pad was observed on December 22, 2015.

```
[ ]  1  %sql SELECT MIN(DATE) AS "First Succesful Landing Outcome in Ground Pad" FROM SPACEX \
     2  WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

First Succesful Landing Outcome in Ground Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- We filtered boosters that successfully landed on a drone ship using the WHERE clause and applied an AND condition to select those with a payload mass greater than 4000 and less than 6000, using the query below.

```
[ ]  1  %sql SELECT BOOSTER_VERSION FROM SPACEX WHERE LANDING__OUTCOME = 'Success (drone ship)' \
     2  AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

```
[ ]  1  %sql SELECT COUNT(MISSION_OUTCOME) AS "Successful Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Success%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Successful Mission

100

```
[ ]  1  %sql SELECT COUNT(MISSION_OUTCOME) AS "Failure Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Failure%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Failure Mission

1

```
[ ]  1  %sql SELECT COUNT(MISSION_OUTCOME) AS "Total Number of Successful and Failure Mission" FROM SPACEX \
     2  WHERE MISSION_OUTCOME LIKE 'Success%' OR MISSION_OUTCOME LIKE 'Failure%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Total Number of Successful and Failure Mission

101

- Wildcard like ‘%’ was used to filter for WHERE MissionOutcome was a success or a failure.

Boosters Carried Maximum Payload

- Using a subquery in the WHERE clause and the MAX() function, the booster that has carried the maximum payload was determined.

```
[ ]  1  %sql SELECT DISTINCT BOOSTER_VERSION AS "Booster Versions which carried the Maximum Payload Mass" FROM SPACEX \
     2  WHERE PAYLOAD_MASS__KG_ =(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX);
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Booster Versions which carried the Maximum Payload Mass

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

2015 Launch Records

- For the year 2015, we filtered for failed landing outcomes in drone ship, their booster versions, and launch site names using a combination of the WHERE clause, LIKE, AND, and BETWEEN conditions.

```
[ ]  1  %sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE DATE LIKE '2015-%' AND \
     2  LANDING__OUTCOME = 'Failure (drone ship)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

```
[ ]  1  %sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE year(DATE) = '2015' AND \
     2  LANDING__OUTCOME = 'Failure (drone ship)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

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

- The query selected Landing outcomes and the COUNT of landing outcomes from the data, filtered for landing outcomes between 2010-06-04 to 2010-03-20 using the WHERE clause.
- The GROUP BY clause was applied to group the landing outcomes and the ORDER BY clause was used to order the grouped landing outcome in descending order.

```
[ ]  1  %sql SELECT LANDING__OUTCOME as "Landing Outcome", COUNT(LANDING__OUTCOME) AS "Total Count" FROM SPACEX \
     2  WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
     3  GROUP BY  LANDING__OUTCOME \
     4  ORDER BY COUNT(LANDING__OUTCOME) DESC ;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Landing Outcome	Total Count
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

```
[ ]  1  %sql SELECT COUNT(LANDING__OUTCOME) AS "Rank success count between 2010-06-04 and 2017-03-20" FROM SPACEX \
     2  WHERE LANDING__OUTCOME LIKE '%Success%' AND DATE > '2010-06-04' AND DATE < '2017-03-20' ;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

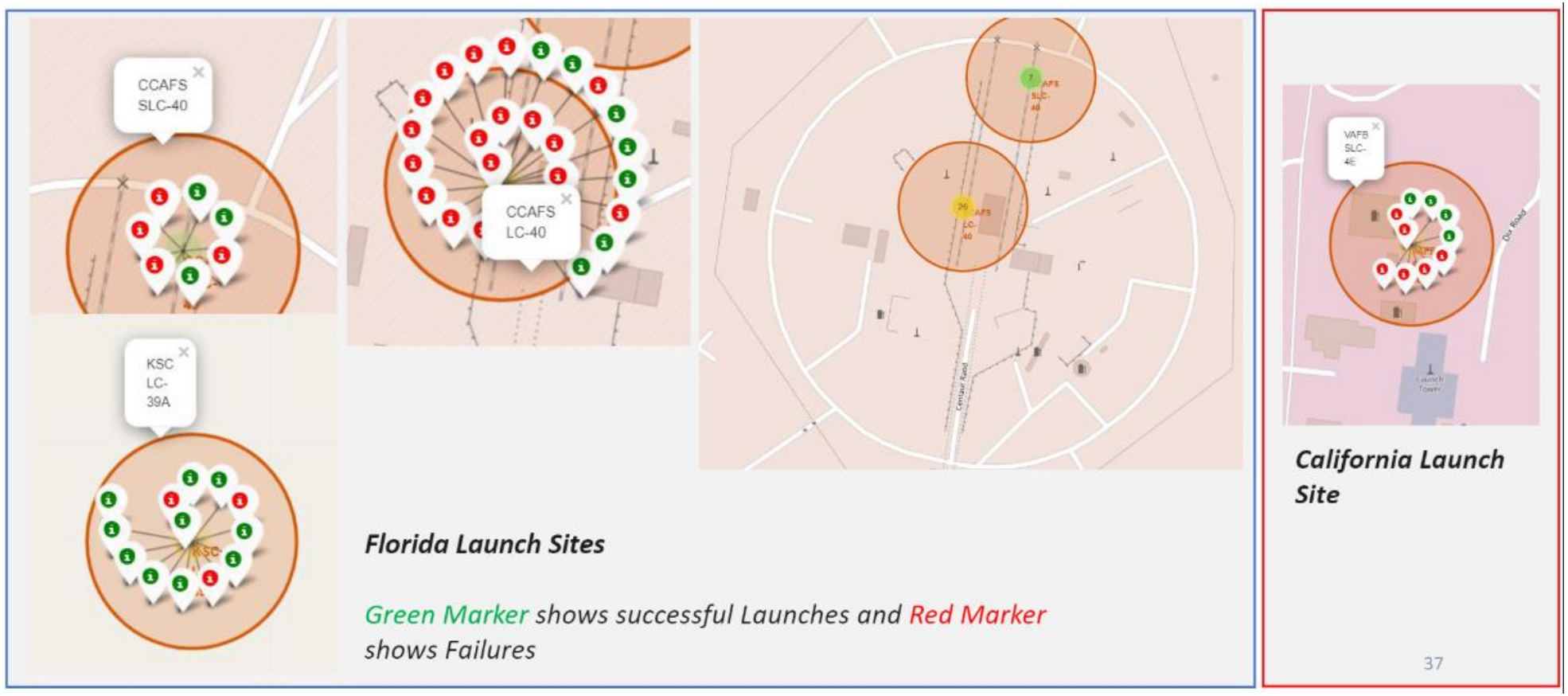
Rank success count between 2010-06-04 and 2017-03-20
8

Section 4

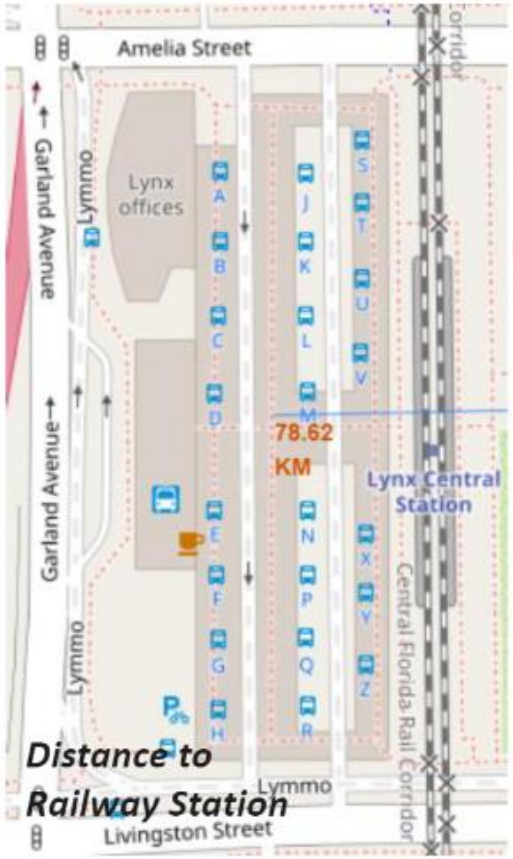
Launch Sites Proximities Analysis

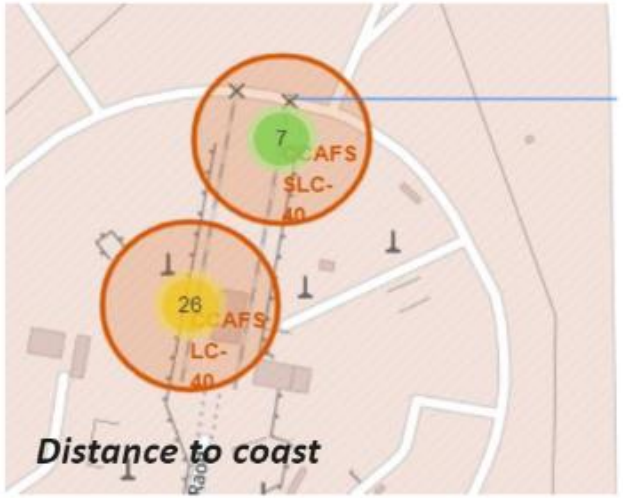
All launch sites global map markers

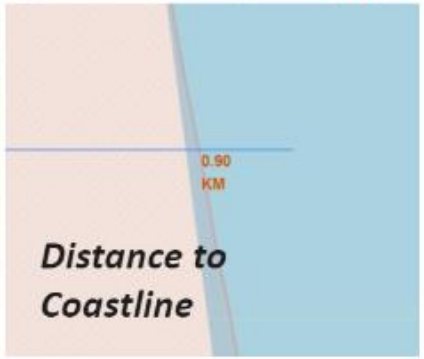
Markers showing launch sites with color labels



Launch Site distance to landmarks



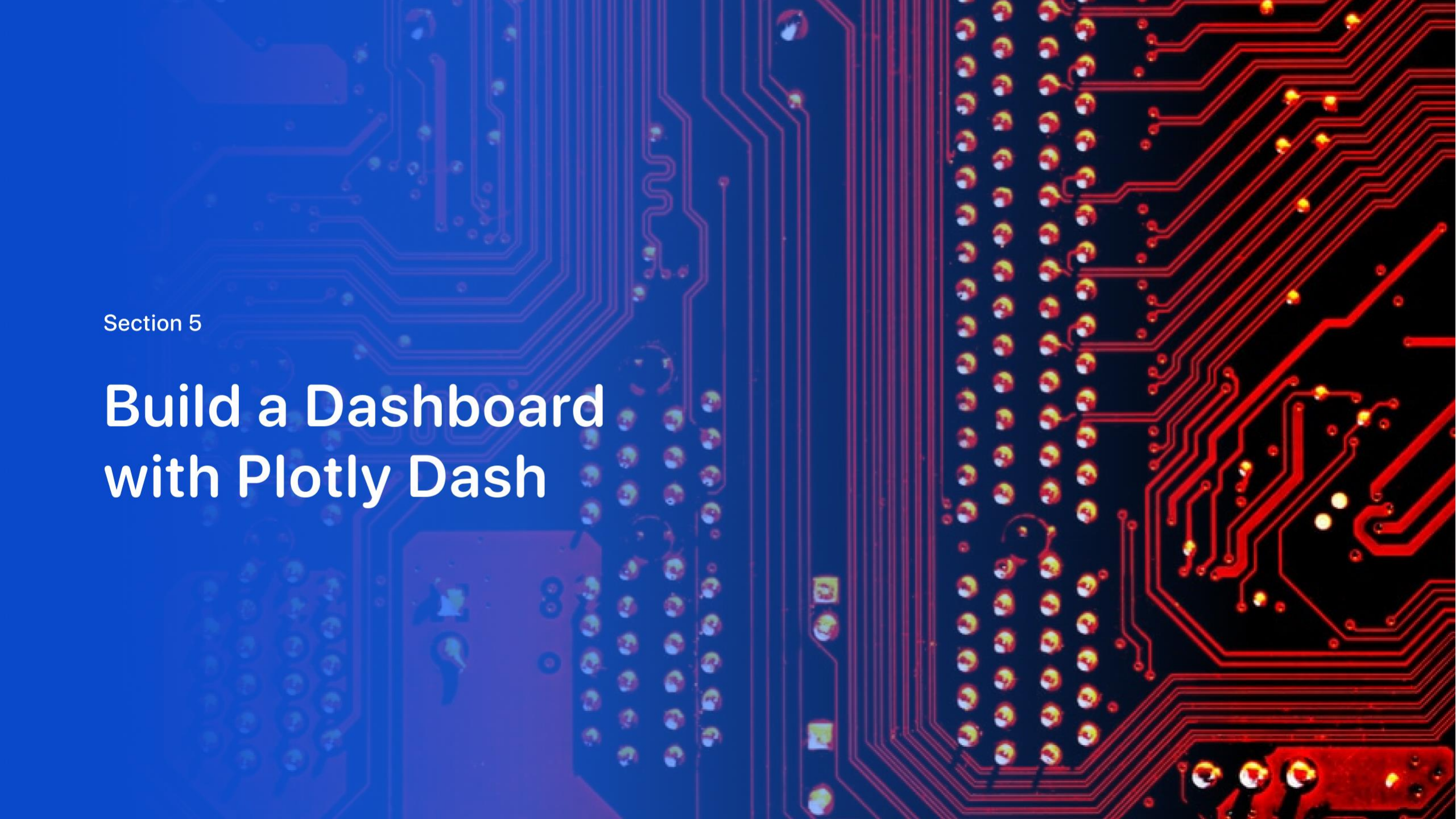




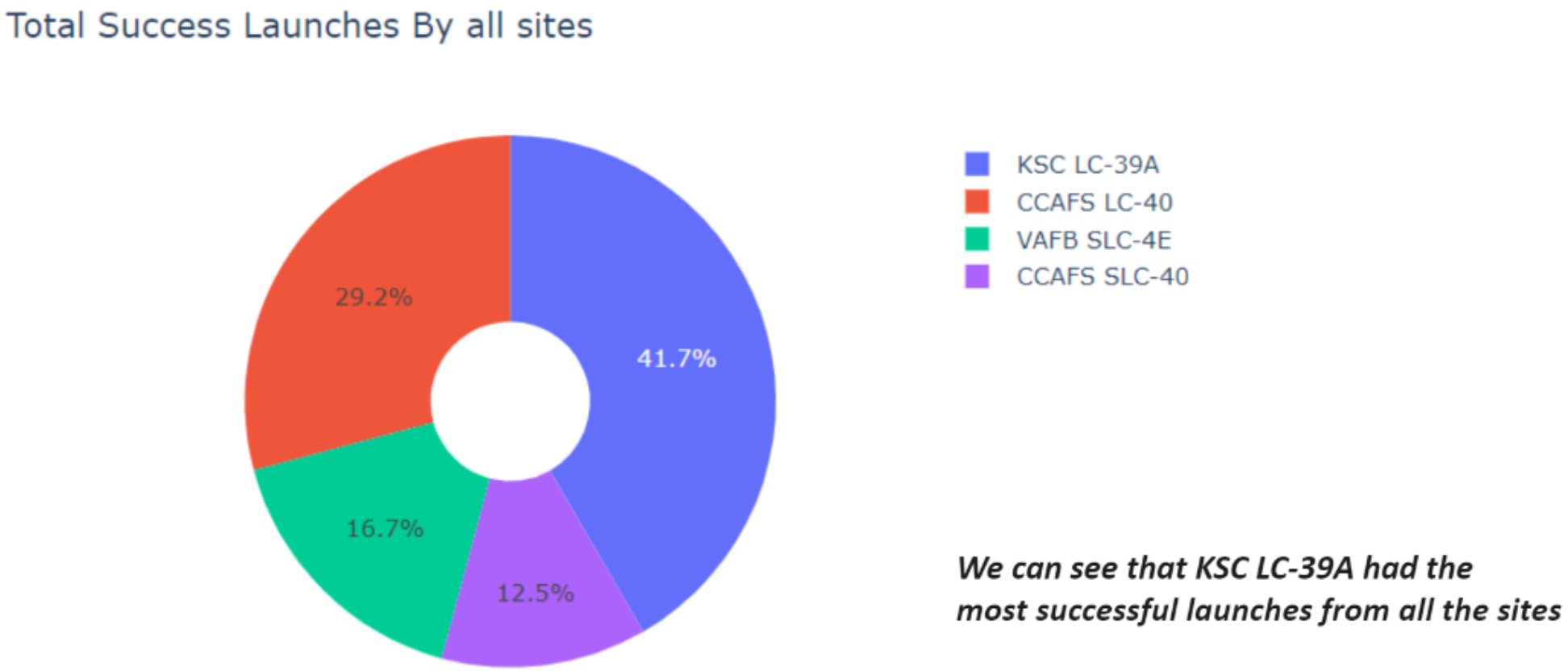
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

Section 5

Build a Dashboard with Plotly Dash

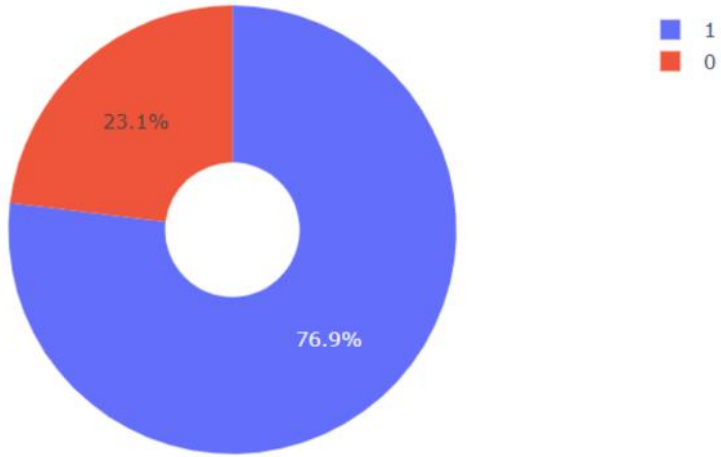


Pie chart showing the success percentage achieved by each launch site



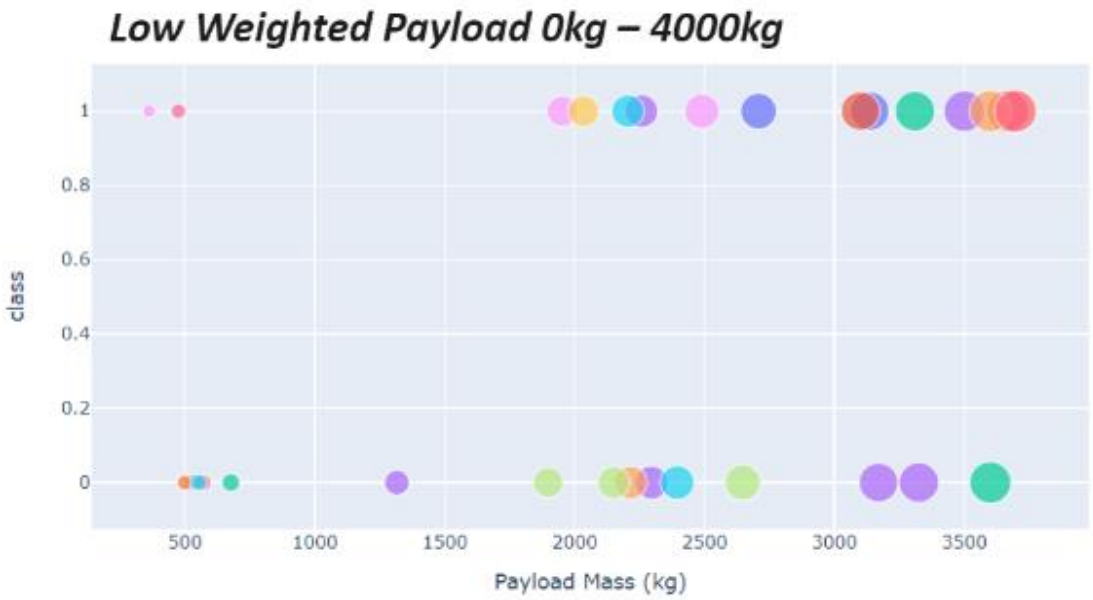
We can see that KSC LC-39A had the most successful launches from all the sites

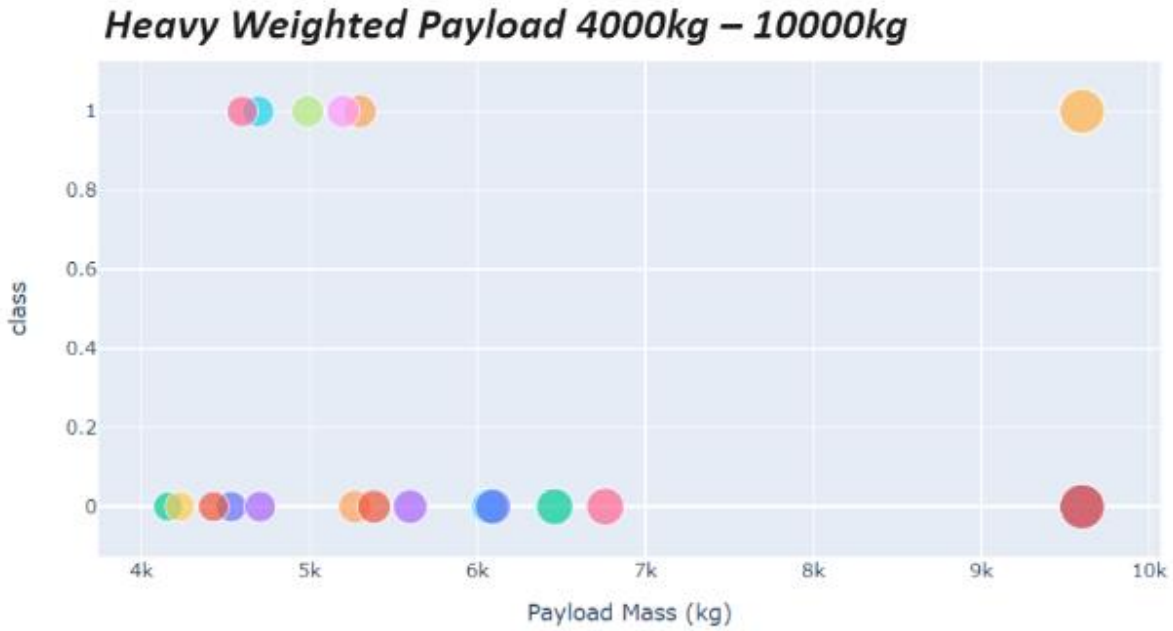
Pie chart showing the Launch site with the highest launch success ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider





We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 6

Predictive Analysis (Classification)

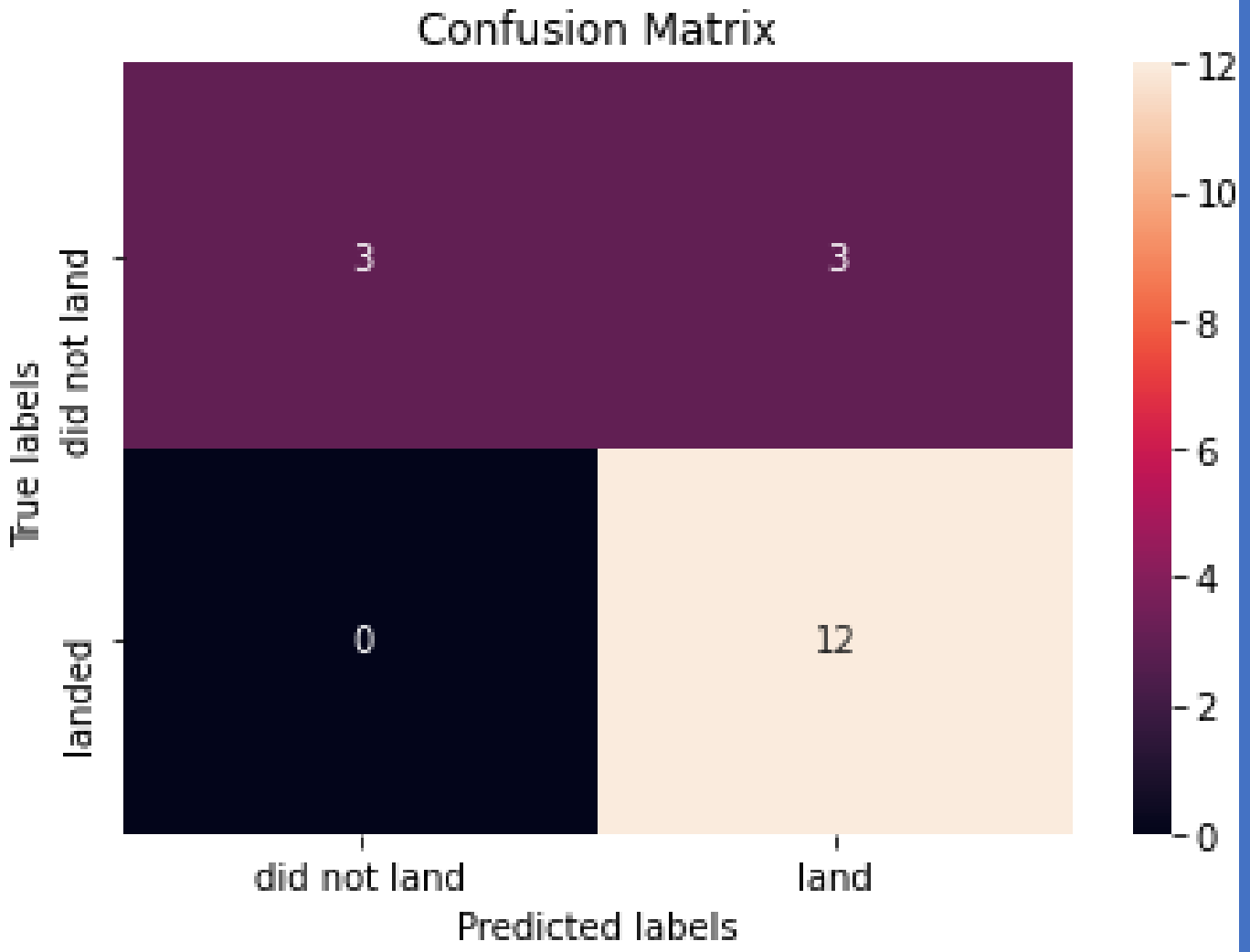
Classification Accuracy

```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm]
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto'
```

- The decision tree classifier is the model with the highest classification accuracy



Confusion Matrix

- Based on the confusion matrix for the decision tree classifier, it can be observed that the classifier is able to differentiate between the different classes. However, the main issue is the occurrence of false positives, which is when unsuccessful landings are incorrectly classified as successful landings by the classifier.

Conclusions

- The following conclusions can be drawn from the analysis:
- The success rate at a launch site is positively correlated with the flight amount at that site.
- From 2013 to 2020, the success rate of launches showed a consistent increase.
- Orbits such as ES-L1, GEO, HEO, SSO, and VLEO have the highest success rates.
- KSC LC-39A had the most successful launches compared to other sites.
- Based on the analysis, the Decision tree classifier is the optimal machine learning algorithm for this task.

Thank you!