Winning Space Race with Data Science

Ana Pastore
April 4, 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data Wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building a dashboard with Dash
  - Predictive analysis (Classification)
- Summary of all results
  - Exploratory data analysis
  - Interactive analytics
  - Predictive analysis results

Introduction

- Project background and context
  - We are living in the era of commercial space, with companies like SpaceX, Blue Origin, and others making space travel affordable.
  - In particular SpaceX advertises the cheapest launch among competitors, as much of the savings are due to the reuse of the first stage.
  - We want to predict if the Falcon 9 first stage will land successfully. If this is possible, we can determine the cost of a launch.

- Problems you want to find answers
  - Correlations between each rocket variables and successful landing rate
  - Conditions to get the best results and ensure the best successful landing rate

Section 1

Methodology

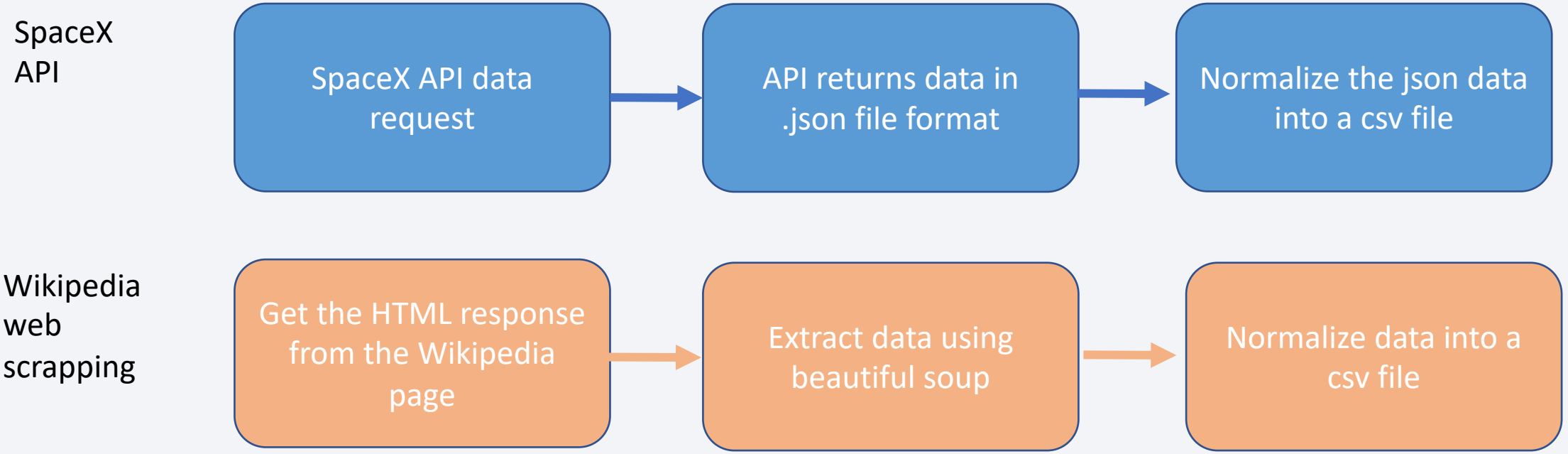
Methodology

Executive Summary

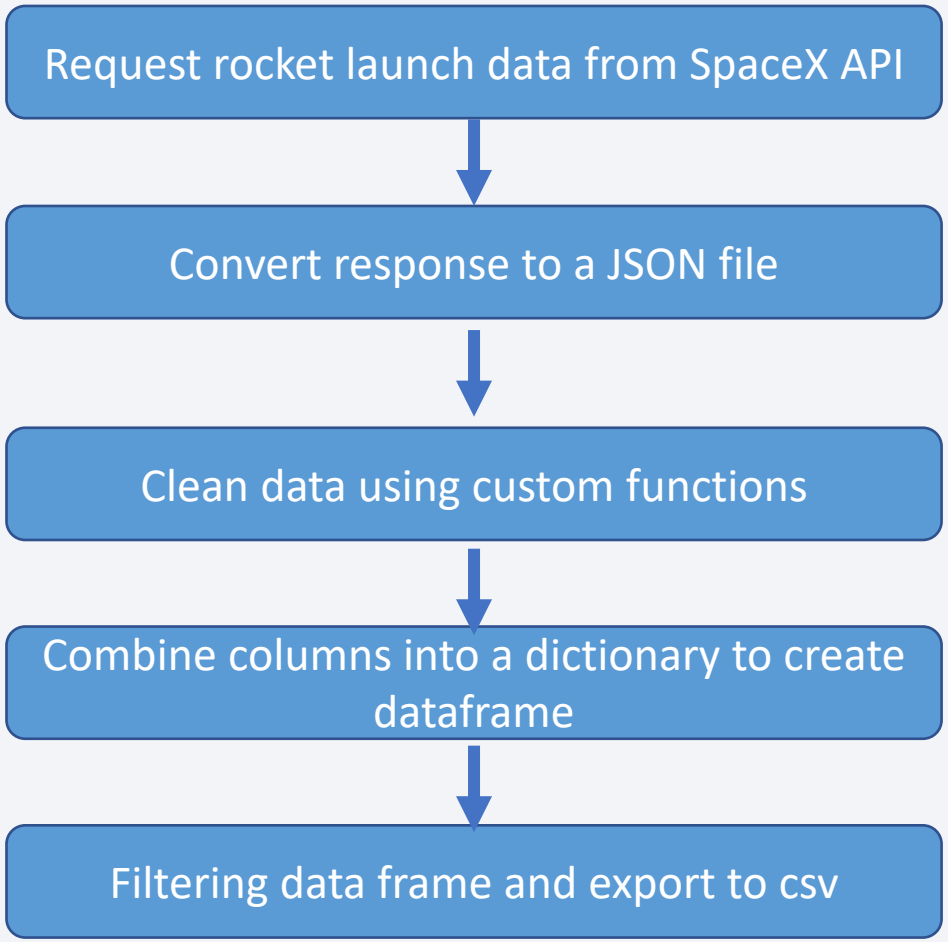
- Data collection methodology:
  - We will collect the data from SpaceX API and a Wikipedia page containing the info about Falcon 9 launches via Web Scrapping.
- Perform data wrangling
  - We will convert the outcomes of the launches into training labels for the booster (successfully/unsuccessful)
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Find best hyperparameters for SVM, classification trees and logistic regression

Data Collection

The data collected included a combination of API requests from the SpaceX API and web scrapping data from a table in a Wikipedia page with information about the SpaceX's Falcon 9 and Falcon Heavy Launches records.



Data Collection – SpaceX API



[Github]

```
spacex_url="https://api.spacexdata.com/v4/launches/past"

response = requests.get(spacex_url)
```

```
# Use json_normalize meethod to convert the json result into a dataframe
from pandas import json_normalize

results = response.json()
data = pd.DataFrame()
data = pd.json_normalize(results)
```

```
# Call getLaunchSite
getLaunchSite(data)

# Call getPayloadData
getPayloadData(data)

# Call getCoreData
getCoreData(data)
```

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

```
df = pd.DataFrame.from_dict(launch_dict)
```

Show the summary of the dataframe

```
# Show the head of the dataframe
df.head()
```

Data Collection - Scraping

Getting response from HTML

```
data  = requests.get(static_url).text
```

Create a beautiful soup object

```
# Use BeautifulSoup() to create a BeautifulSoup object
soup = BeautifulSoup(data, 'html5lib')
```

Finding all tables

```
# Use the find_all function in the BeautifulSoup object,
# Assign the result to a list called `html_tables`
html_tables = soup.find_all('table')
```

Extract column's names

```
column_names = []

# Apply find_all() function with `th` element on first_la
# Iterate each th element and apply the provided extract_
# Append the Non-empty column name (`if name is not None

first_launch_table.find_all('tr')
column_names = []
for row in first_launch_table.find_all('tr'):
    columns = row.find_all('th')
    for i in range(len(columns)):
        print(i,columns[i].text.strip())
        column_names.append(columns[i].text.strip())
    break
```

Creating empty dictionary with the correct keys

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date andtime (UTC)']

# Let's initial the launch_dict with each value
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload[c]'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launchoutcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

Fill up the "launch_dict" with records

Create Dataframe and exporting to csv

```
df=pd.DataFrame(launch_dict)
print(df)
```

[Github]

Data Wrangling

Compute number of launches per site

```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

Compute the number and occurrence of each orbit

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

Compute number and occurrence of mission outcome per orbit type

```
# landing_outcomes = values on Outcome column
landing_outcomes = df.Outcome.value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

Create a landing outcome (successfully/unsuccesful)

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
landing_class = []

for row in range(len(df['Outcome'])):
    outc = df['Outcome'].iloc[row]
    if outc in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
```

Calculate rate of success for every landing

```
df["Class"].mean()
```

```
0.6666666666666666
```

Export data into a csv file

[Github]

EDA with Data Visualization

Scatter chart: A scatter plot shows how much one variable is affected by another. The relationship between two variables is called a correlation. This plot is generally composed of large data bodies.

- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Flight Number vs. Orbit Type
- Payload vs. Orbit Type

Bar chart: A Bar chart makes it easy to compare datasets between multiple groups at a glance. One axis represents a category and the other axis represents a discrete value. The purpose of this chart is to indicate the relationship between the two axes.

- Orbit Type vs. Success Rate

Line chart: A Line chart shows data variables and trends very clearly and helps predict the results of data that has not yet been recorded.

- Year vs. Success Rate

Github

EDA with SQL

Loading the dataset into the corresponding table in a Db2 database, and executing SQL queries to answer following questions:

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'CCA'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date when the first successful landing outcome in ground pad was achieved
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster_versions which have carried the maximum payload mass
- Listing the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground
- pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Github

Build an Interactive Map with Folium

Objects created and added to a folium map:

- Markers that show all launch sites on a map
- Markers that show the success/failed launches for each site on the map
- Lines that show the distances between a launch site to its proximities

By adding these objects, following geographical patterns about launch sites are found:

- Are launch sites in close proximity to railways? Yes
- Are launch sites in close proximity to highways? Yes
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

Github

Build a Dashboard with Plotly Dash

The dashboard application contains a pie chart and a scatter plot chart.

Pie chart: We use this type of chart to show the total success of launches by site.
This chart can be selected to indicate a successful landing distribution across all launch sites or to indicate the success rate of individual launch sites.

Scatter chart: We use this chart to show the relationship between two variables: Outcomes and Payload_mass_kg by booster type.

- Has 2 inputs: All sites/individual site & Payload mass on a slider between 0 and 10000kg.
- This chart helps determine how success depends on the launch point, payload mass, and booster version categories.

Github

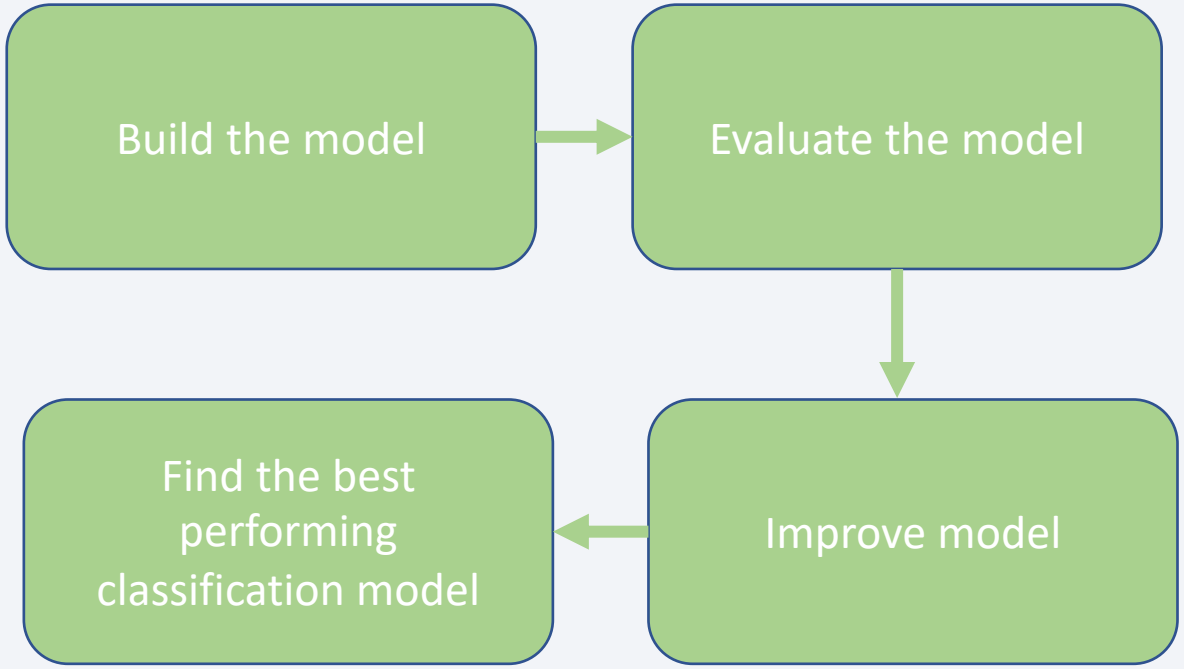
Predictive Analysis (Classification)

Objectives
Perform exploratory Data Analysis and determine Training Labels:

- create a column for the class
- Standardize the data
- Split into training data and test data

Find best Hyperparameter for SVM, Classification Trees and Logistic Regression

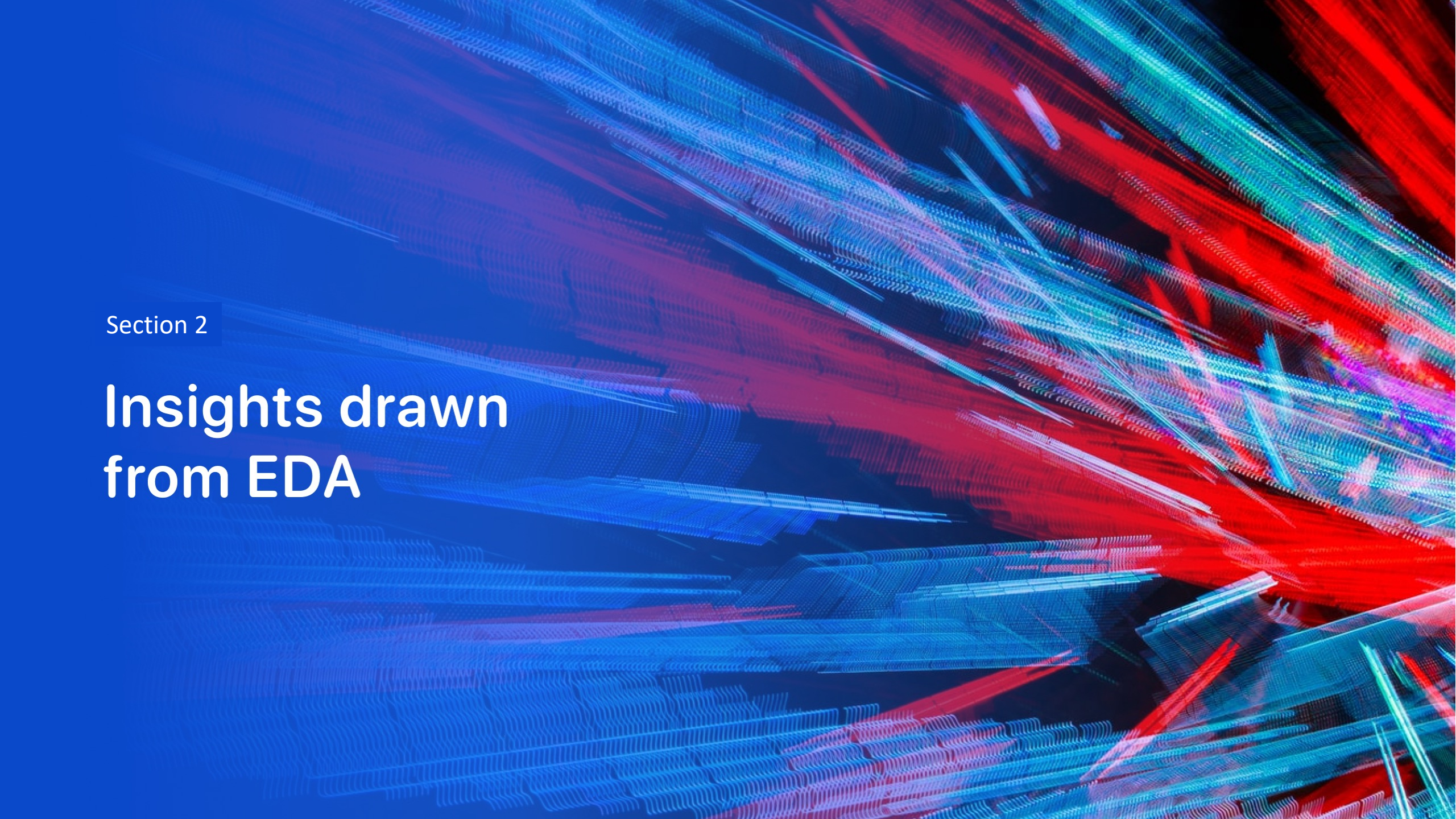
- Find the method performs best using test data



Github

Results

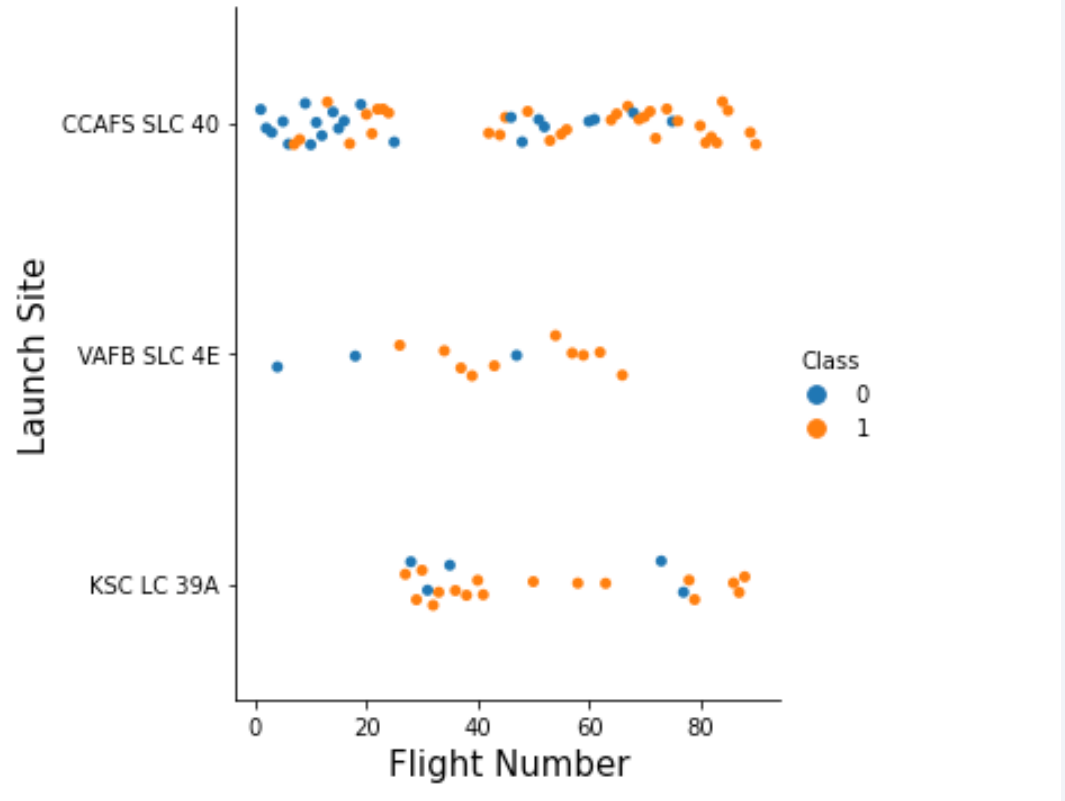
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.

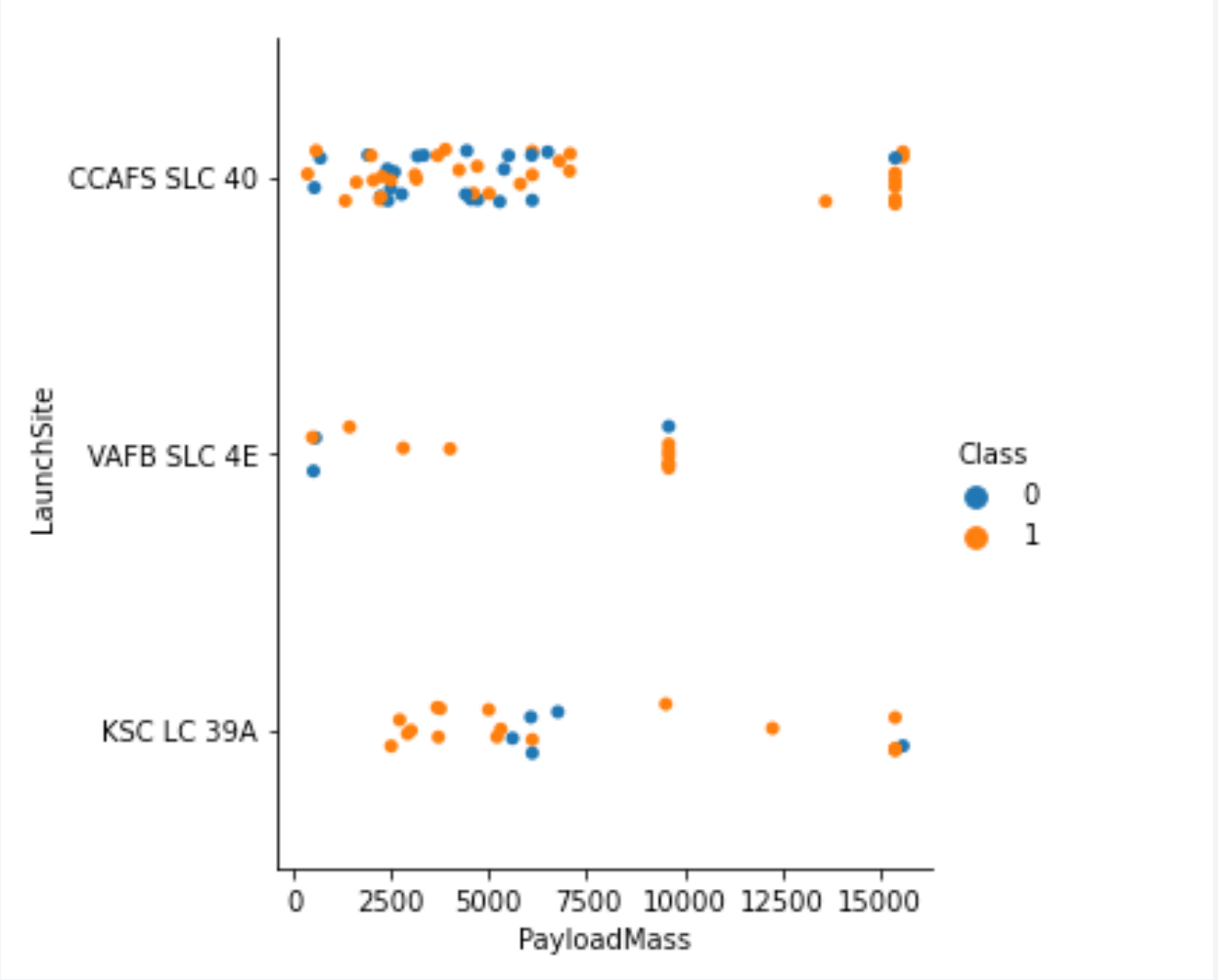
This figure shows that the success rate increased as the number of flights increased.



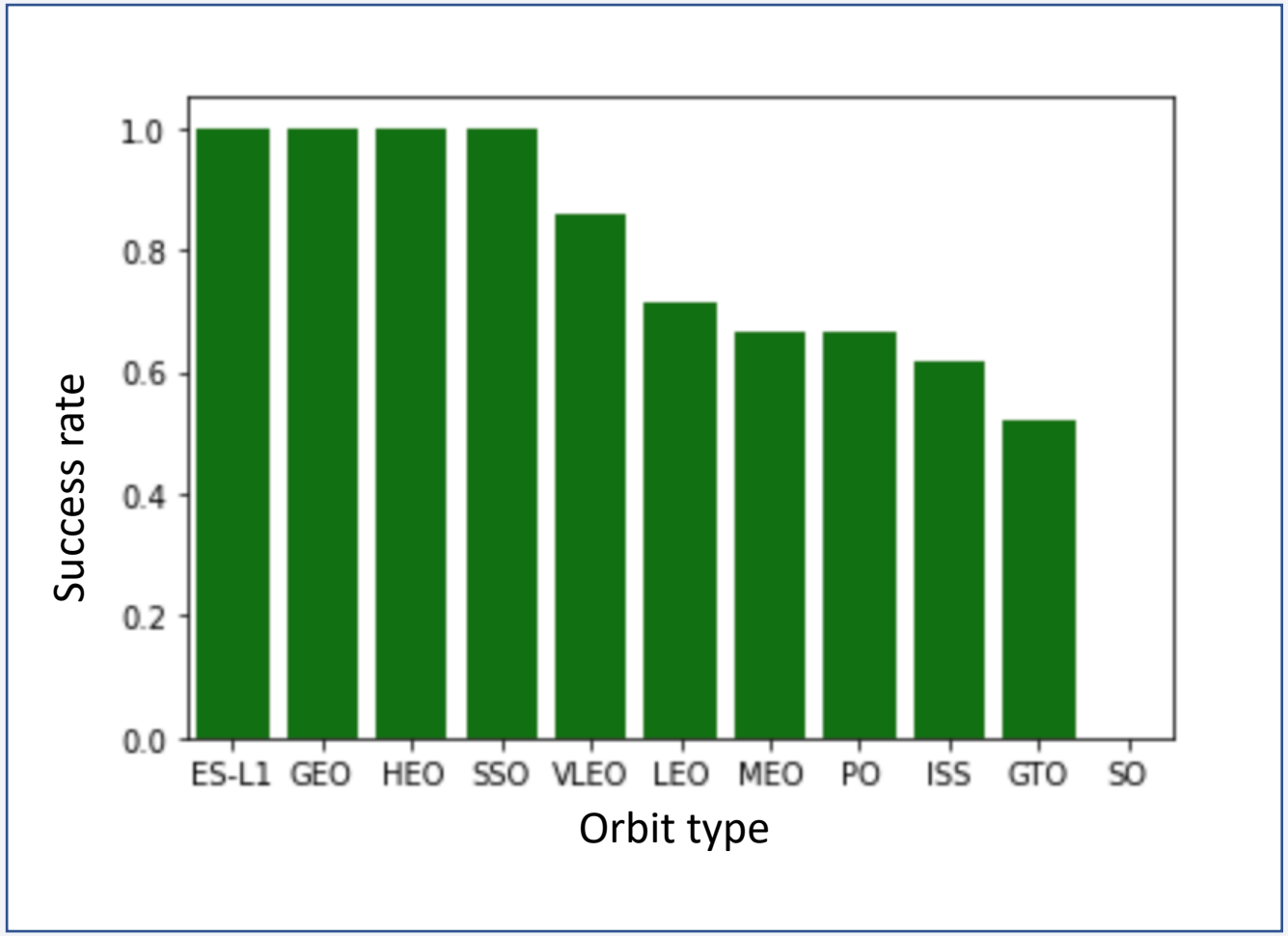
Payload vs. Launch Site

Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.

The larger pay load mass, the higher the rocket's success rate, however there is no clear pattern between successful launch and Pay Load Mass.



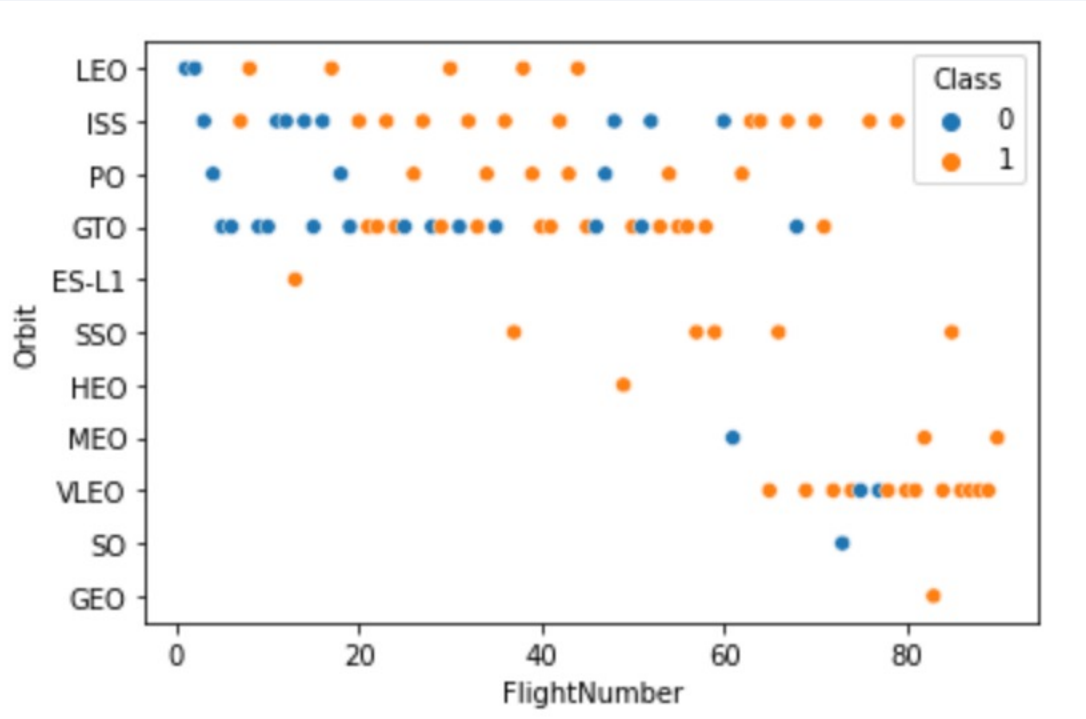
Success Rate vs. Orbit Type



- Orbit types SSO, HEO, GEO, and ES-L1 have the highest success rates (100%).

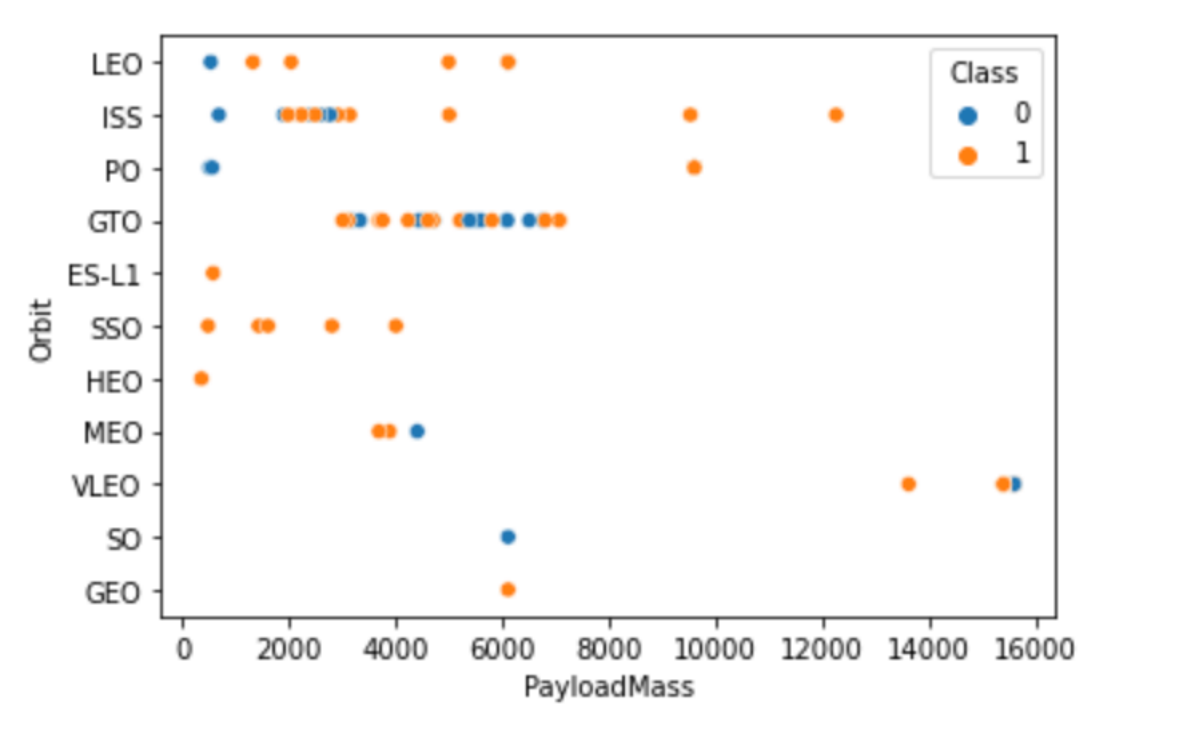
- On the other hand, the success rate of orbit type GTO is only 50%, and it is the lowest except for type SO, which recorded failure in a single attempt.

Flight Number vs. Orbit Type



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.

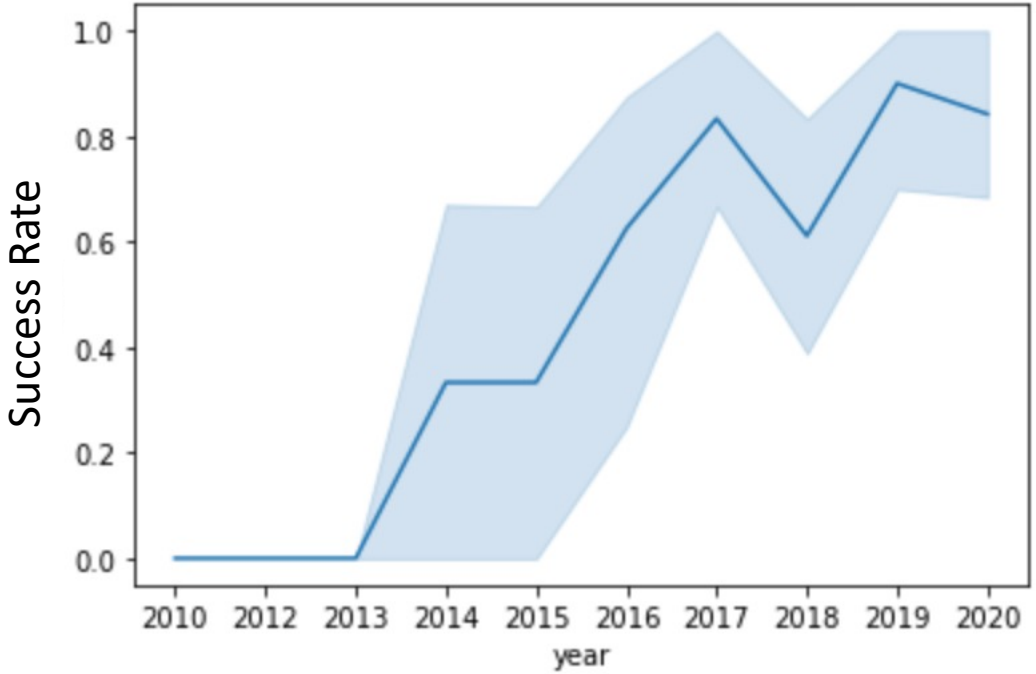
Payload vs. Orbit Type



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.

Launches Success Yearly Trend



The success rate since 2013 kept increasing till 2020

All Launch Site Names

Display the names of the unique launch sites in the space mission

```
%%sql
SELECT DISTINCT(launch_site)
FROM SPACEX;
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c
Done.
```

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

There are 4 different sites

We use the “DISTINCT” statement to select the different launch sites.

Launch Site Names Begin with 'CCA'

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

```
%%sql
SELECT * FROM SPACEX
WHERE LAUNCH_SITE LIKE 'CCA%'
LIMIT 5
```

We use the “%” character to retrieve all sites that start with “CCA”, and limited to the first 5

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32304/BLUDB
Done.
```

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

```
%%sql
SELECT SUM(payload_mass__kg_) AS total_payload_mass_kg
FROM SPACEX
WHERE CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32304/BLUDB
Done.
```

total_payload_mass_kg
45596

We use the SUM statement to the total payload mass

Average Payload Mass by F9 v1.1

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

```
%%sql
SELECT AVG(payload_mass__kg_) AS booster_version_avg FROM SPACEX
WHERE booster_version = 'F9 v1.1';
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj
Done.
```

booster_version_avg
2928

We use the AVG statement to the average payload mass for the booster version F9 v1.1

First Successful Ground Landing Date

List the date when the first successful landing outcome in ground pad was acheived.

Hint:Use min function

```
%%sql

SELECT min(DATE) AS first_succes FROM SPACEX
WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0
Done.
```

first_succes
2015-12-22

We find the successful landing in a ground pad.

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%%sql
SELECT BOOSTER_VERSION FROM SPACEX
WHERE LANDING__OUTCOME = 'Success (drone ship)'
AND payload_mass__kg_ BETWEEN 4000 AND 6000;
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32
Done.
```

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

We list the names of the boosters which have a success in drone ship and have a payload mass between 4,000 and 6,000

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%%sql
SELECT COUNT(MISSION_OUTCOME),MISSION_OUTCOME FROM SPACEX
GROUP BY MISSION_OUTCOME;
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900b
Done.
```

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

Missions outcomes

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%%sql
SELECT DISTINCT(BOOSTER_VERSION),payload_mass__kg_ FROM SPACEX
WHERE payload_mass__kg_ in (SELECT max(payload_mass__kg_) FROM SPACEX);
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdoma
Done.
```

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

Booster versions with the maximum payload mass

2015 Launch Records

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%%sql

SELECT booster_version,launch_site,landing__outcome FROM SPACEX
WHERE landing__outcome = 'Failure (drone ship)'
AND DATE LIKE '2015-%';
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.app
Done.
```

booster_version	launch_site	landing__outcome
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

```
%%sql
SELECT landing__outcome,COUNT(landing__outcome) AS count FROM SPACEX
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP by landing__outcome
ORDER BY count DESC;
```

```
 * ibm_db_sa://bkj23793:***@b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32304/BLUDB
Done.
```

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


Section 3
Launch Sites
Proximities Analysis

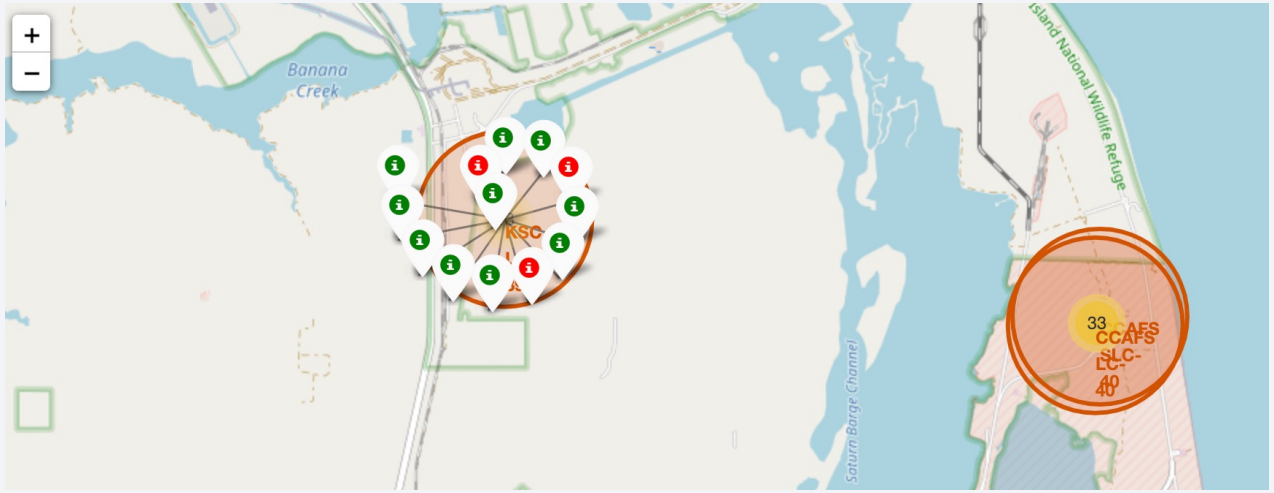
Launch Sites' Locations for SpaceX

- The top map shows all SpaceX launch sites in the United States.
- The map on the left also shows that all launch sites are all in the United States.
- As can be seen on the map, all launch sites are near the coast.

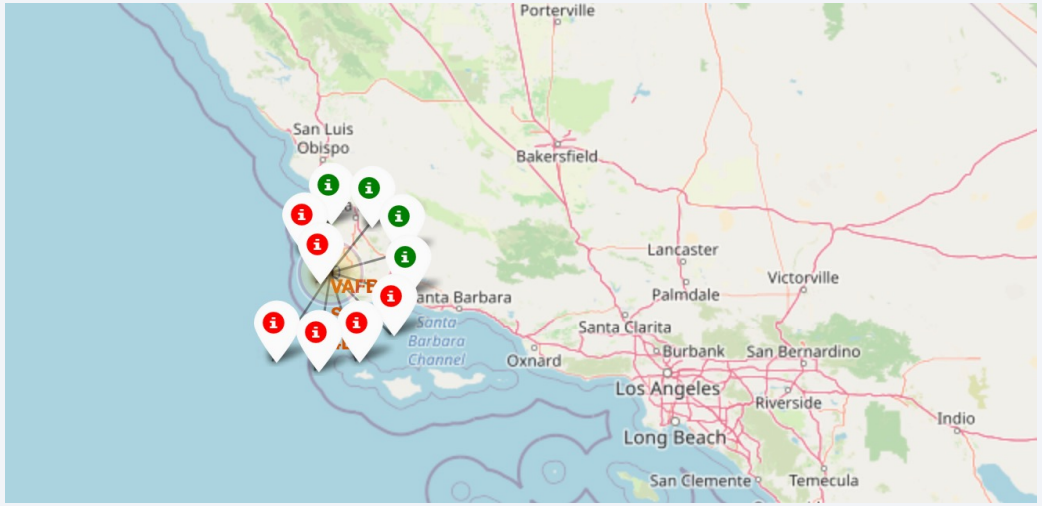
Color labeled launch outcomes

By clicking on the marker clusters, successful landing (green) or failed landing (red) are displayed.

Sites in Florida (some)



Sites in California

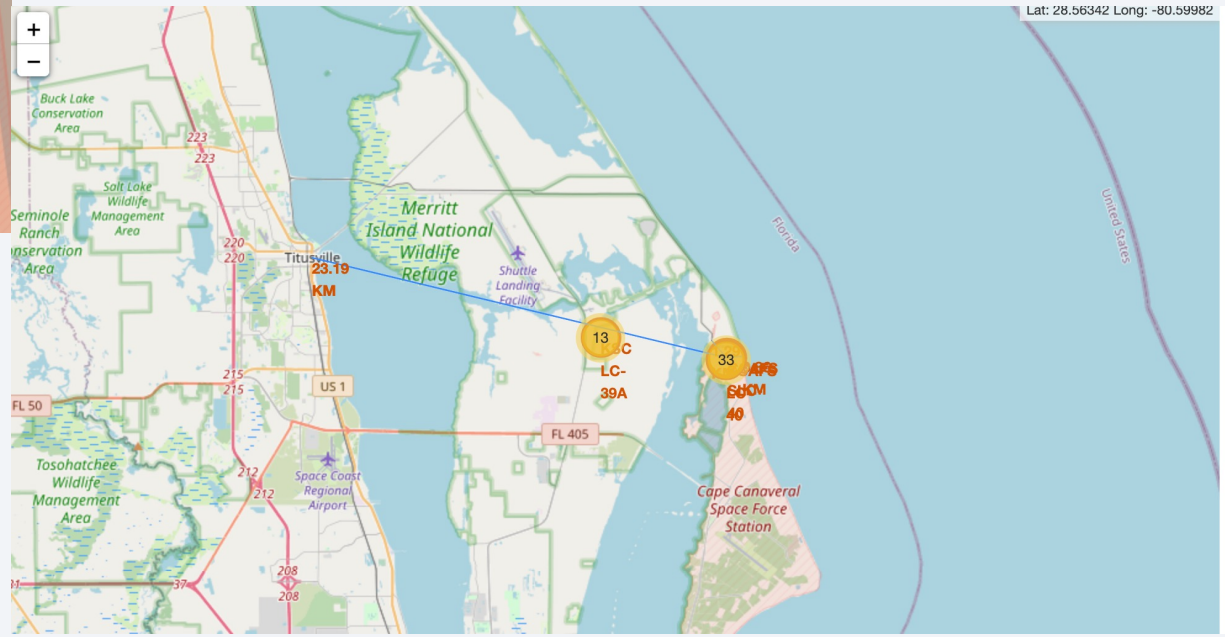


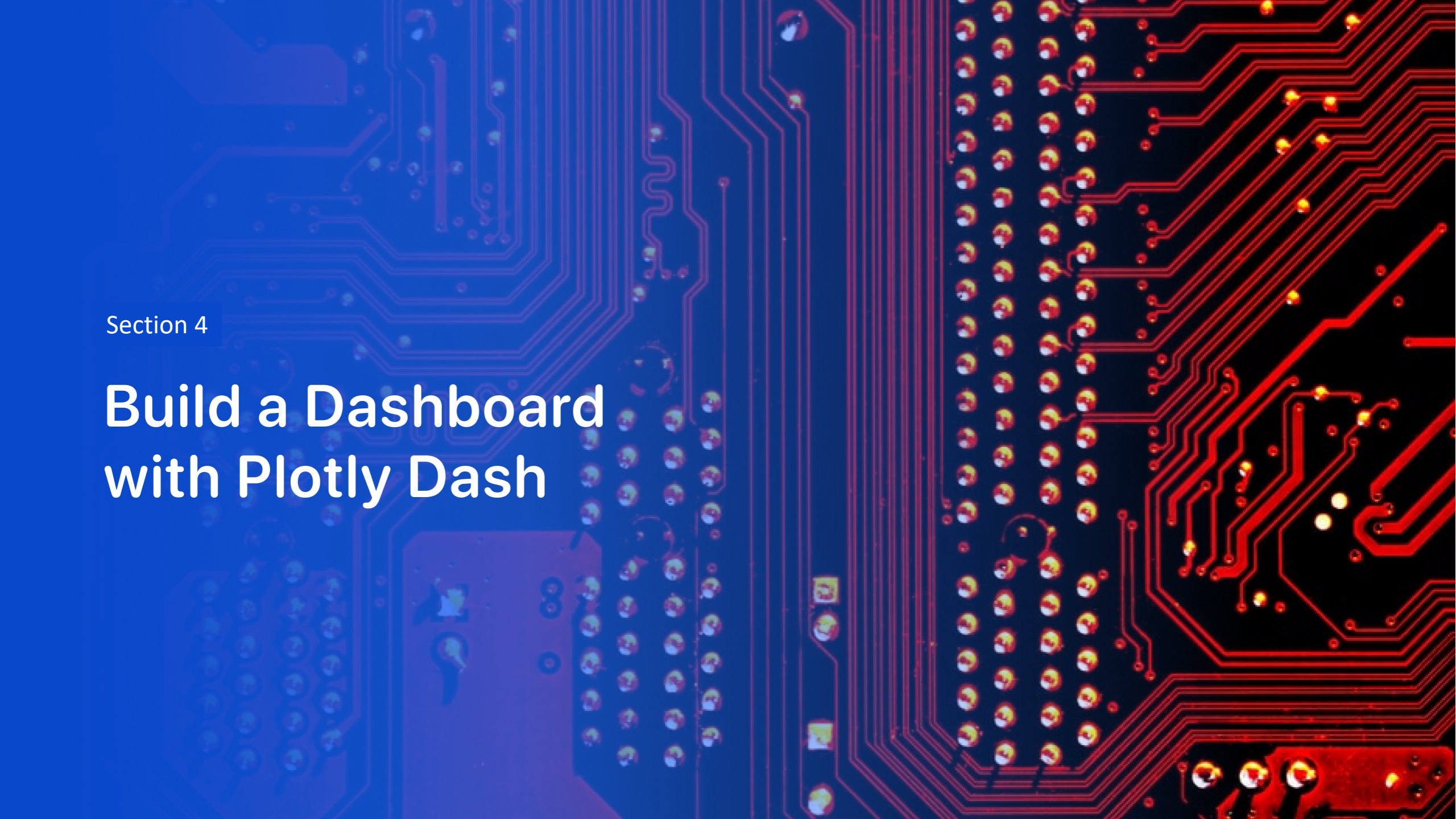
Proximity of launching sites

Proximity to the Ocean

The launch site is close to railways and highways for transportation of equipment or personnel, and is also close to coastline and relatively far from the cities so that launch failure does not pose a threat

Proximity to the cities

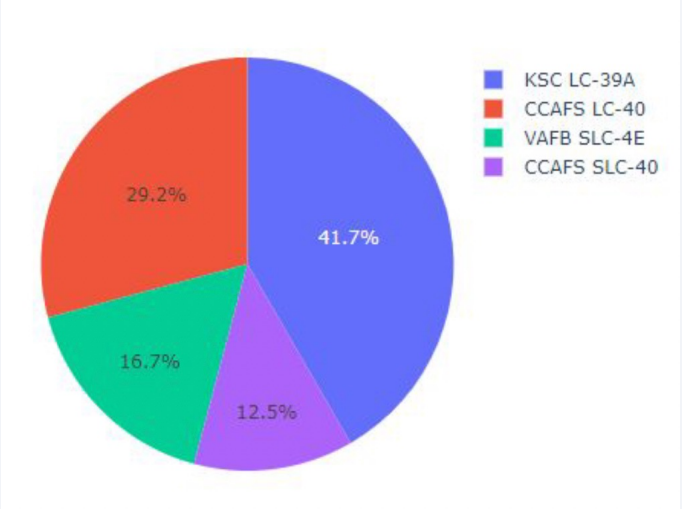



Section 4
Build a Dashboard with Plotly Dash

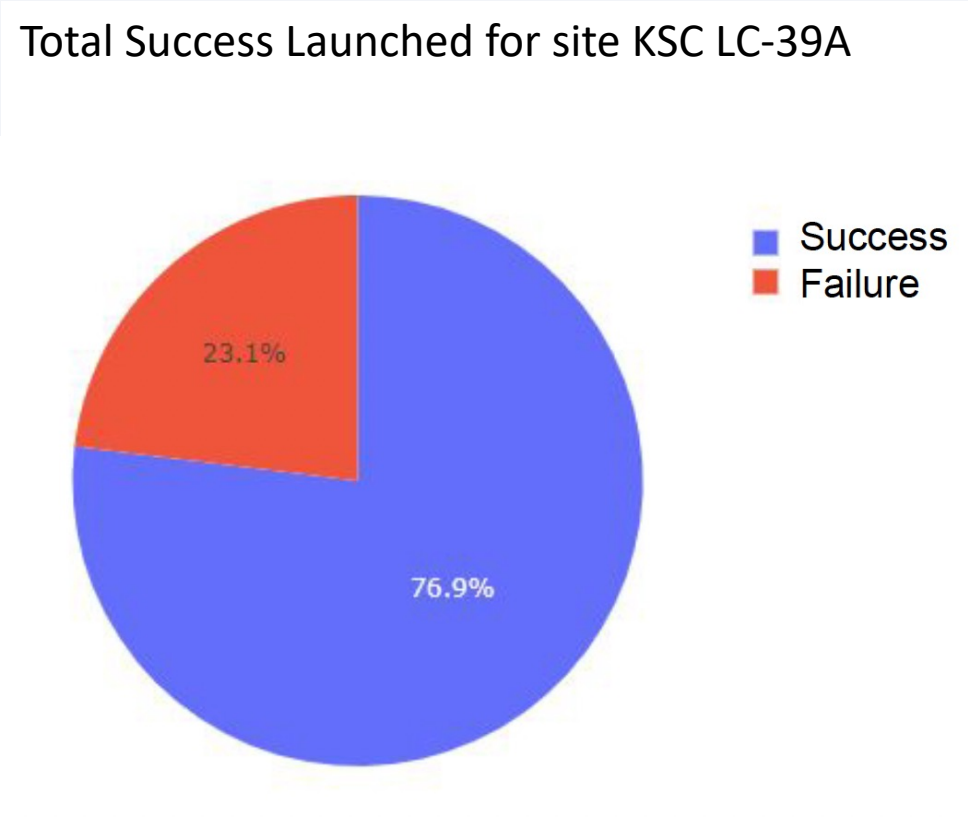
Launches by site

KSLC-39A records the most launch success among all sites.

• The VAFB SLC-4E has the fewest launch success, possibly because it is the only site located in California, so the launch difficulty on the west coast may be higher than on the east coast.

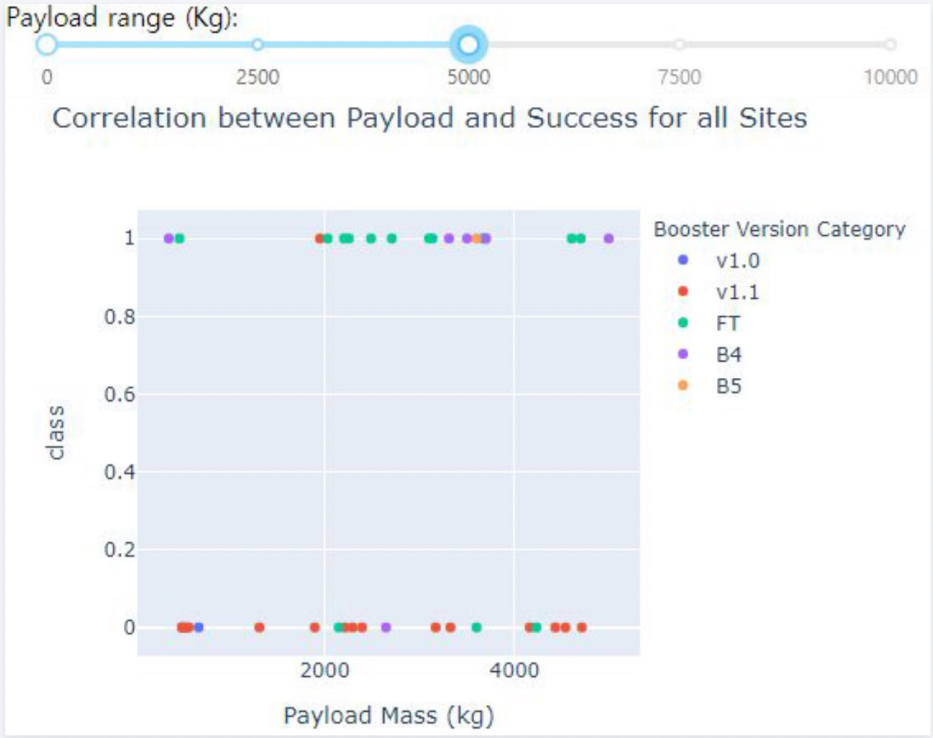


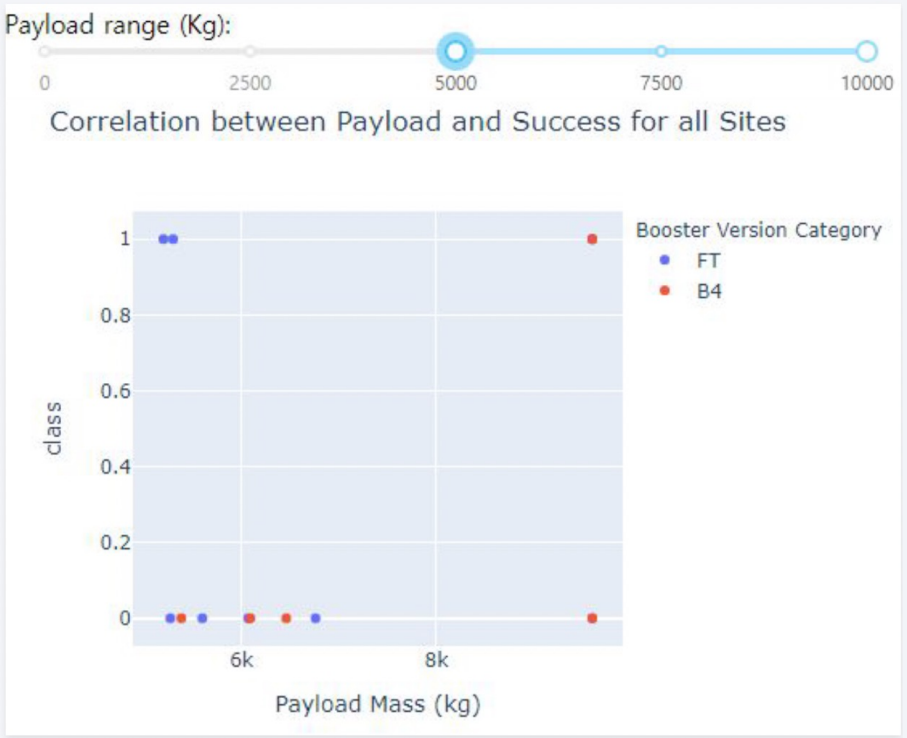
Launch site with highest launch success ratio



KSLC-39A has the highest success rate with 10 landing successes (76.9%) and 3 landing failures (23.1%).

Scatter plot: Payload vs Launch Outcomes



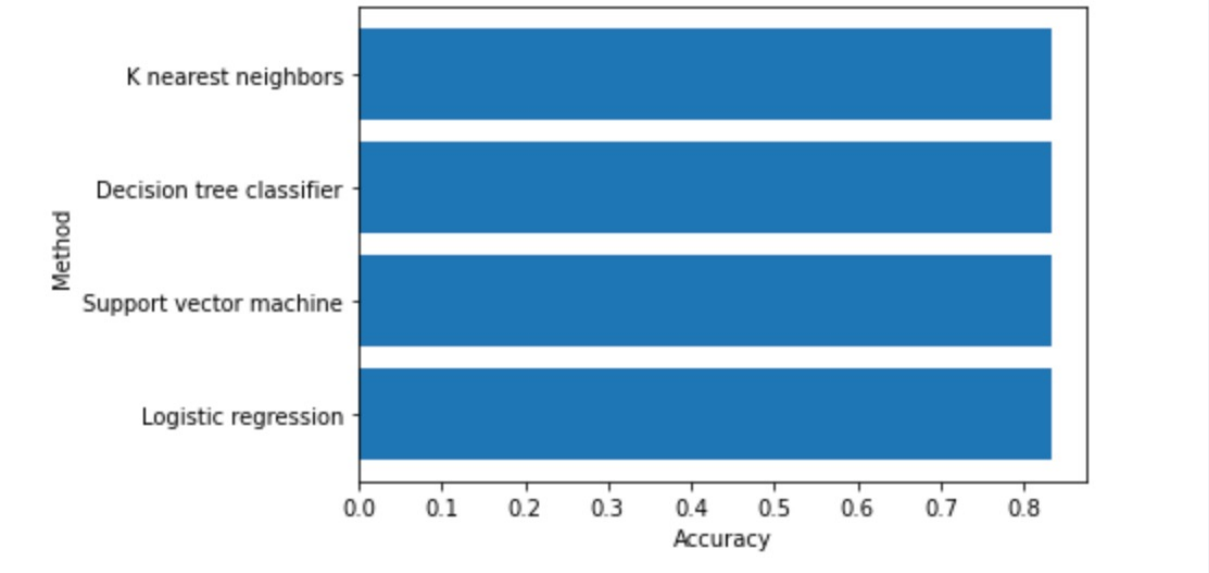


These figures show that the launch success rate (class 1) for low weighted payloads(0-5000 kg) is higher than that of heavy weighted payloads(5000-10000 kg).

Section 5

Predictive Analysis (Classification)

Classification Accuracy




```
Logistic regression 0.8333333333333334
Support vector machine 0.8333333333333334
Decision tree classifier 0.8333333333333334
K nearest neighbors 0.8333333333333334
```

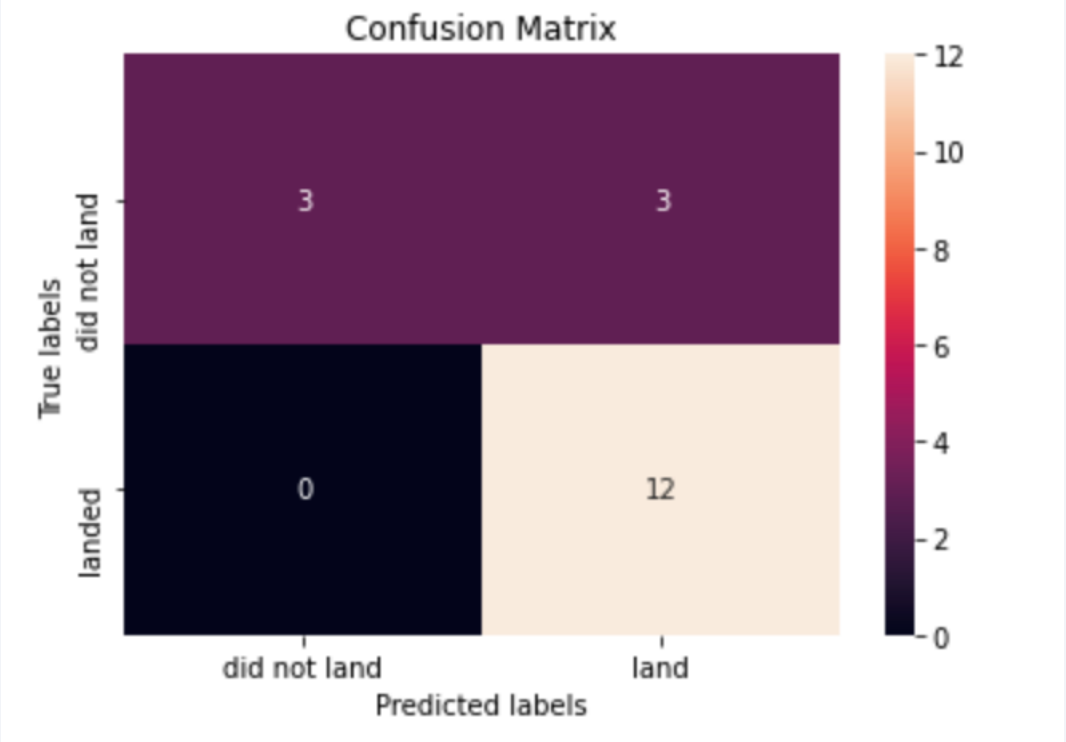
- In the test set, the accuracy of all models was virtually the same at 83.33%.
- Note that the test size was small at 18.
- Therefore, more data is needed to determine the optimal model.

Confusion Matrix

• The confusion matrix is the same for all models because all models performed the same for the test set.

• The models predicted 12 successful landings when the true label was successful and 3 failed landings when the true label was failure. But there were also 3 predictions that said successful landings when the true label was failure (false positive).

• Overall, these models predict successful landings.



Conclusions

- As the number of flights increased, the success rate increased, and recently it has exceeded 80%.
- Orbital types SSO, HEO, GEO, and ES-L1 have the highest success rate (100%).
- The launch site is close to railways, highways, and coastline, but far from cities.
- KSLC-39A has the highest number of launch successes and the highest success rate among all sites.
- The launch success rate of low weighted payloads is higher than that of heavy weighted payloads.
- In this dataset, all models have the same accuracy (83.33%), but it seems that more data is needed to determine the optimal model due to the small data size

Appendix

- Github

Thank you!