Winning Space Race with Data Science

Naiemuddin Ahmed
10.24.23

Outline

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

Executive Summary

Summary of Methodologies:

- Data Collection : The process was shown by using the SpaceX Rest API, along with the use of Web scraping
- Data Wrangling was performed in order to clean and prepare the data for further modification, this included handling missing values and data formatting
- Exploratory Data Analysis was carried out with the use of SQL queries and visualization libraries in Python
- Data Visualization: Plotly Dash & Folium maps were implemented to create Dashboards.
- Classification: scikit-learn package was implemented to perform predictive analysis'

Introduction

Project Insight

- SpaceX, an American spacecraft manufacturer, has revolutionized commercial space travel by offering relatively affordable prices, approximately 37% lower than competitors, for their Falcon 9 rockets priced at 62 million USD. This remarkable cost reduction is attributed to the successful reuse of the first launch stage. The ability to predict the first stage's successful landing and reusability is crucial in estimating launch costs for SpaceX. On the other hand, SpaceY emerges as a new player in the space tourism market, aiming to challenge SpaceX's dominance and explore opportunities in the rapidly evolving space industry.
- Significant Questions:
- Which parameters have an effect on successful landings during the first stage?
- How does the interaction between features have an impact on success or failure of landing rockets?
- What is a good fit when looking into classification models to predict the success of a landing or not?
- Rate of successful landings?

Section 1

Methodology

Methodology

Executive Summary

- Summary of results
- Data Collection and Data Wrangling created a specialized dataset.
- EDA identified the features to be used for the predictive analysis.
- Interactive Analytics Visualizations are shown with the use of screenshots.
- Predictive Analysis's with the use of Machine Learning techniques indicated the best classification model

- Data collection methodology:
- WebScraping → https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- SpaceX REST API → https://api.spacexdata.com/v4/rockets/

- Perform data wrangling → Filtering Data, Handling missing data, One Hot Coding for categories
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models → Building, Tuning, Evaluating using Regression, SVM, Decision Trees Classifier, KNN Classifier

Data Collection

Methods

1) SpaceX REST API

The data retrieved from the API came from the columns, the column headers are presented in the text below:

FlightNumber,Date,BoosterVersion,PayloadMass,Orbit,LaunchSite,Outcome,Flights,GridFins,Reused,Legs,LandingPad,Block,ReusedCount,Serial,Longitude,Latitude

2) Web Scraping SpaceX Website

The data retrieved from the website came from the columns, the column headers are presented in the text below:

Flight No., Launch site, Payload,PayloadMass, Orbit, Customer, Launchoutcome, VersionBooster, Booster landing, Date, Time

Data Collection – SpaceX API

- GITHUB Link: Data Collection

Request Response

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

```
response = requests.get(spacex_url)
```

Converting to json/df

```
# Use json_normalize meethod to convert the json result into a dataframe
data=pd.json_normalize(response.json())
```

Transform 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}
```

Then, we need to create a Pandas data frame from the dictionary launch_dict.

```
# Create a data from launch_dict
df=pd.DataFrame.from_dict(launch_dict)
```

```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9=df[df['BoosterVersion']!='Falcon 1']
```

Now that we have removed some values we should reset the FlgihtNumber column

```
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9
```

Filter & Export Data

Data Collection - Scraping

GITHUB Link: Webscraping

Request HTML response

Beautiful Soup

Extracting Columns

```
# use requests.get() method with the provided static_url
# assign the response to a object
response=requests.get(static_url)
data=response.text
```

```
# Use BeautifulSoup() to create a Beauti
soup=BeautifulSoup(data, 'html.parser')
```

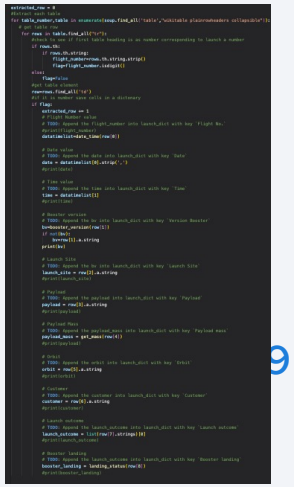
```
column_names = []

# Apply find_all() function with `th` element on
# Iterate each th element and apply the provided
# Append the Non-empty column name (`if name is n
for row in first_launch_table.find_all('th'):
    name=extract_column_from_header(row)
    if(name != None and len(name) > 0):
        column_names.append(name)
```

Parsing Data

DataFrame

```
df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })
```



Data Wrangling

Process:

- Create a label, 1 denoted for success, 0 denoted for fail
- Successful Outcomes: True ASDS,RTLS,OCEAN (Class 1)
- Unsuccessful Outcomes: False ASDS,RTLS,OCEAN (Class 0)
- Unknown Outcomes: None None, None ASDS (Class 0)

Tasks Performed:

1) Initial EDA to understand dataset
2) Calculation of the launches
3) Creation of Categorical training label

GITHUB LINK: Data Wrangling

GITHUB LINK: [Data Wrangling]

Site Launces

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

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

Launces per orbit type

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

```
GTO     27
ISS     21
VLEO    14
PO       9
LEO      7
SSO      5
MEO      3
ES-L1    1
HEO      1
SO       1
GEO      1
Name: Orbit, dtype: int64
```

Mission Outcome

```
# 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
```

Landing Outcome

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
landing_class = []
for outcome in df['Outcome']:
    if outcome in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
```

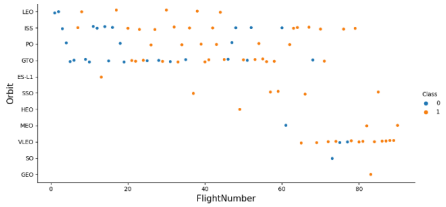
```
df['Class']=landing_class
df[['Class']].head(8)
```

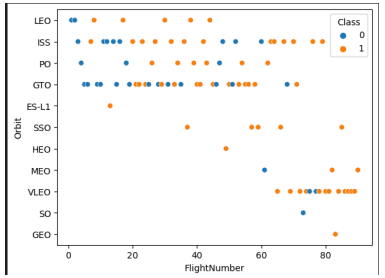
CSV

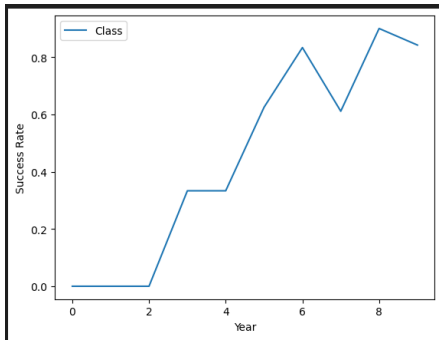
```
df.to_csv("dataset_part_2.csv", index=False)
```

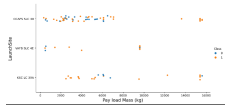
Explanatory Data Analysis with Data Visualization

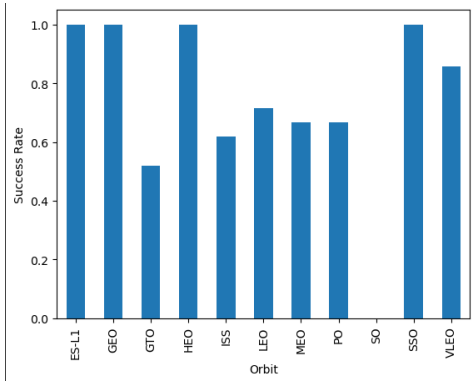
- Scatter Plot → This plot is used to show how much a variable is affected d by a secondary variable. The relation between the two variables is a correlation.
  - Flight Number vs Launch Site
  - Flight Number vs Orbit type
  - Payload vs Launch Site
  - Payload vs Orbit type
- Bar Plot → Used to simplify and compare multiple groups of data, the two axis' represent a category and a discrete value. In general used to identify a relation.
  - Orbit Type vs Success Rate
- Line Graph → used to show a trend that can help with predictions
  - Year vs Success Rate



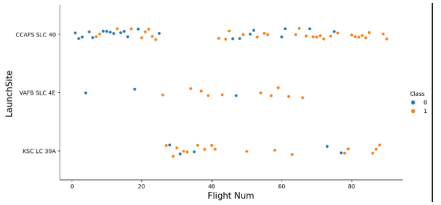








EDA Visualizations



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 ' C A '

• Displaying the total payload m a s s carried by boosters launched by NASA (CRS) O

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 boosterversions which have carried the maximum payload mass

• Listing the failed landing_ outcomes ni 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

Building Map with Folium

- Objects:
  - Markers (Launch Sites)
  - Markers for Launches (Success/Fail)
  - Lines as Distances

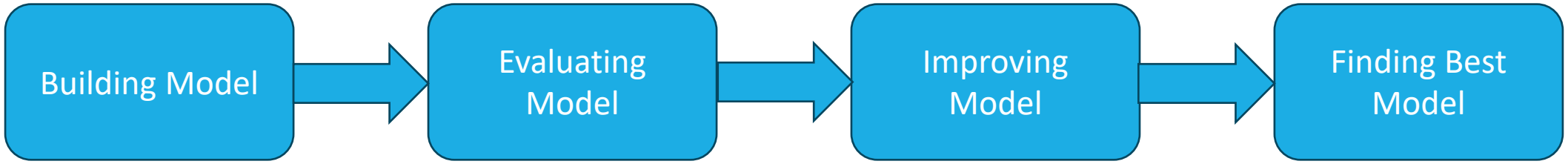
- Patterns:

- 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

Dashboard with Plotly Dash

- The dashboard application contains a pie chart and a scatter point chart. • Pie chart
- For showing total success launches by sites
- 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
- For showing the relationship between Outcomes and Payload mass (Kg) by different boosters
- Has 2 inputs: Al sites/individual site & Payload mass on a slider between 0 and 10000 kg
- This chart helps determine how success depends on the launch point, payload mass, and booster version categories.

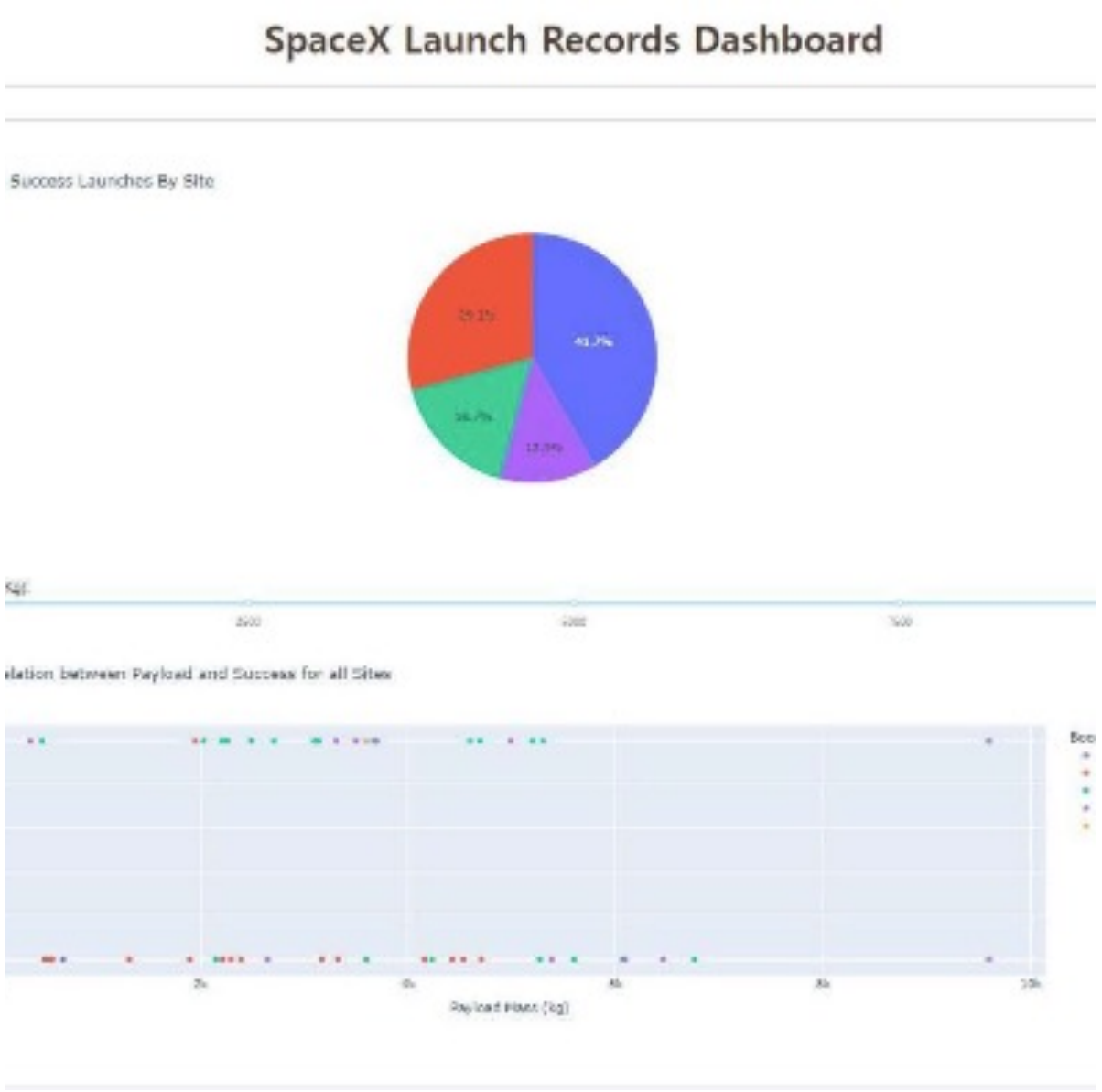
Classification – Predictive Analysis

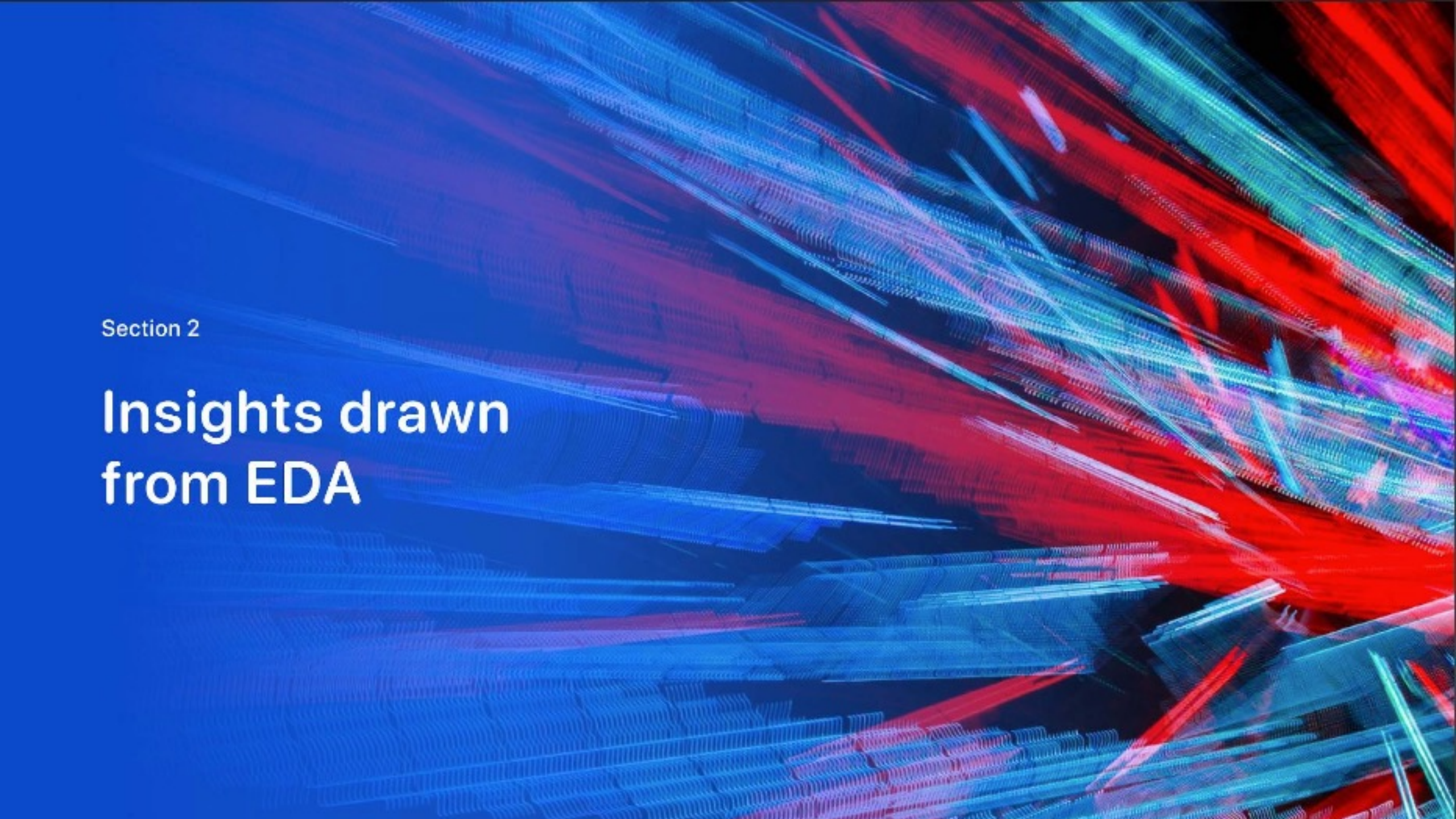


Perform exploratory Data Analysis
and determine Training Labels
- Create a column for the class
- Standardize the data
- Split into training data and test data

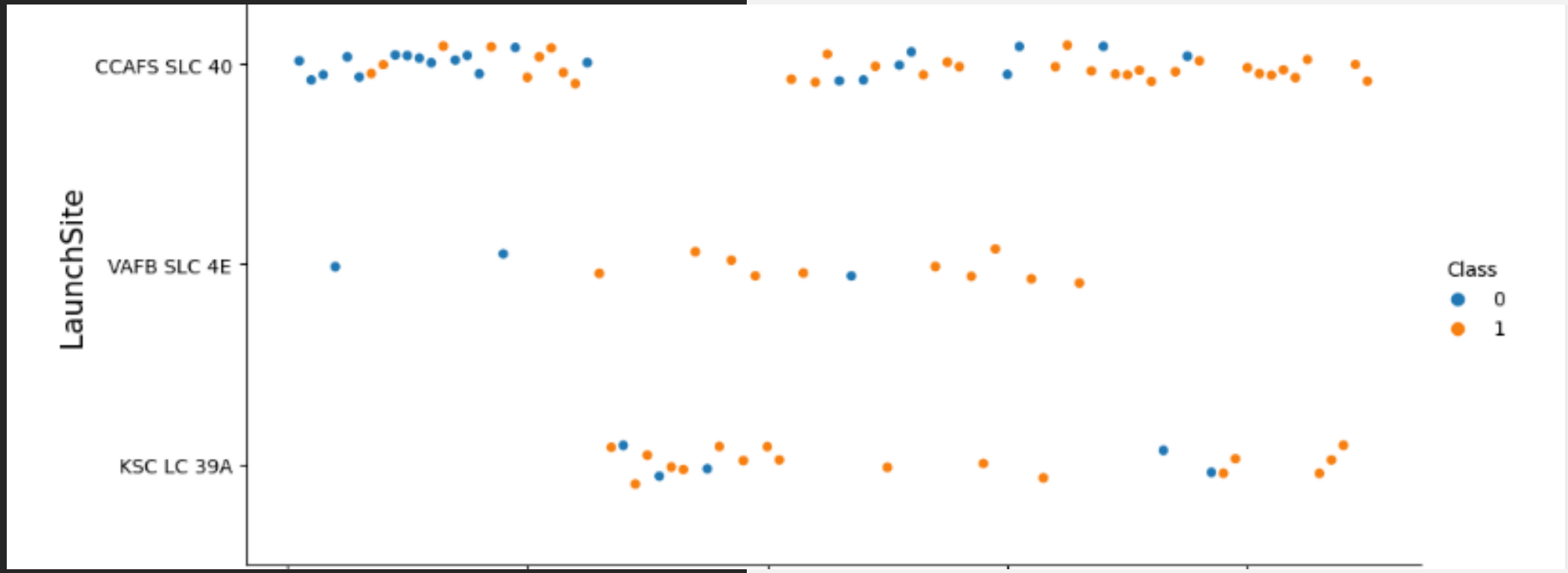
Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
- Find the method performs best using test data

The first picture shows a sneak peek of the Plotly Dash Dashboard. In the next slides, you'll see the results of different analyses: looking at data with graphs, using SQL, creating an interactive map with Folium, and a lively Interactive Dashboard. When we check how accurate these methods are, an interesting thing pops up. No matter which method we use—whether it's looking at graphs, using SQL, playing with maps, or exploring the Interactive Dashboard—they all give us about 83% accuracy for the test data.



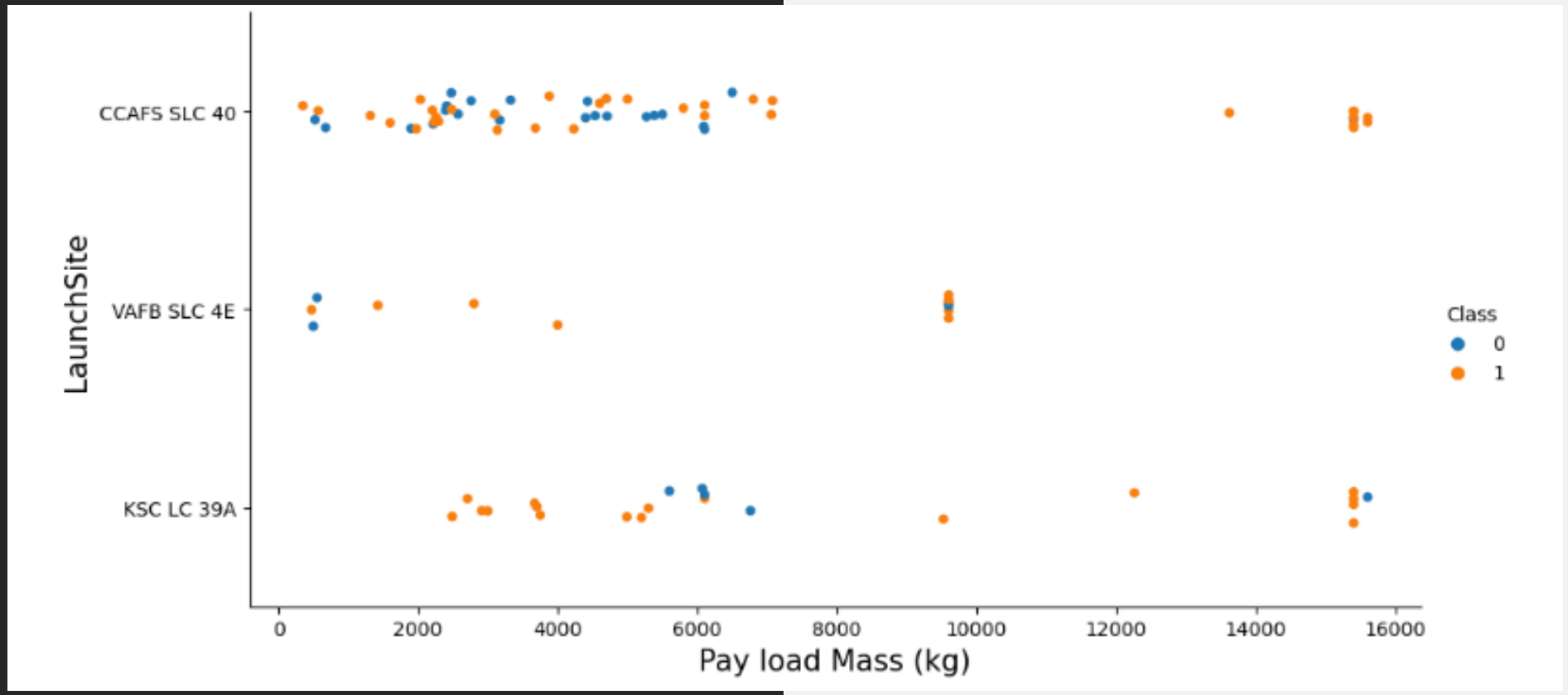

Section 2
Insights drawn
from EDA

Flight Number vs Launch Site



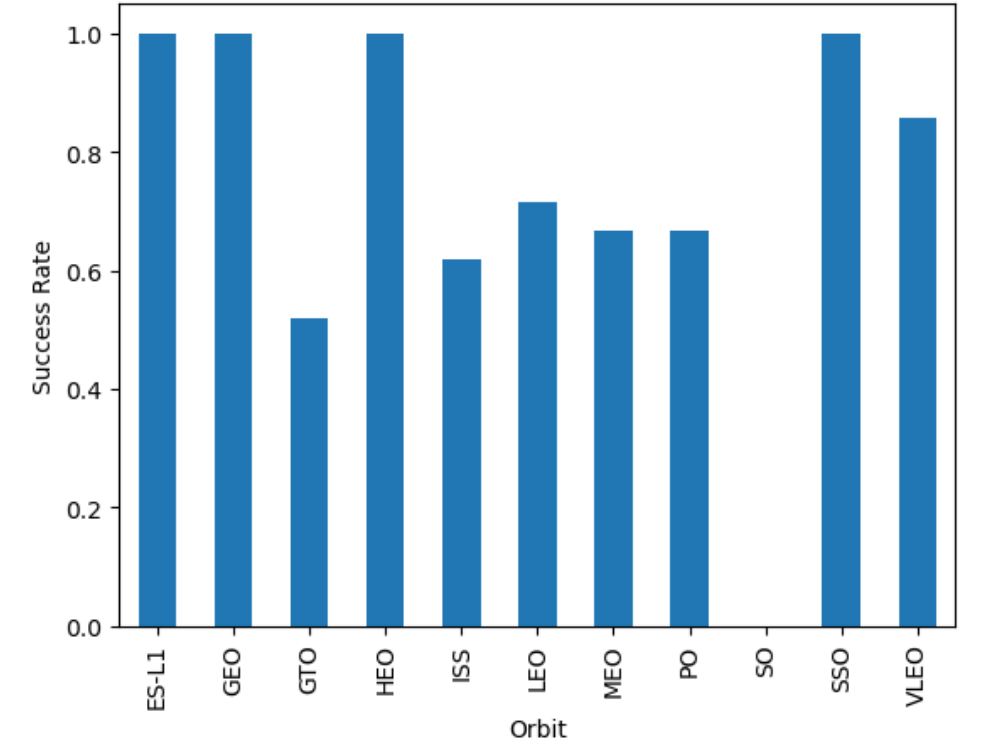
- The blue color in Class 0 indicates unsuccessful launches, while the orange color in Class 1 represents successful launches.
- This visual demonstrates a rising success rate corresponding to the increasing number of flights.
- Notably, the success rate has seen a significant uptick starting from the 20th flight, marking a substantial breakthrough at this juncture.

Payload vs Launch Site



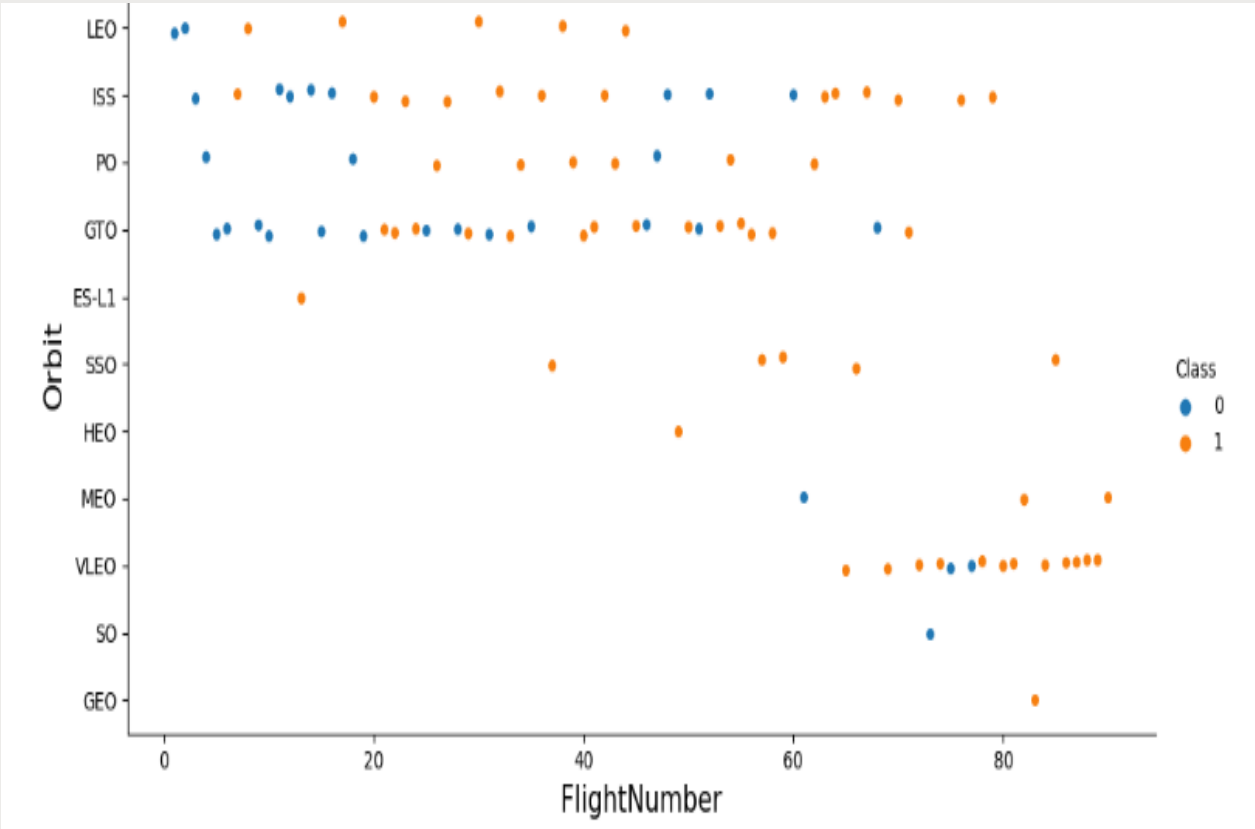
•• Class 0 (blue) denotes unsuccessful launches, while Class 1 (orange) signifies successful ones. Upon initial observation, it appears that a higher payload mass is associated with a greater rocket success rate. However, interpreting decisions from this figure proves challenging due to the absence of a discernible pattern linking successful launches to payload mass.

Success Rate vs Orbit Type



•• Orbit types SSO, HEO, GEO, and ES-L1 boast perfect success rates of 100%. • In contrast, the success rate for the GTO orbit type is only 50%, ranking as the lowest except for the SO type, which experienced a single failure attempt.

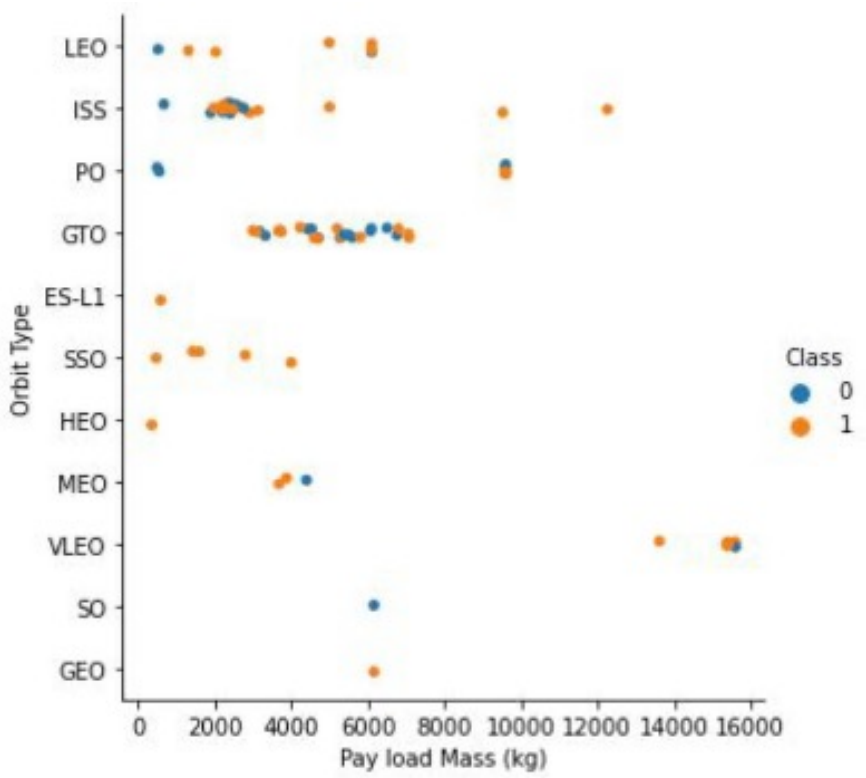
Flight Number vs Orbit Type



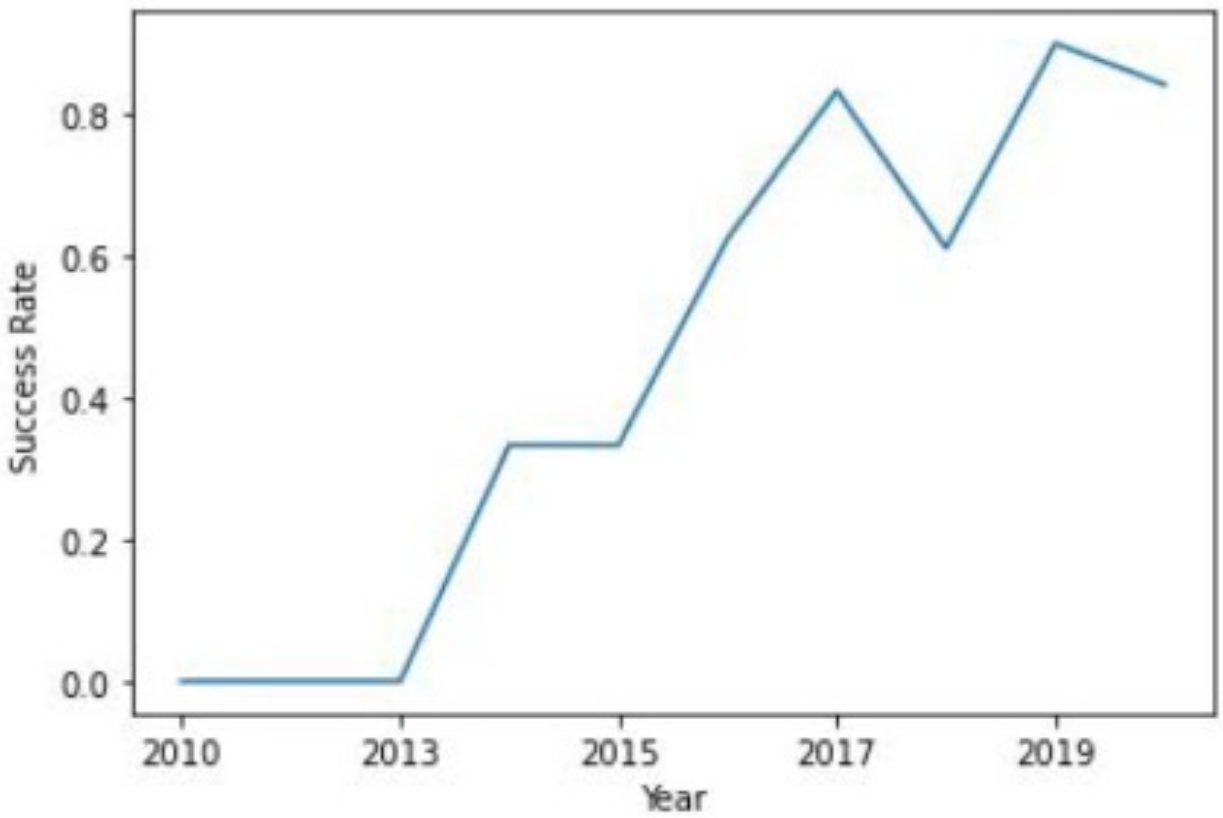
- Class 0 (depicted in blue) signifies an unsuccessful launch, while Class 1 (depicted in orange) denotes a successful launch.
- Generally, there appears to be a correlation between launch outcomes and the flight number.
- However, when considering the GTO orbit, there is no apparent relationship between flight numbers and success rates.
- The journey commences with LEO, showing a moderate success rate, while VLEO, with its notably high success rate, emerges as the frequently chosen option in recent launches.

Payload vs Orbit Type

- Class 0 (depicted in blue) signifies an unsuccessful launch, while Class 1 (depicted in orange) denotes a successful launch.
- Notably, when dealing with heavy payloads, the likelihood of a successful or positive landing is higher for LEO and ISS.
- Conversely, in the context of GTO, discerning between positive and negative landing rates is challenging as they are all closely clustered together.



Launch Success Yearly Trend



- From 2013-2017 the success rate has been increasing
- Slight decrease around 2018
- Roughly 80% success rate

Section 3

EDA with SQL

Launch Site Names

- When the SQL DISTINCT clause is employed within the query, only distinct values appear in the LaunchSite column from the Space table. • There are four distinct launch sites: CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E.

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
```

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

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

Launch Site Names Beginning with ”CAA”

- Only five records of the SpaceX table were displayed using LIMIT 5
- clause in the query.
- • Using the LIKE operator and the
- percent sign (%) together, the Launch S i t e name starting with
- CAA could be called.

```
%%sql SELECT * FROM SPACEXTABLE
      WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Ou
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	S
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	S
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	S
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	S
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	S

```
%%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS
      FROM SPACEXTABLE
      WHERE Customer LIKE 'NASA (CRS)'
```

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

TOTAL_PAYLOAD_MASS
45596

Total Payload Mass

- Sum() used to calculate Payload mass
- Where clause filters the data to fit specifications

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

```
%%sql SELECT AVG(PAYLOAD_MASS__KG_) as AVERAGE_PAYLOAD_MASS
      FROM SPACEXTABLE
      WHERE Booster_Version = 'F9 v1.1'
```

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

AVERAGE_PAYLOAD_MASS
2928.4

Avg Payload Mass

- AVG() used to calculate Payload mass
- Where clause filters the data to fit specifications

```
%%sql SELECT MIN(Date) AS First_Succ_Landing_Date
      FROM SPACEXTABLE
      WHERE Landing_Outcome = 'Success (ground pad)'
```

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

First_Succ_Landing_Date
2015-12-22

First Landing (Successful)

- Min(Date) used to calculate the date
- Where clause filters the data to fit specifications

Successful Drone Ship Landing with Payload between 4000 and 6000

- Where specifies drone ship success landing
- AND statements create a range

```
]: %%sql SELECT Booster_Version
         FROM SPACEXTABLE
         WHERE Landing_Outcome = 'Success (drone ship)'
           AND PAYLOAD_MASS__KG_ > 4000
           AND PAYLOAD_MASS__KG_ < 6000;
 * sqlite:///my_data1.db
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

- Count(*) counts all columns
- Where filters data so that it is data that is only not null
- Groupby groups rows that have same values

List the total number of successful and failure mission outcomes

```
%%sql SELECT Mission_Outcome, COUNT(*) AS Outcome_Count
      FROM SPACEXTABLE
      WHERE Mission_Outcome IS NOT NULL
      GROUP BY Mission_Outcome
```

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

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

Boosters Carried Maximum Payload

Using a subquery, first, find the maximum value of
the payload by using MAX() function, and second, filter the dataset to perform a search if
PAYLOAD MASS KG is the maximum value of the
payload.

- According to the result, version F9 B5 B10xx.× boosters could carried the maximum payload.

```
%%sql SELECT Booster_Version
        FROM SPACEXTABLE
        WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
```

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

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

2015 Launch Records

- In the WHERE clause, filter the dataset to perform a search if Landing_outcome is Failure (drone ship).
- Using the AND operator to display a record if additional condition YEAR is 2015.
- In 2015, there were two landing failures on drone ships.

```
%%sql  SELECT
          SUBSTR(Date, 6, 2) AS Month,
          Booster_Version,
          Launch_Site,
          Landing_Outcome
       FROM
          SPACEXTABLE
       WHERE
          SUBSTR(Date, 0, 5) = '2015'
          AND Landing_Outcome LIKE 'Failure%'
          AND Landing_Outcome LIKE '%drone ship%';
```

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

Month	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- In the WHERE clause, filter the dataset to perform a search if the date is between 2010-06-04 and 2017-03-20.
- • Using the ORDER BY keyword to sort the records by total number of landing, and using DESC keyword to sort the records in descending order.
- • According to the results, the number of successes and failures between 2010-06-04 and 2017-03-20 was similar.

```
%%sql
SELECT
    Landing_Outcome,
    COUNT(*) AS Outcome_Count
FROM
    SPACEXTABLE
WHERE
    Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY
    Landing_Outcome
ORDER BY
    Outcome_Count DESC;
```

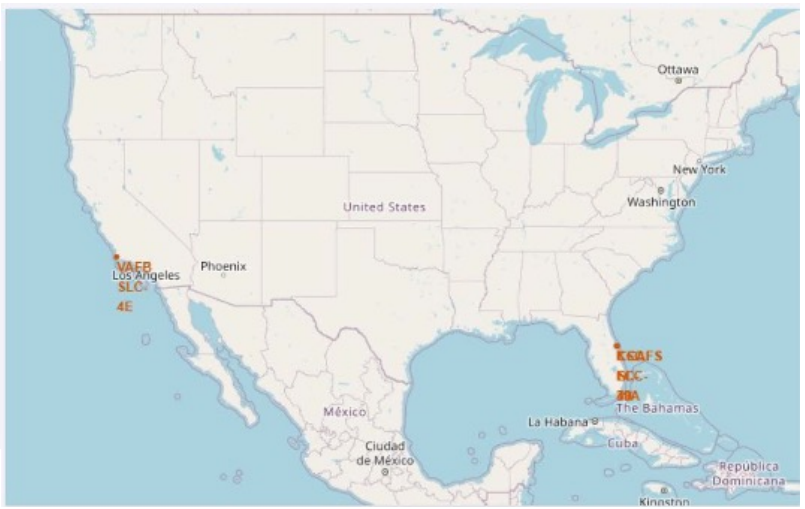
```
 * sqlite:///my_data1.db
Done.
```

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

Launch Sites Locations

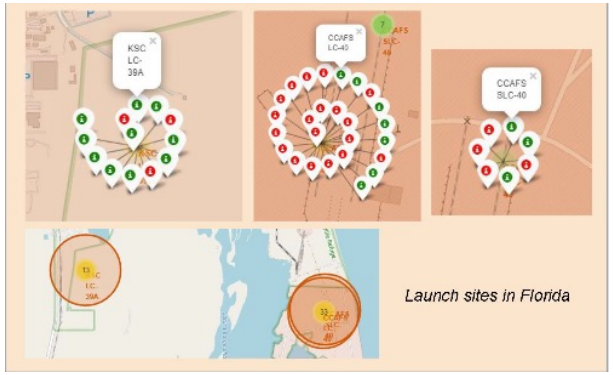
- The left map shows all *SpaceX* launch sites, and the right map also shows that all launch sites are in the United States.
- As can be seen on the map, all launch sites are near the coast.



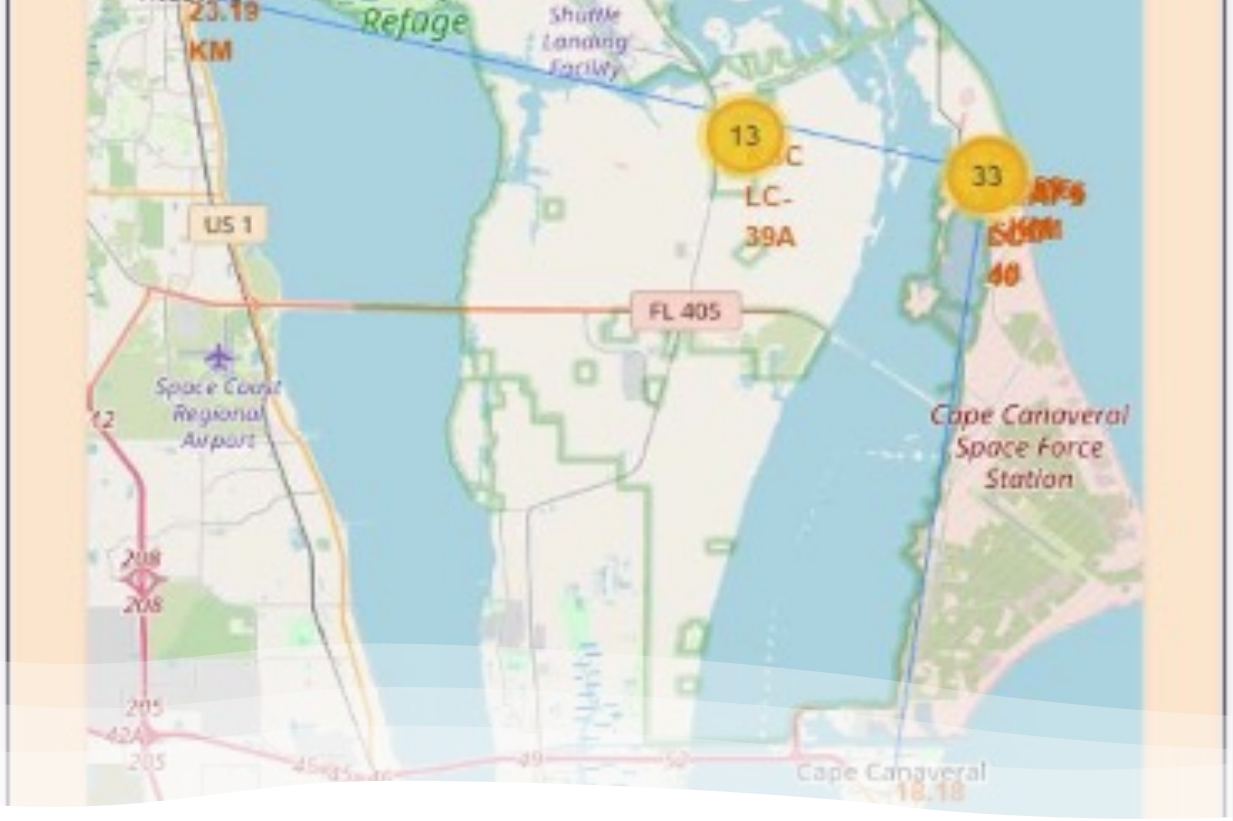
Color Coordinated Launch Outcome

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

Launch site in California

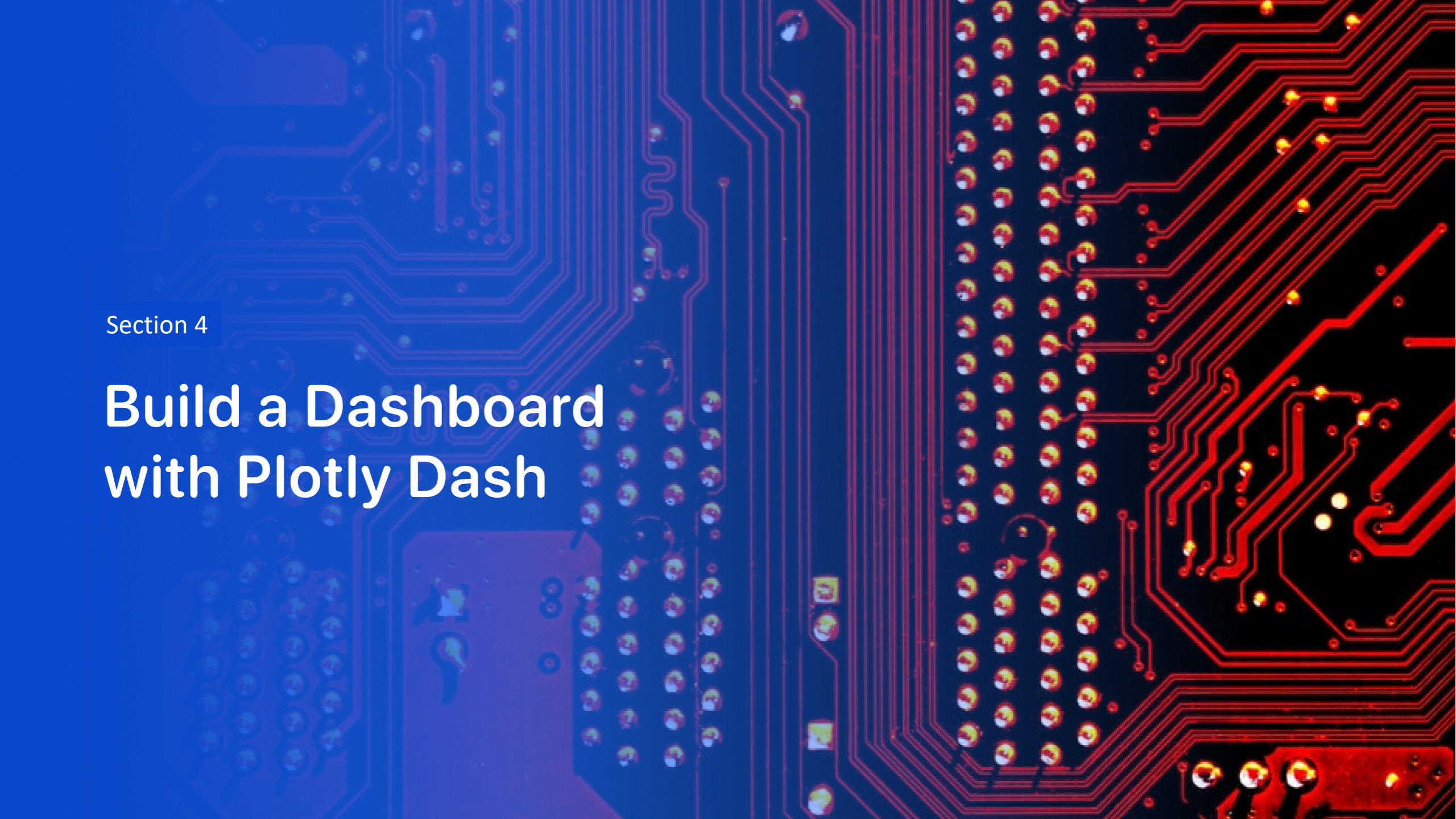


Launch sites in Florida

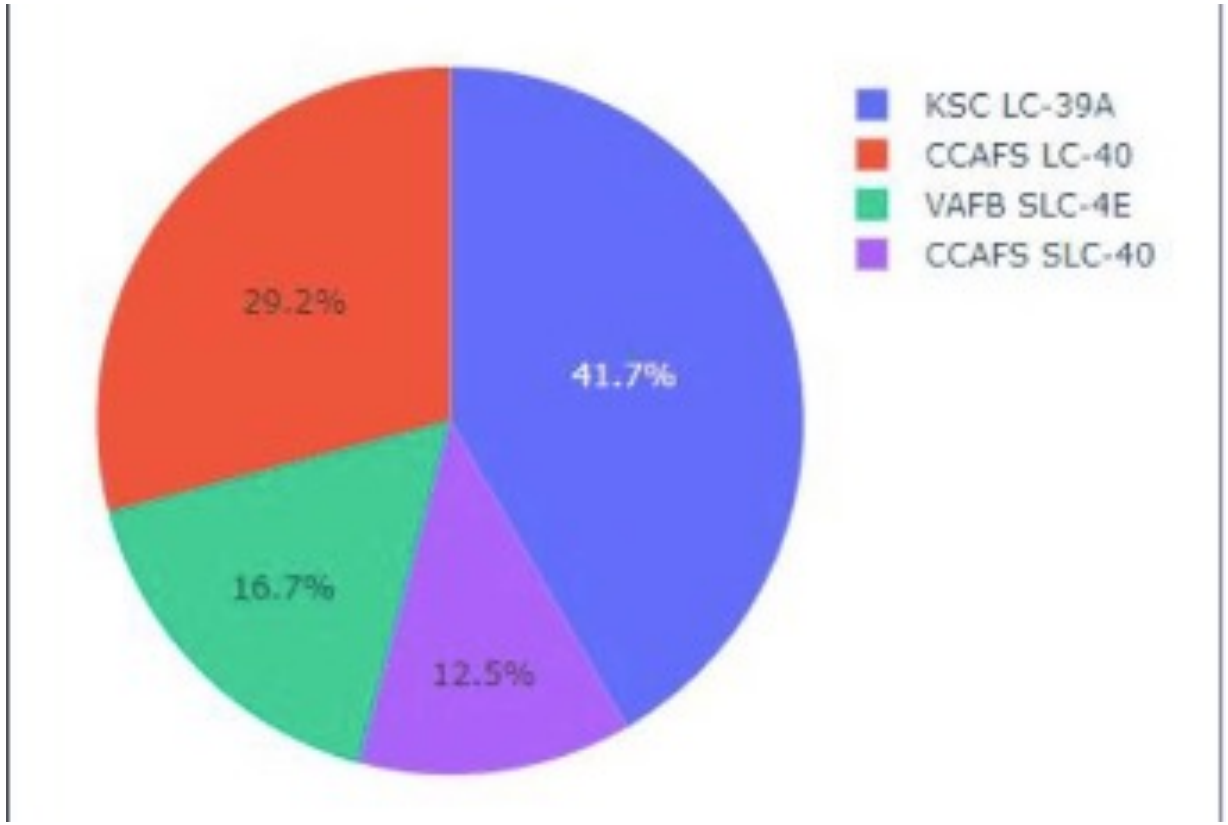


Proximities of Launch Sites

The launch site is strategically situated in proximity to railways and highways for efficient transportation of equipment and personnel. Additionally, it is strategically positioned near coastlines and at a considerable distance from urban areas to mitigate the potential threat of launch failures.


Section 4
Build a Dashboard with Plotly Dash

Total Success Launches

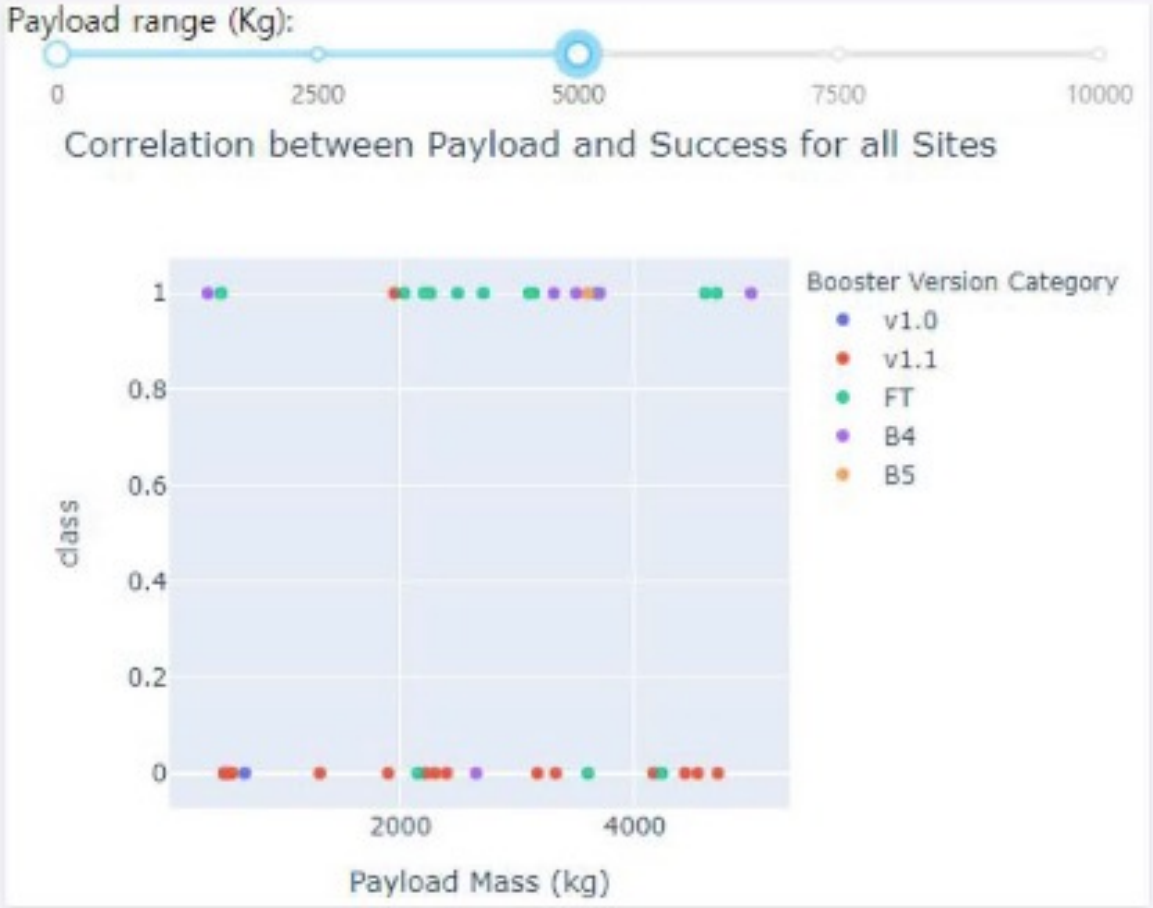


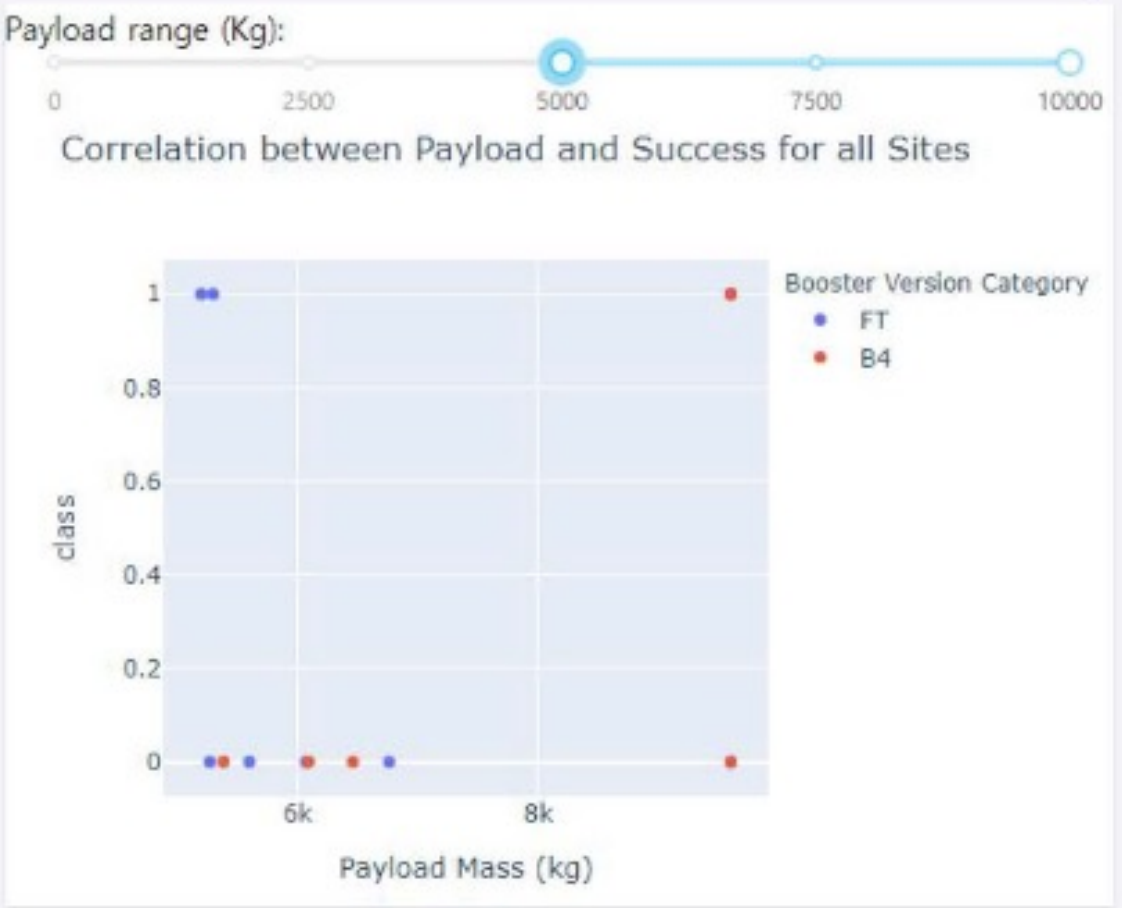
- KSLC-39A records the most launch success among all sites.
- The VAFB SLC-4E has the fewest launch success, possibly because
  - the data sample is small, or
  - because it is the only site located in California, so the launch difficulty on the west coast may be higher than on the east coast.

Highest Launch Success Ratio

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

Payload vs. Launch Outcome Scatter Plot for all sites



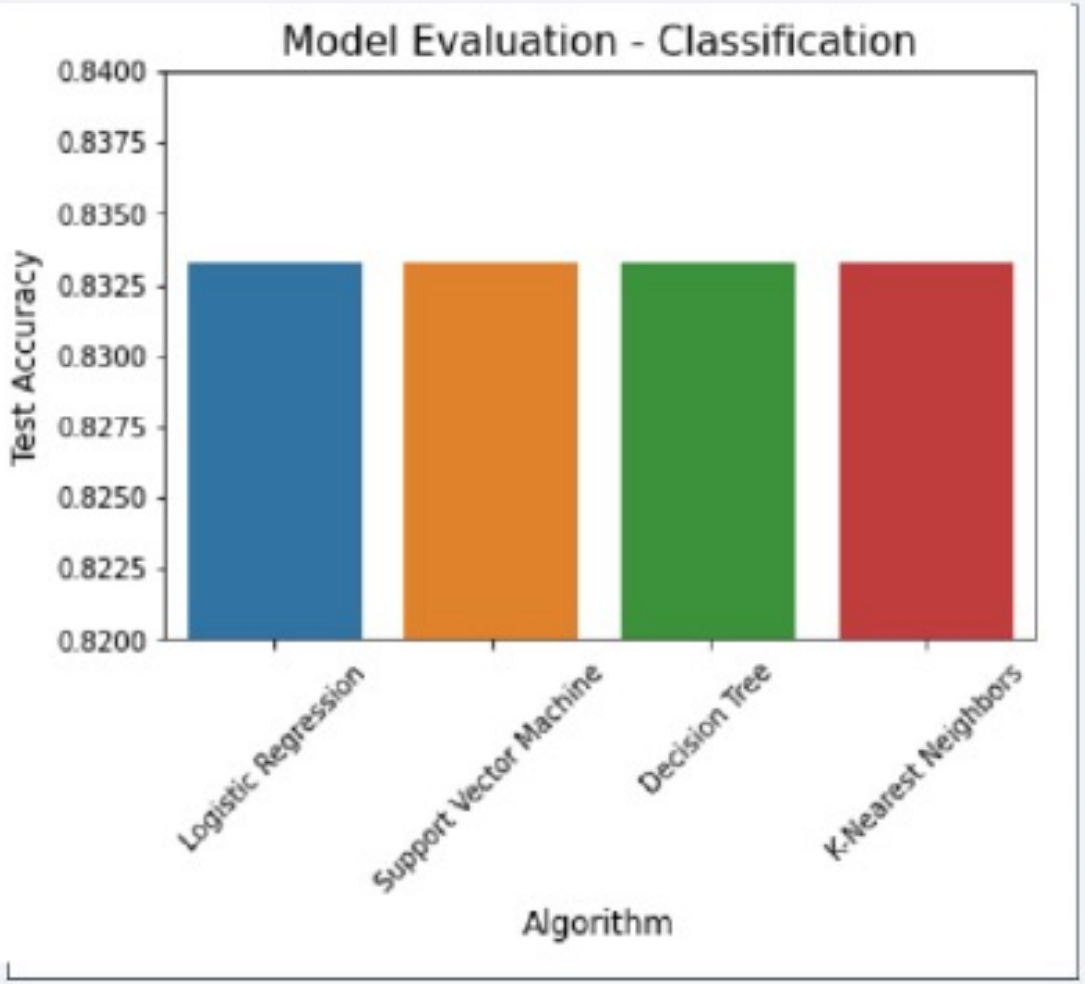


- These figures show that **the launch success rate** *(class 1)* **for low weighted payloads***(0-5000 kg)* **is higher** than that of heavy weighted payloads*(5000-10000 kg)*.

Section 5

Predictive Analysis (Classification)

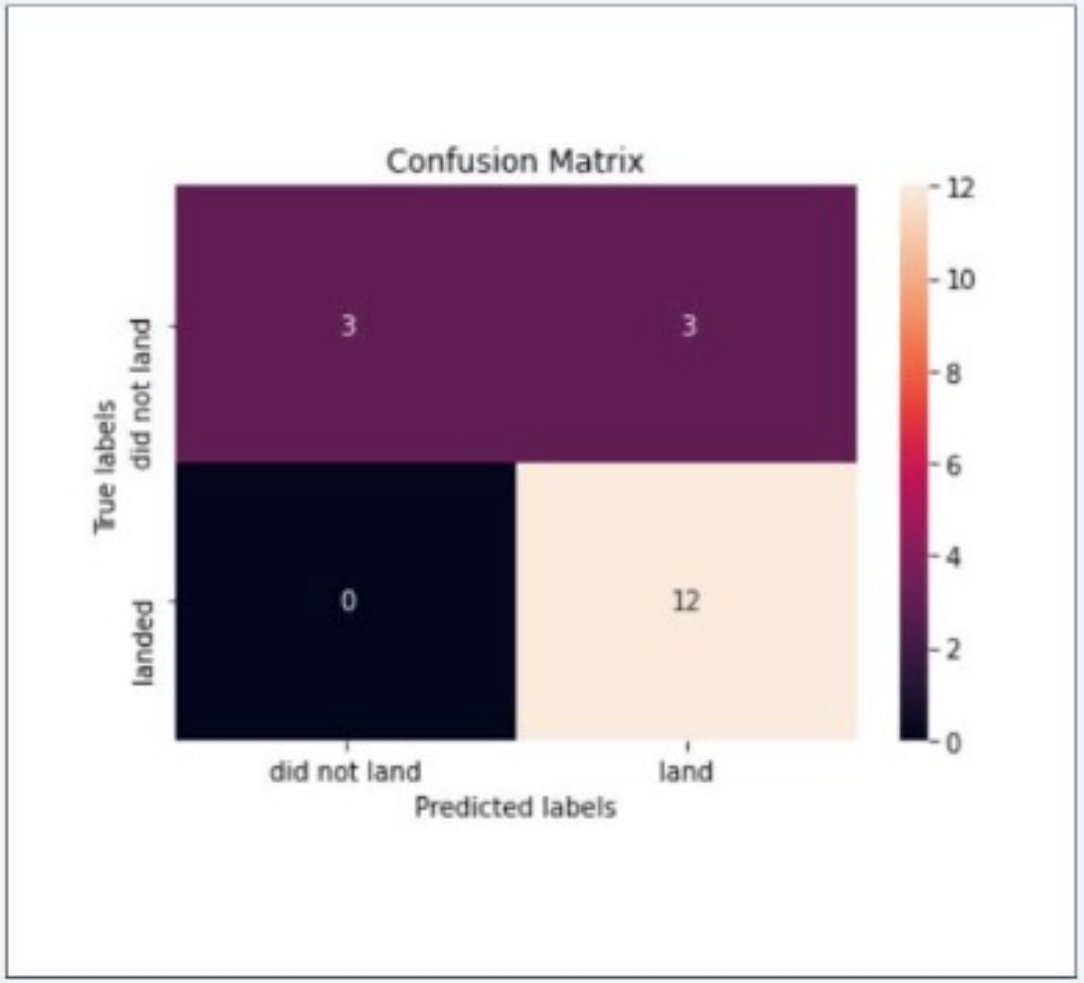
Classification Accuracy



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

	Algorithm	Test Accuracy	Training Score
0	Logistic Regression	0.833333	0.846429
1	Support Vector Machine	0.833333	0.848214
2	Decision Tree	0.833333	0.875000
3	K-Nearest Neighbors	0.833333	0.848214

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

• With the increasing number of flights, the success rate has risen, recently surpassing 80%. • Orbital types SSO, HEO, GEO, and ES-L1 boast a flawless success rate of 100%. • The launch site is strategically positioned near railways, highways, and coastlines while maintaining a considerable distance from urban areas. • KSLC-39A stands out with the highest number of launch successes and the highest success rate among all sites. • Launch success rates for low-weighted payloads exceed those of heavy-weighted payloads. • Across the dataset, all models showcase identical accuracy at 83.33%, suggesting that more data is needed to determine the optimal model due to the dataset's limited size.

Thank you!