IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

- Data Science Capstone Project

Abdallah Fathy
December, 2023

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 Plotly Dash
  - ❑ Predictive analysis (Classification)
- **Summary of all results**
  - ❑ Exploratory data analysis results
  - ❑ Interactive analytics demo in screenshots
  - ❑ Predictive analysis results

Introduction

- **Project background and context**

The era of commercial space has arrived, and there are several companies that are making space travel affordable for everyone. Perhaps the most successful of them is *SpaceX*, and one of the reasons is that their rocket launch is relatively inexpensive.

SpaceX advertises *Falcon 9* rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because *SpaceX*
can reuse the first stage.

Therefore, we will predict if the *Falcon 9* first stage will land successfully. If we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against *SpaceX* for a rocket 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:
  - SpaceX API & Web Scrapping [Falcon 9 and Falcon Heavy launched Records from Wikipedia]
- Perform data wrangling
  - Convert outcomes into Training Labels with the booster successfully/unsuccessful landed
- 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 Hyperparameter for SVM, Classification Trees and Logistic Regression

Data Collection

- The data collection process includes a combination of API requests from the

SpaceX API and web scraping data from a table in the Wikipedia page of

SpaceX, Falcon 9 and Falcon Heavy Launches Records.

  - ❑ SpaceX API Data Columns: `FlightNumber, Date, BoosterVersion,PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused,Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude`
  - ❑ `Wikipedia Web Scrape Data Columns: Flight No., Launch site, Payload,PayloadMass, Orbit, Customer, Launch outcome, Version Booster,Booster landing, Date, Time`

Space X API

Request data from SpaceX API → API returns SpaceX data in .JSON file → Normalize data into CSV file

Web Scrapping

Get HTML Response from Wikipedia page → Extract data using beautiful soup → Normalize data into CSV file

Data Collection – SpaceX API

1.Requesting rocket launch data from Space X API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
0.0s
```

```
response = requests.get(spacex_url)
response
1.1s
```

2.Converting Response to a JSON file

```
json_data = response.json()
data = json_normalize(json_data)
```

3.Using custom functions to clean data

```
# Call getCoreData
getCoreData(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getBoosterVersion
getBoosterVersion(data)
```

4.Combining the columns into a dictionary to create data frame

```
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(launch_dict)
```

5.Filtering dataframe and exporting to a CSV

```
data_falcon9 = df[df['BoosterVersion'] != 'Falcon 1']
```

```
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

1.Getting response from HTML

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

2.Creating a BeautifulSoup object

```
soup = BeautifulSoup(response.text, 'html')
```

3.Finding all tables and assigning the result to a list

```
html_tables = soup.find_all('table')
```

4.Extracting column name one by one

```
column_names = []

for th in first_launch_table.find_all('th'):
    column_name = extract_column_from_header(th)
    if column_name is not None and len(column_name) > 0:
        column_names.append(column_name)
```

5.Creating an empty dictionary with keys

```
launch_dict= dict.fromkeys(column_names)
# Remove an irrelvant column
del launch_dict['Date and time ( )']
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

6.Filling up the launch_dict with launch records

(Too long to put in here, so please refer to the notebook)

7.Creating a Dataframe and exporting it to a CSV

```
df = pd.DataFrame(launch_data_list)
```

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Data Wrangling

- There are several cases in which the booster failed to successfully land on the dataset, and sometimes it attempted to land but failed because of accident.
  - ❑ `True Ocean`: the mission result has successfully landed in a specific area of the ocean
  - ❑ `False Ocean`: the mission result has not successfully landed in a specific area of the ocean
  - ❑ `True RTLS`: the mission result successfully landed on the ground pad
  - ❑ `False RTLS`: the mission result has not successfully landed on the ground pad
  - ❑ `True ASDS`: the mission result has successfully landed on the drone ship
  - ❑ `False ASDS`: the mission result has not landed on the drone ship
- Converting these results into training labels:
  - ❑ `1` = successful / `0` = failure

Data Wrangling

1.Calculating the number of launches at each site

```
launch_site_count = df['LaunchSite'].value_counts()
```

2.Calculating the number and occurrence of each orbit

```
orbit_count = df['Orbit'].value_counts()
```

3.Calculating the number and occurrence of mission outcome per orbit type

```
landing_outcomes = df['Outcome'].value_counts()
```

4.Creating a landing outcome label from Outcome column

```
landing_class = [0 if outcome in bad_outcomes else 1 for outcome in df['Outcome']]
```

```
df['Class']=landing_class
```

5.Calculating the success rate for every landing in dataset

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

```
0.6666666666666666
```

6.Exporting dataset to a CSV

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

EDA with Data Visualization

- **Scatter Chart:**
  - ❑ Flight Number VS. Launch Site.
  - ❑ Payload VS. Launch Site.
  - ❑ Flight Number VS. Orbit Type.
  - ❑ Payload Vs. Orbit Type.
  - ❑ A Scatter plot shows how much one variables is affected by another. The relation between two variables is called a correlation. This plot is generally composed of large data bodies
- **Bar Chart:**
  - ❑ Orbit Type VS. Success Rate.
  - ❑ 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.
- **Line Char:**
  - ❑ Year VS. Success Rate.
  - ❑ A Line chart shows data variables and trends very clearly and helps predict the result of data that hasn't yet been recorded.

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 (groundpad)) between the date 2010-06-04 and 2017-03-20, in descending order

Build an Interactive Map with Folium

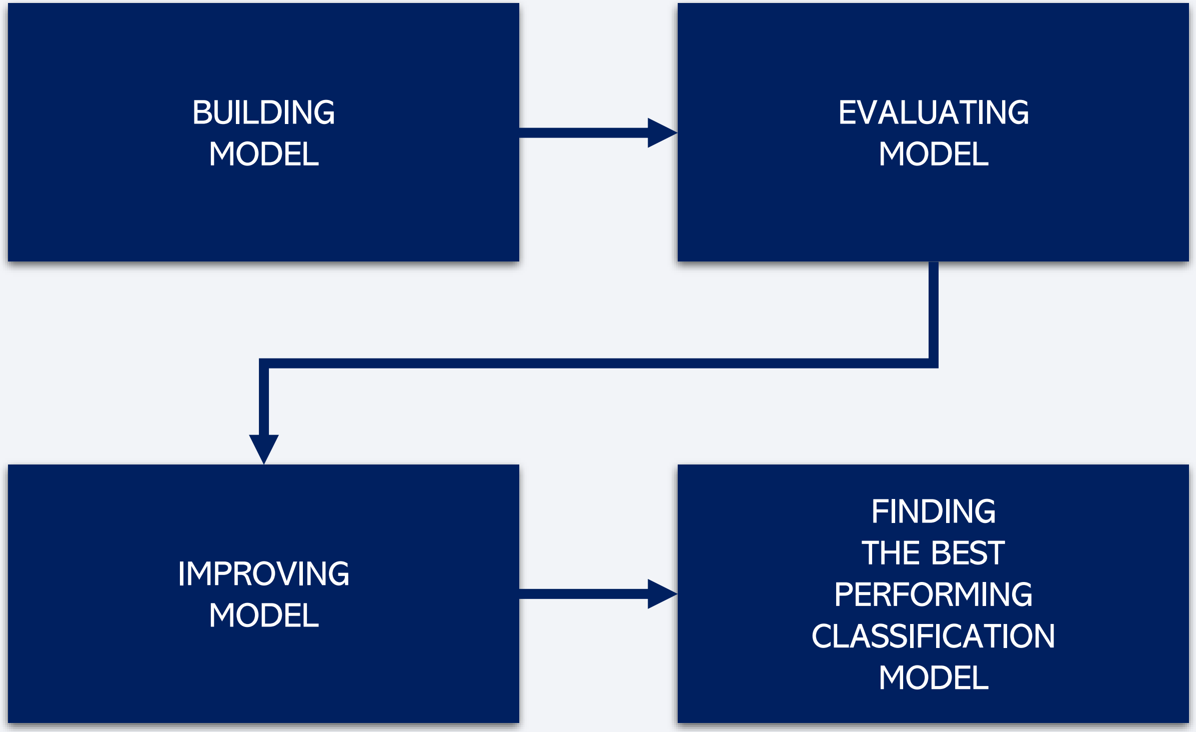
- Objects created and added to a folium map:
  - ❑ Markers that show all launch sites on a map
  - ❑ Markers that show the success/failed launches for each site on the map
  - ❑ Lines that show the distances between a launch site to its proximities
- By adding these objects, following geographical patterns 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

Build a 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: All 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.

Predictive Analysis (Classification)

- **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



Results

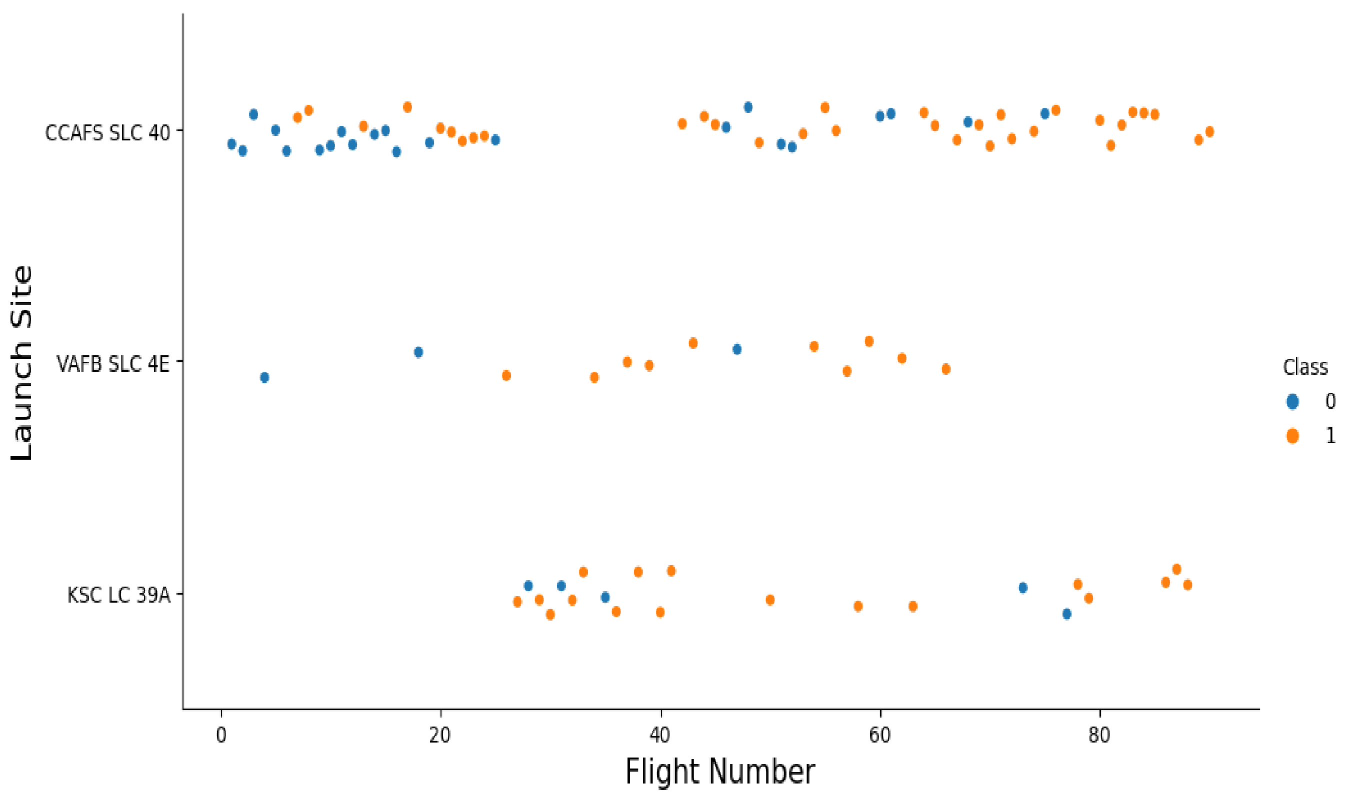
SpaceX Launch Records Dashboard

- The left screenshot is a preview of the Dashboard with Plotly Dash.
- The results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and Interactive Dashboard will be shown in the next slides.
- Comparing the accuracy of the four methods, all return the same accuracy of about 83% for test data.

Section 2

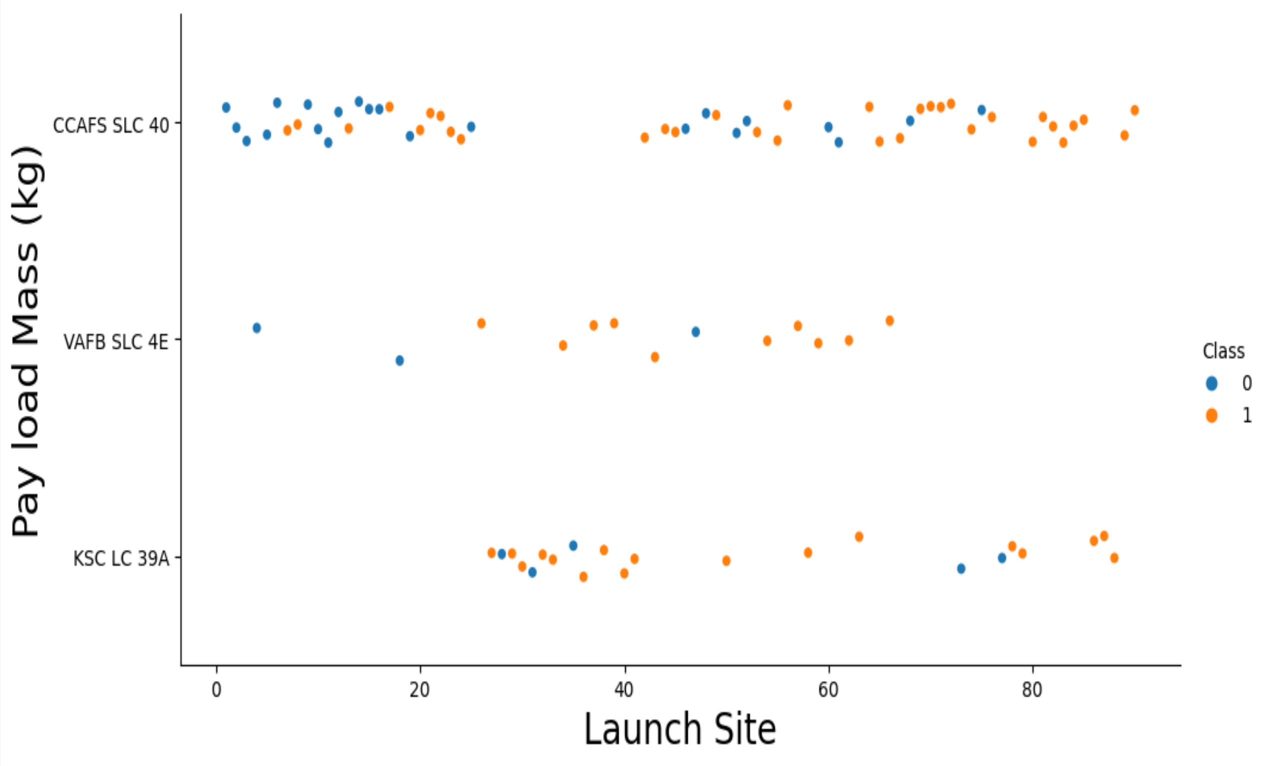
Insights drawn from EDA

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.**
- As the success rate has increased considerably since the 20th flight, this point seems to be a big breakthrough.

Payload vs. Launch Site

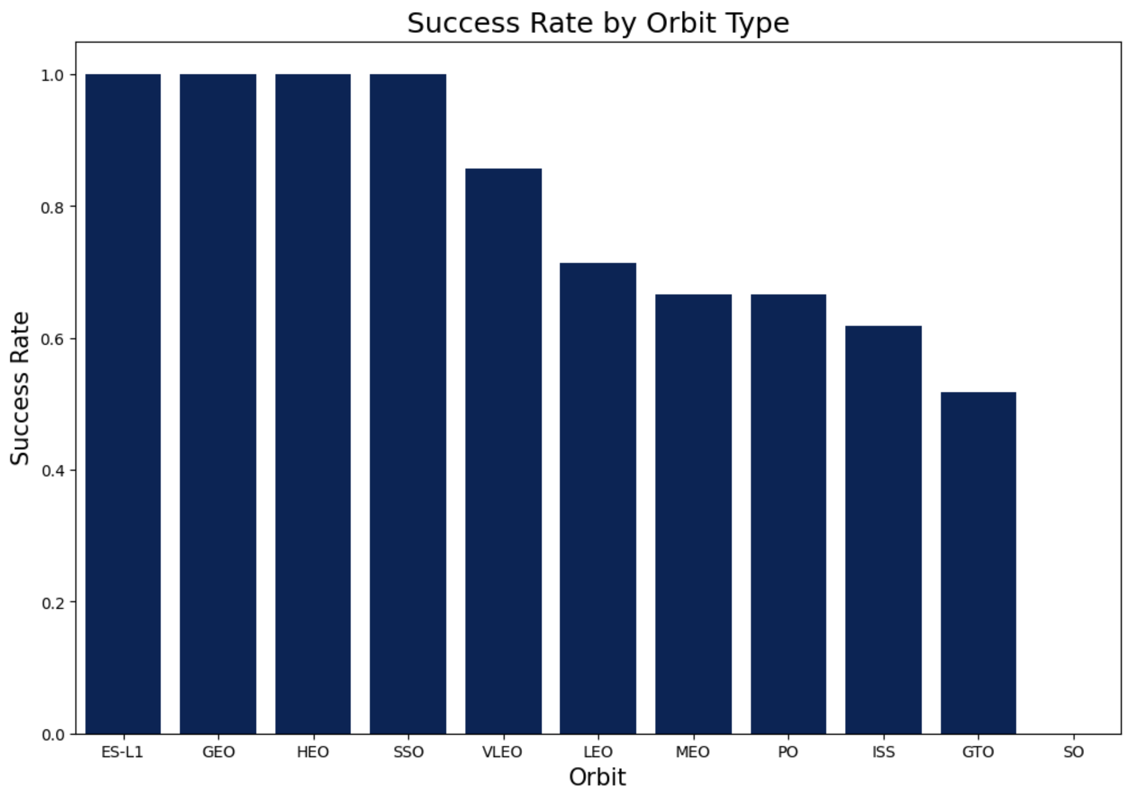


- Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.
- At first glance, the larger pay load mass, the higher the rocket's success rate, but it seems difficult to make decisions based on this figure because **no clear pattern can be found between successful launch and Pay Load Mass.**

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

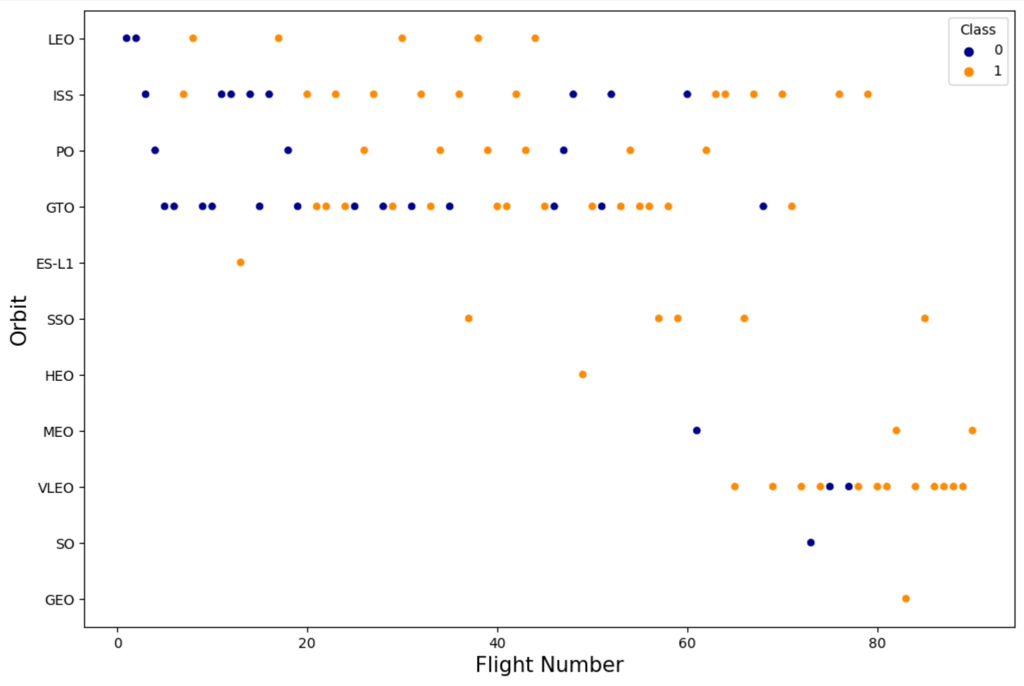
- Show the screenshot of the scatter plot with explanations



Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type

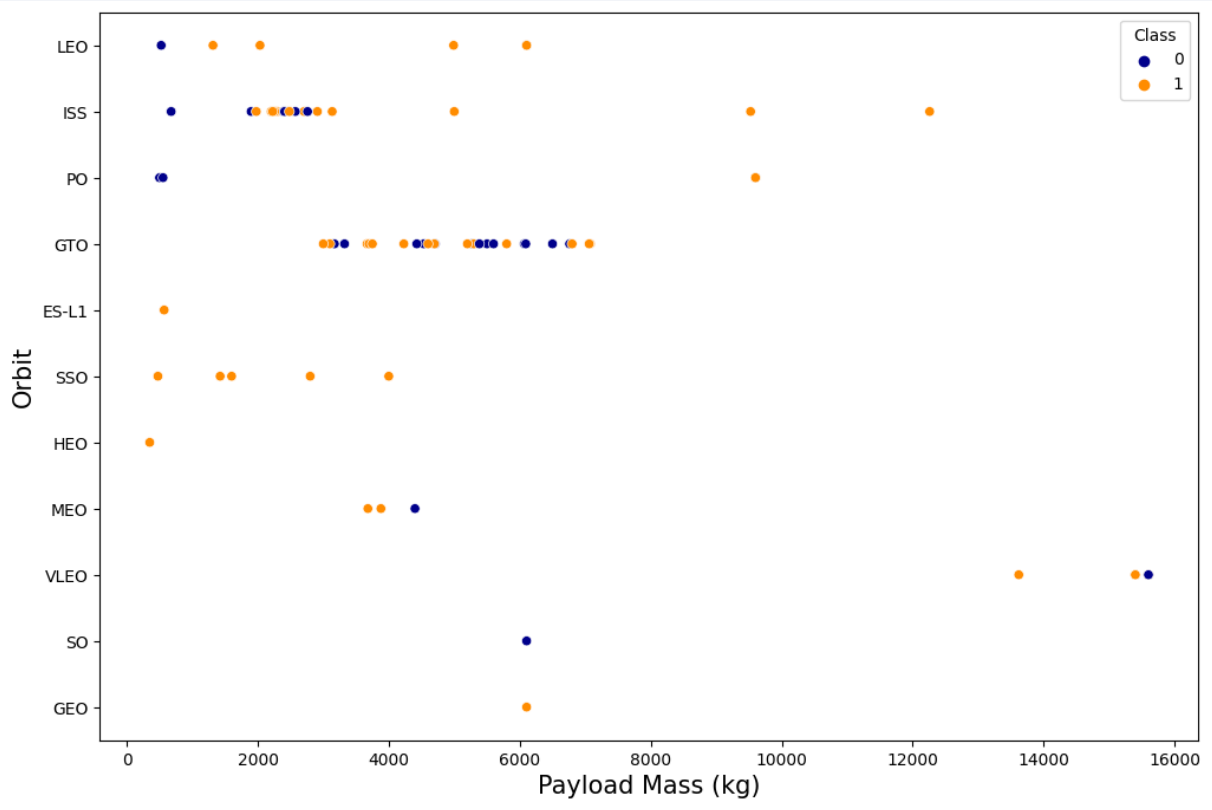
- Show the screenshot of the scatter plot with explanations



Payload vs. Orbit Type

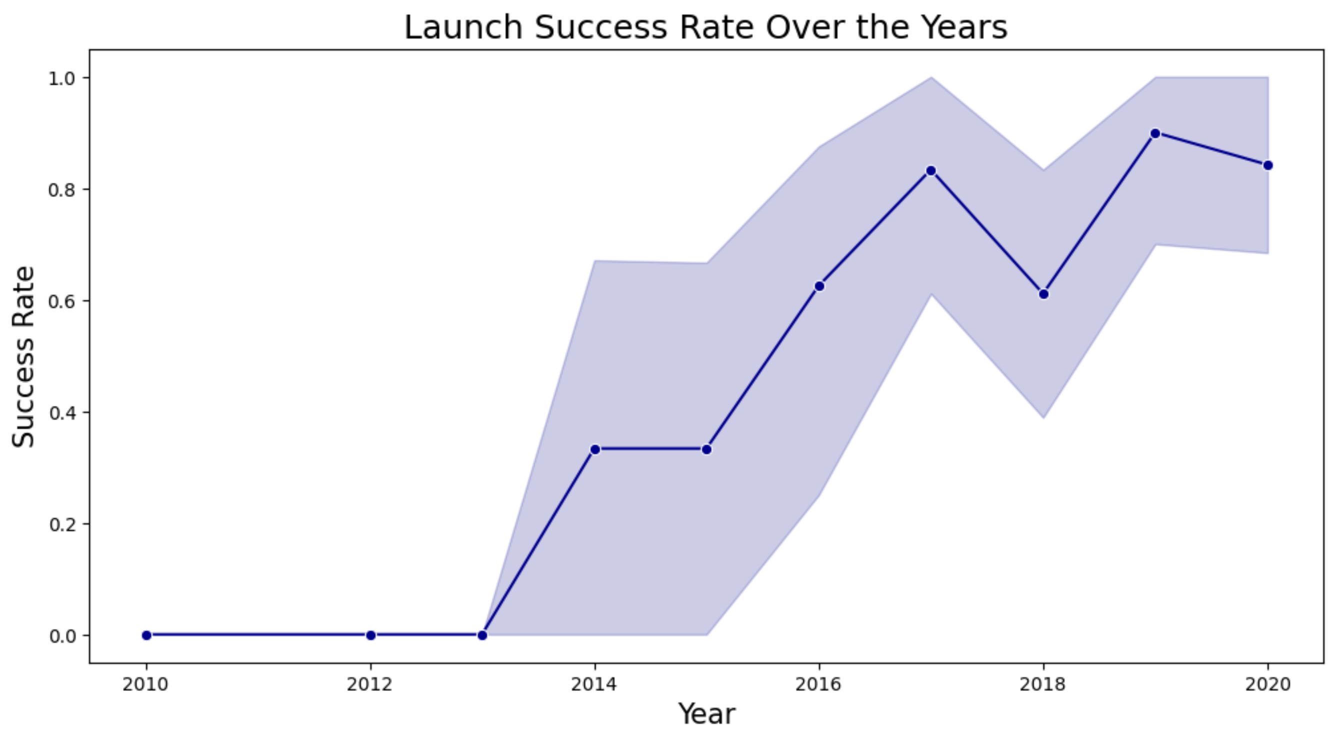
- Show a scatter point of payload vs. orbit type

- Show the screenshot of the scatter plot with explanations



Launch Success Yearly Trend

- Show a line chart of yearly average success rate



Section 3

EDA with SQL

All Launch Site Names

- Query

```
SELECT DISTINCT LAUNCH_SITE
FROM SPACEXTBL
```

- Result

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

- When the SQL DISTINCT clause is used in the query, only unique values are displayed in the Launch_Site column from the SpaceX table.
- There are four unique launch sites:

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

Launch Site Names Begin with 'CCA'

- Query

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

- Result

- 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_Site name starting with CAA could be called.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Query

```
SELECT SUM(PAYLOAD_MASS__KG_)
AS total_payload_mass_kg
FROM SPACEXTBL
WHERE CUSTOMER = 'NASA (CRS)'
```

- Result

TotalPayloadMassNASA_CRS
45596

- Using the SUM() function to calculate the sum of column PAYLOAD_MASS__KG_.
- In the WHERE clause, filter the dataset to perform calculations only if Customer is NASA (CRS).

Average Payload Mass by F9 v1.1

- Query

```
SELECT AVG(PAYLOAD_MASS__KG_)
AS avg_payload_mass_kg
FROM SPACEXTBL
WHERE BOOSTER_VERSION = 'F9 v1.1'
```

- Result

AveragePayloadMass
2928.4

- Using the AVG() function to calculate the average value of column PAYLOAD_MASS__KG_.
- In the WHERE clause, filter the dataset to perform calculations only if Booster_version is F9 v1.1.

First Successful Ground Landing Date

- Query

```
SELECT MIN(DATE)
FROM SPACEXTBL
WHERE LANDING__OUTCOME
        = 'Success (ground pad)'
```

- Result

first_successful_landing_date
2015-12-22

- Using the MIN() function to find out the earliest date in the column DATE.
- In the WHERE clause, filter the dataset to perform a search only if Landing__outcome is Success (ground pad).

Successful Drone Ship Landing with Payload between 4000 and 6000

- Query

```
SELECT BOOSTER_VERSION
FROM SPACEXTBL
WHERE LANDING__OUTCOME = 'Success (drone ship)'
AND (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000)
```

- Result

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

In the WHERE clause, filter the dataset to perform a search if Landing__outcome is Success (drone ship).

Using the AND operator to display a record if additional condition PAYLOAD_MASS__KG_ is between 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

- Query

```
SELECT MISSION_OUTCOME,
COUNT(*) AS total_number
FROM SPACEXTBL
GROUP BY MISSION_OUTCOME
```

- Result

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

Using the COUNT() function to calculate the total number of columns.

Using the GROUP BY statement, groups rows that have the same values into summary rows to find the total number in each Mission_outcome.

According to the result, SpaceX seems to have **successfully completed nearly 99% of its missions.**

Boosters Carried Maximum Payload

- Query

```
SELECT DISTINCT BOOSTER_VERSION,
       PAYLOAD_MASS__KG_
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (
       SELECT MAX(PAYLOAD_MASS__KG_)
       FROM SPACEXTBL)
```

- 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.x boosters could carried the maximum payload.

- Result

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

2015 Launch Records

- Query

```
SELECT LANDING__OUTCOME,
       BOOSTER_VERSION,
       LAUNCH_SITE
FROM SPACEXTBL
WHERE LANDING__OUTCOME
       = 'Failure (drone ship)’
       AND YEAR(DATE) = '2015'
```

- Result

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

- 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.

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

- Query

```
SELECT LANDING__OUTCOME,
       COUNT(LANDING__OUTCOME) AS total_number
FROM SPACEXTBL
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY LANDING__OUTCOME
ORDER BY total_number DESC
```

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.

- Result

Landing_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

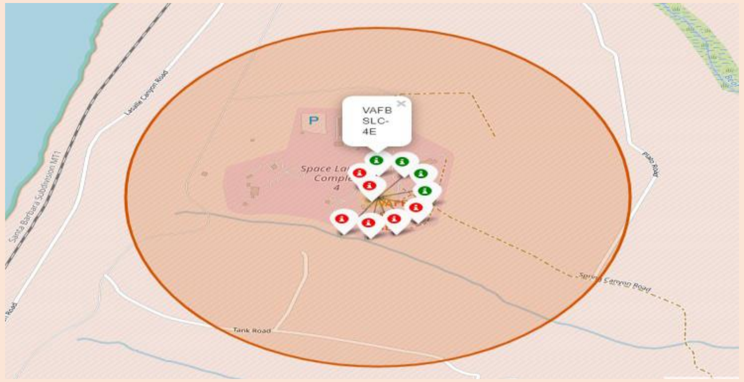
Section 3

Launch Sites Proximities Analysis

All 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-labeled Launch Outcomes

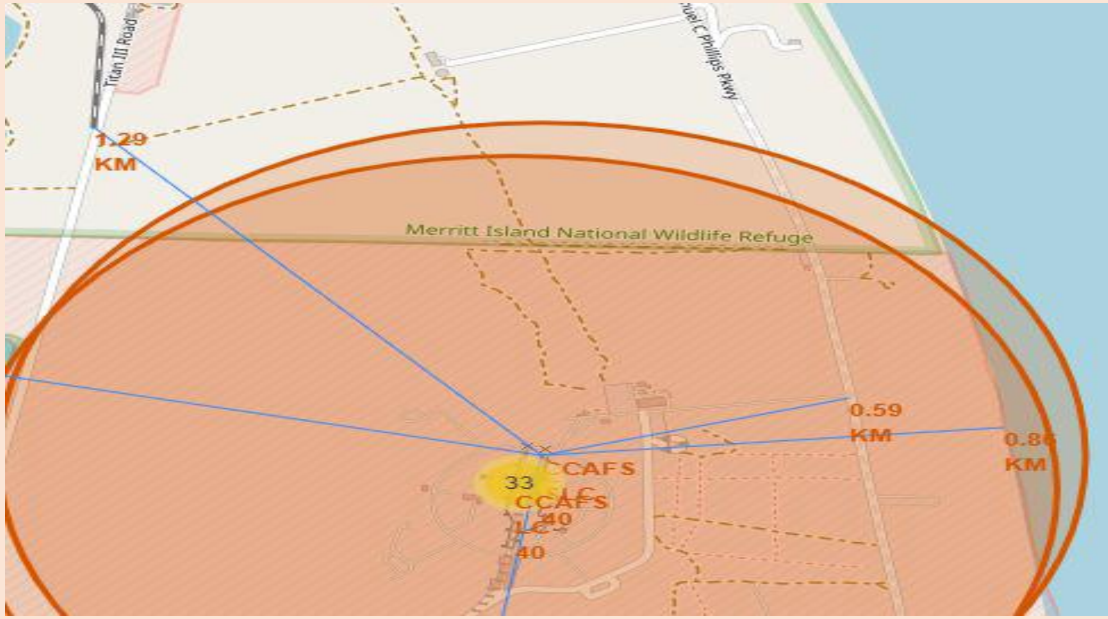


Launch site in California

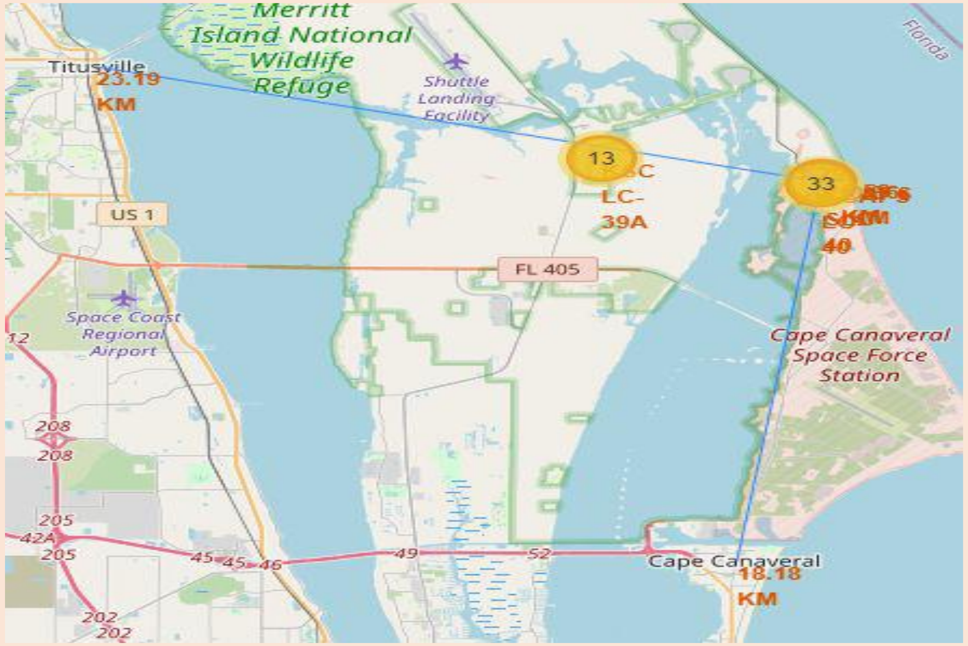
Launch sites in Florida

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

Proximities of Launch Sites



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***



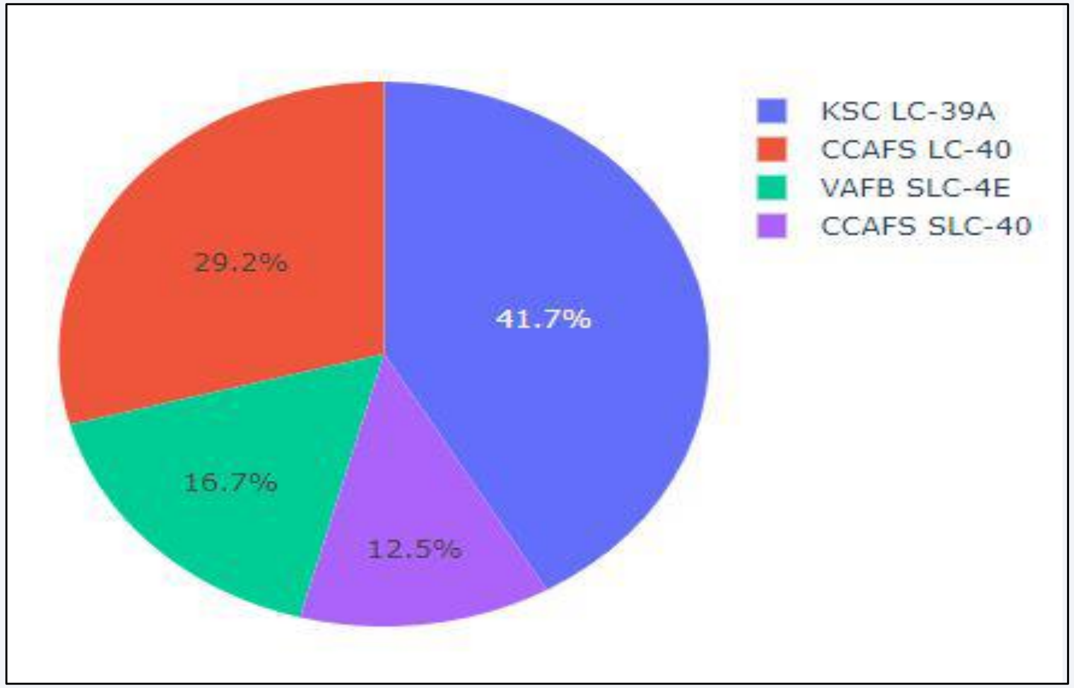
Do launch sites keep certain distance away from cities? ***Yes***

It can be found that 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.

Section 4

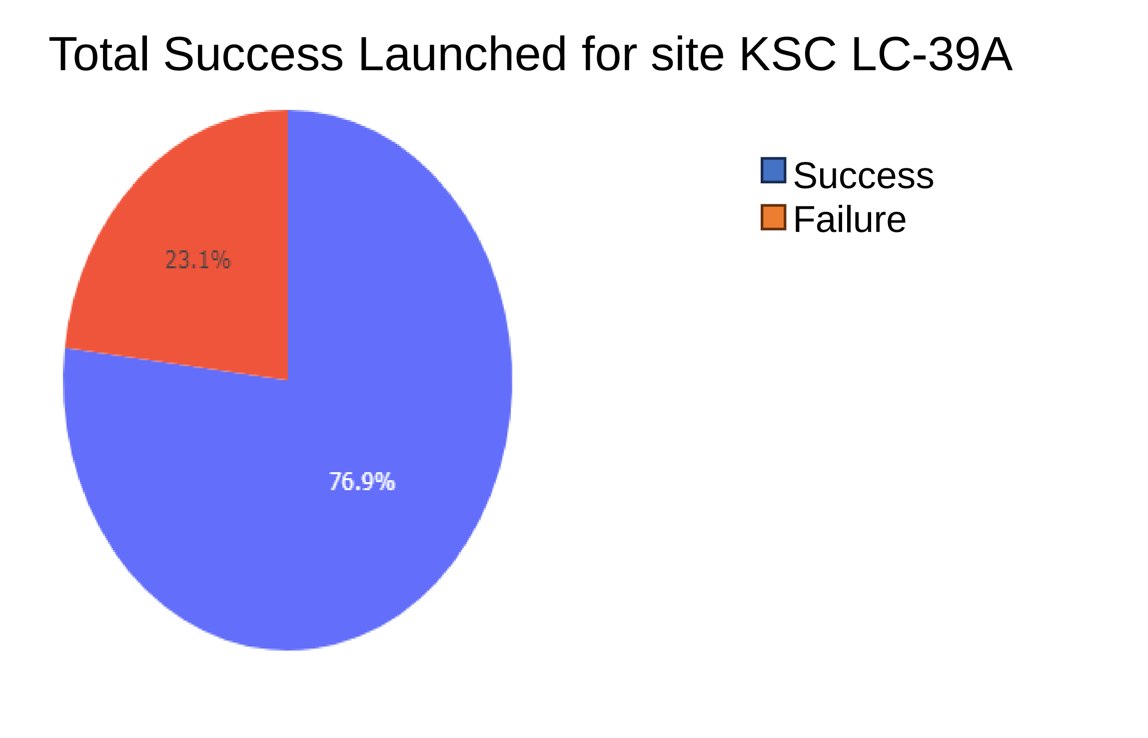
Build a Dashboard with Plotly Dash

Total Success Launches By all sites



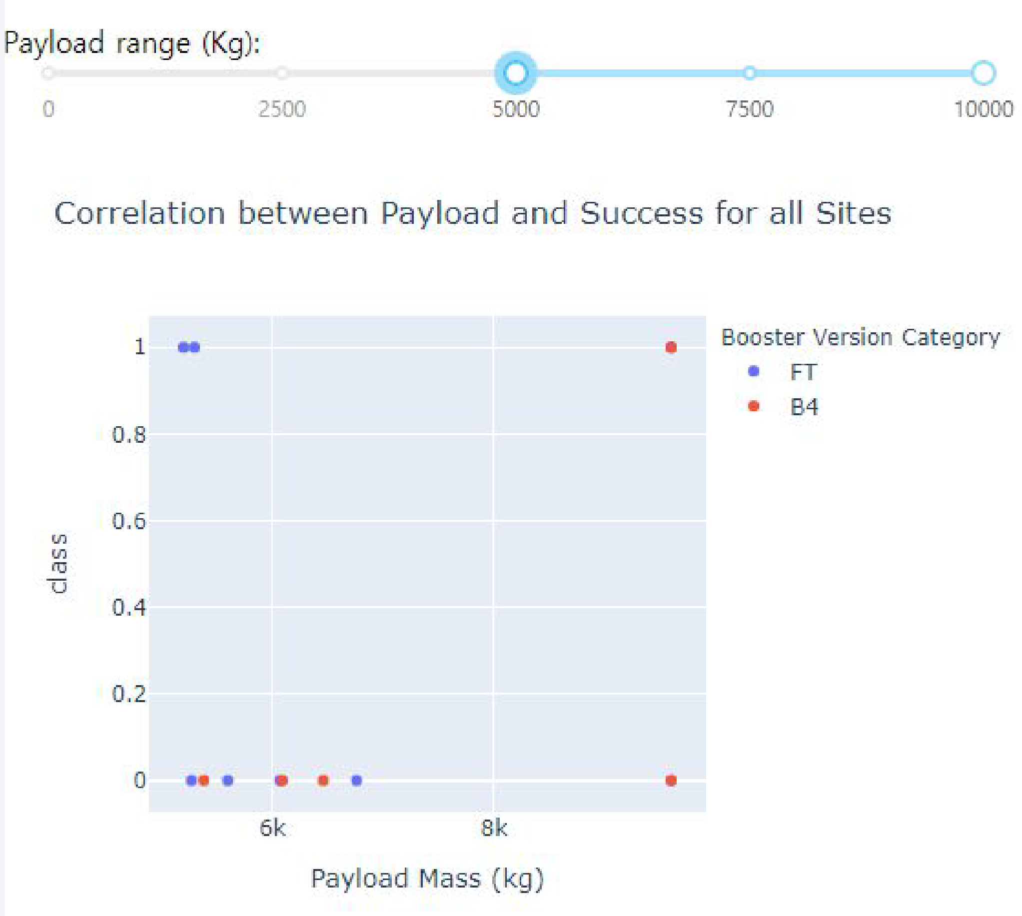
- KSLC-39A records the most launch success among all sites.
- The VAFB SLC-4E has the fewest launch success, possibly because
  - ❑the data sample is small, or
  - ❑because it is the only site located in California, so the launch difficulty on the west coast may be higher than on the east coast.

<Dashboard Screenshot 2>



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

<Dashboard Screenshot 3>

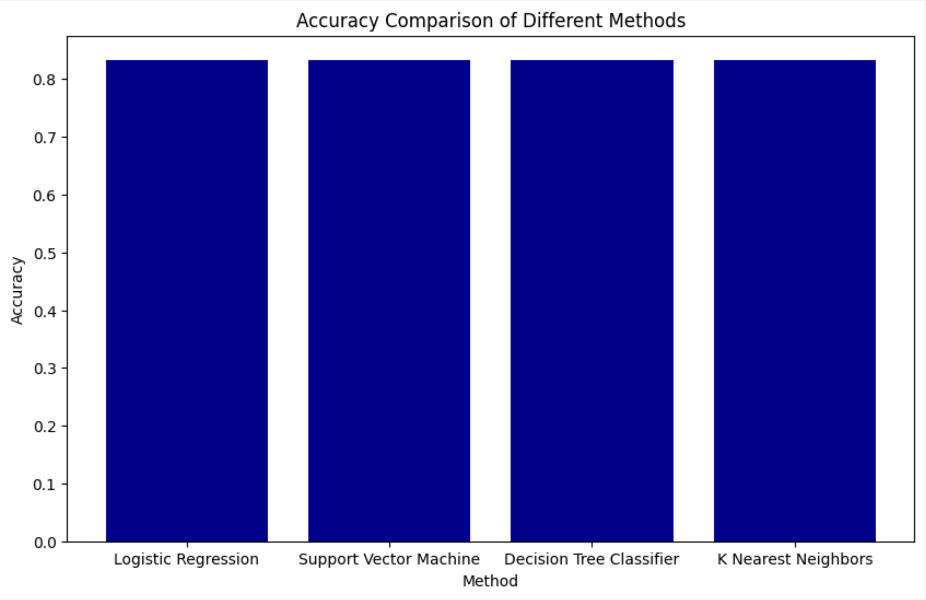


These figures show that **the launch success rate** (class 1) for **low weighted payloads(0-5000 kg) is higher** than that of heavy weighted payloads(5000-10000 kg).

Section 5

Predictive Analysis (Classification)

Classification Accuracy



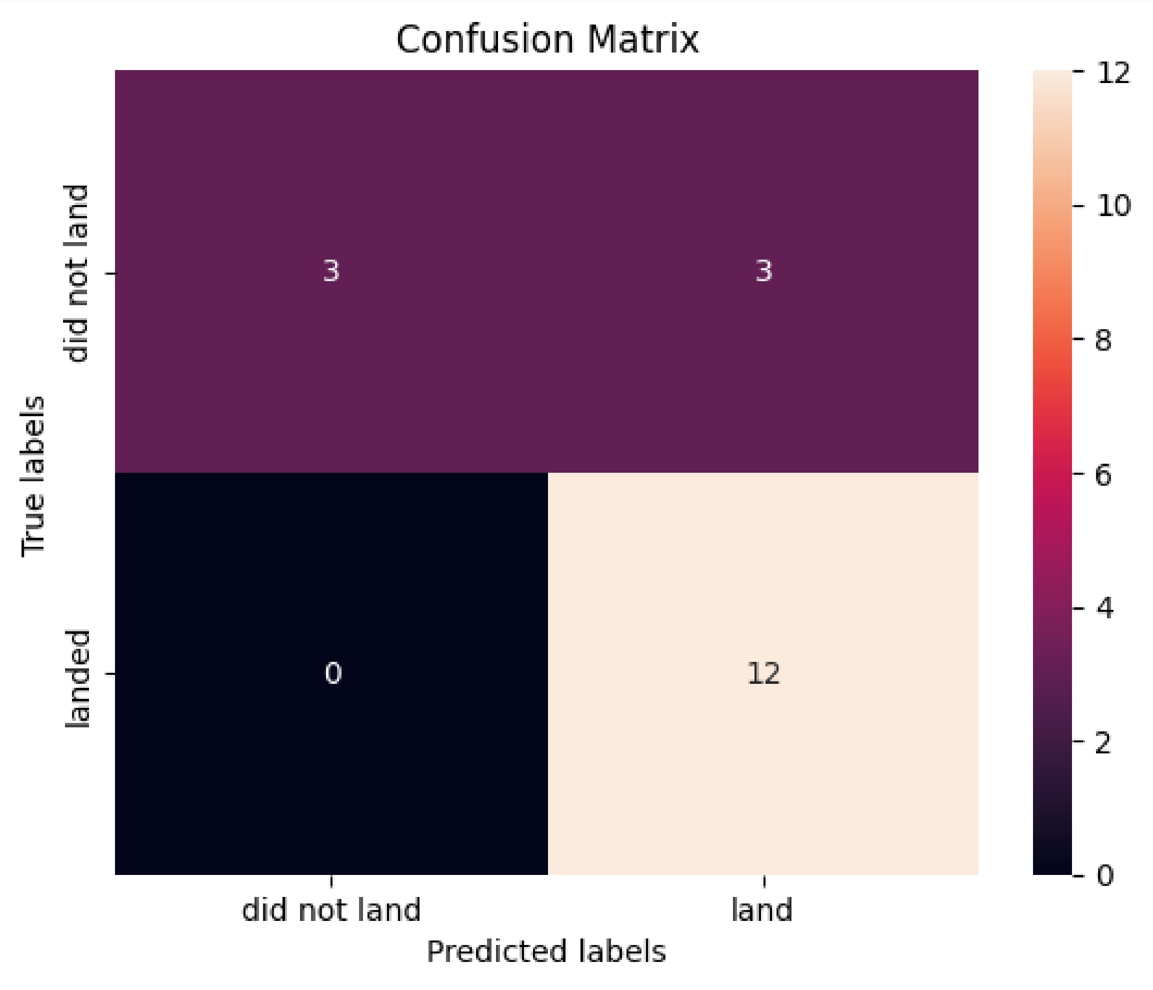
	Methods	Accuracy
0	Logistic Regression	0.833333
1	Support Vector Machine	0.833333
2	K Nearest Neighbors	0.833333
3	Decision Tree Classifier	0.833333

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.

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

Thank you!