IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Marek Ruzicka
13.3.2025

# Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

# Executive Summary

- Summary of methodologies
    - Data collection via API, Web-scraping
    - Data wrangling
    - Exploratory Data Analysis with SQL, Data visualization
    - Machine learning predictions
- Summary of all results
    - Exploratory Data Analysis results
    - Prediction results

# Introduction

- Project background and context
  - 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 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
  - What factors determine if the rocket lands successfully
  - Which conditions need to take place to ensure a successful landing program

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:
  - Data was collected via SpaceX API and Web-scraping from Wikipedia
- Perform data wrangling
  - Change of data types where applicable
  - One-hot encoding for categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Different classification models were used

# Data Collection

- Describe how data sets were collected.
  - Data collection was done via SpaceX API using get request, from Wikipedia using Web-scraping with BeautifulSoup method
  - Content was decoded using .json() function and transformed to pandas dataframe using .json_normalize() function
  - Data were cleaned, checked for missing values and data types

# Data Collection – SpaceX API

- Get request was used to SpaceX API to collect data, clean data and data wrangling

- GitHub URL: [https://github.com/MarekBosch/Data-science/blob/main/jupyter-labs-spacex-data-collection-api.ipynb](https://github.com/MarekBosch/Data-science/blob/main/jupyter-labs-spacex-data-collection-api.ipynb)

Now let's start requesting rocket launch data from SpaceX API with the following URL:

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

```
[7]: response = requests.get(spacex_url)
```

Task 1: Request and parse the SpaceX launch data using the GET request ¶

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
[9]: static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
```

We should see that the request was successfull with the 200 status response code

```
[10]: response=requests.get(static_json_url)
```

```
[11]: response.status_code
```

```
[11]: 200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

```
[19]: # Use json_normalize meethod to convert the json result into a dataframe
      data = response.json()
```

```
[22]: data = pd.json_normalize(data)
```

Using the dataframe `data` print the first 5 rows

```
[23]: # Get the head of the dataframe
      data.head()
```

# Data Collection - Scraping

- BeautifulSoup for web-scraping Falcon9 launch records was used
- It was converted to pandas dataframe
- GitHub URL:

[https://github.com/MarekBosch/Data-science/blob/main/jupyter-labs-webscraping.ipynb](https://github.com/MarekBosch/Data-science/blob/main/jupyter-labs-webscraping.ipynb)

## TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
[5]: # use requests.get() method with the provided static_url
response = requests.get(static_url)
# assign the response to a object
```

Create a `BeautifulSoup` object from the HTML `response`

```
[6]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
[8]: # Use soup.title attribute
page_title = soup.title.string
page_title
```

```
[8]: 'List of Falcon 9 and Falcon Heavy launches - Wikipedia'
```

# Data Wrangling

- Describe how data were processed
  - Understanding the label: The label for the machine learning model is the landing success of the Falcon 9 booster. We want to predict whether the booster will land successfully, so we need to create a target variable that reflects this
  - Combining Landing Outcomes into a Single Label: The next step involves transforming these multiple columns into a single label column that represents landing success.

- GitHub URL:

https://github.com/MarekBosch/Data-science/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

# EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - Scatterplots – to see correlation between parameters
  - Linechart – to see the trend
  - Bar charts – to check visually the relationships
- GitHub URL:

https://github.com/MarekBosch/Data-science/blob/main/edadataviz.ipynb

# EDA with SQL

- Performed SQL queries:
  - the names of the unique launch sites in the space mission
  - 5 records where launch sites begin with the string 'CCA'
  - the total payload mass carried by boosters launched by NASA (CRS)
  - average payload mass carried by booster version F9 v1.1
  - the date when the first succesful landing outcome in ground pad was acheived
  - the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - the total number of successful and failure mission outcomes
  - the names of the booster_versions which have carried the maximum payload mass. Use a subquery
  - the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
  - 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
- GitHub URL:

https://github.com/MarekBosch/Data-science/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

# Build an Interactive Map with Folium

- All launch sites were marked, map objects markers, circles, lines to mark the success or failure of launches were added
- Using color marker we identified which launches sites have higher success rate
- GitHub URL:

https://github.com/MarekBosch/Data-science/blob/main/lab_jupyter_launch_site_location.ipynb

# Build a Dashboard with Plotly Dash

- Added plots/graphs to a dashboard
  - Interactive dashboard with Plotly dash was created
  - Pie charts, scatterplot
- Plots were added to show relationships and interactions

# Predictive Analysis (Classification)

- Data were loaded using numpy and pandas, data were transformed, split into training and testing
- Different machine learning models were created and tuned using GridSearchCV
- Accuracy was used as a metric for model, feature engineering was used for improvements
- In the end, the best performing classification model was defined
- GitHub URL:

https://github.com/MarekBosch/Data-science/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

# Results

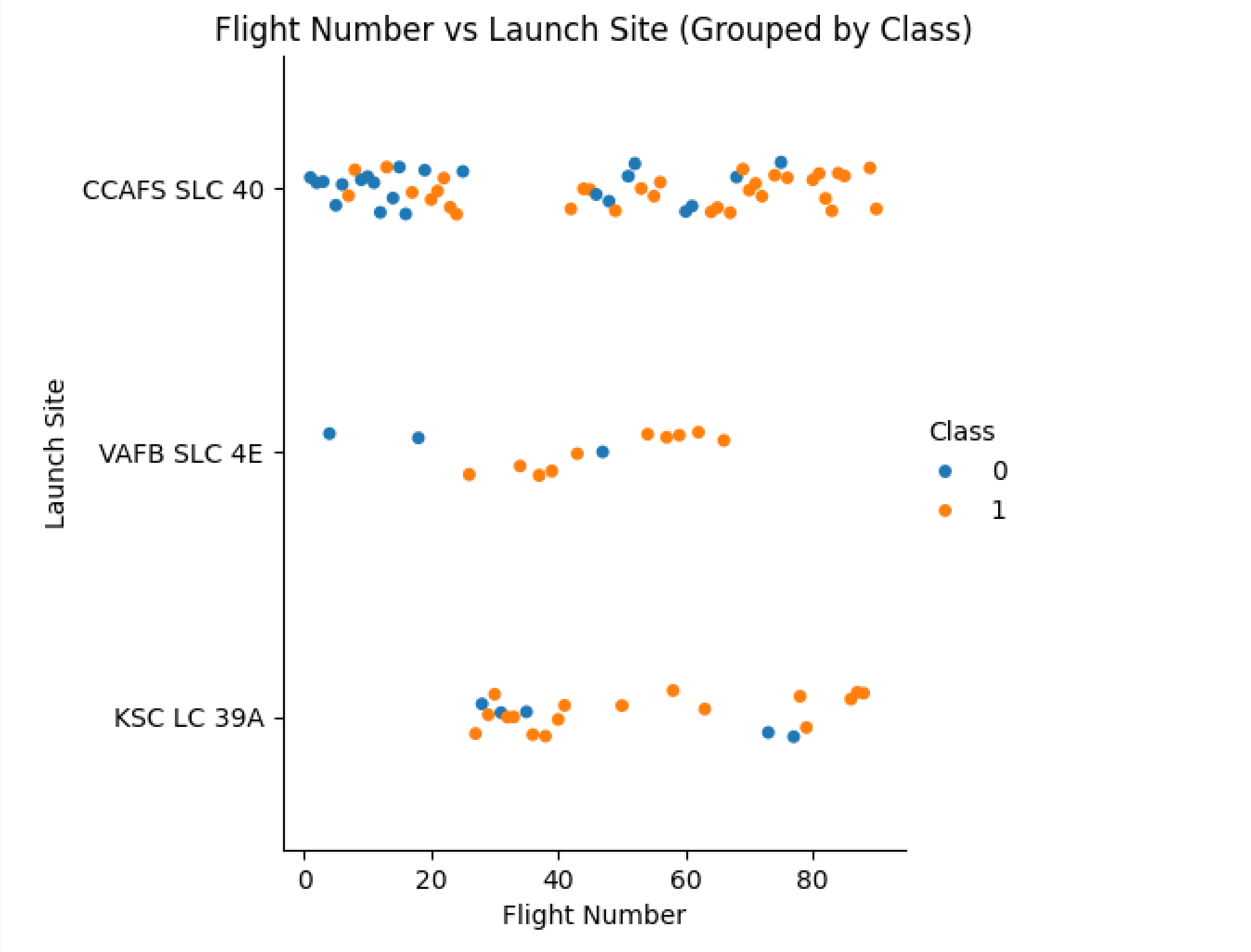
- Exploratory data analysis results
  - Distribution of landing success
  - Launch site success rate
- Interactive analytics demo in screenshots
- Predictive analysis results
  - Model performance
  - Confusion matrix

Section 2

# Insights drawn from EDA

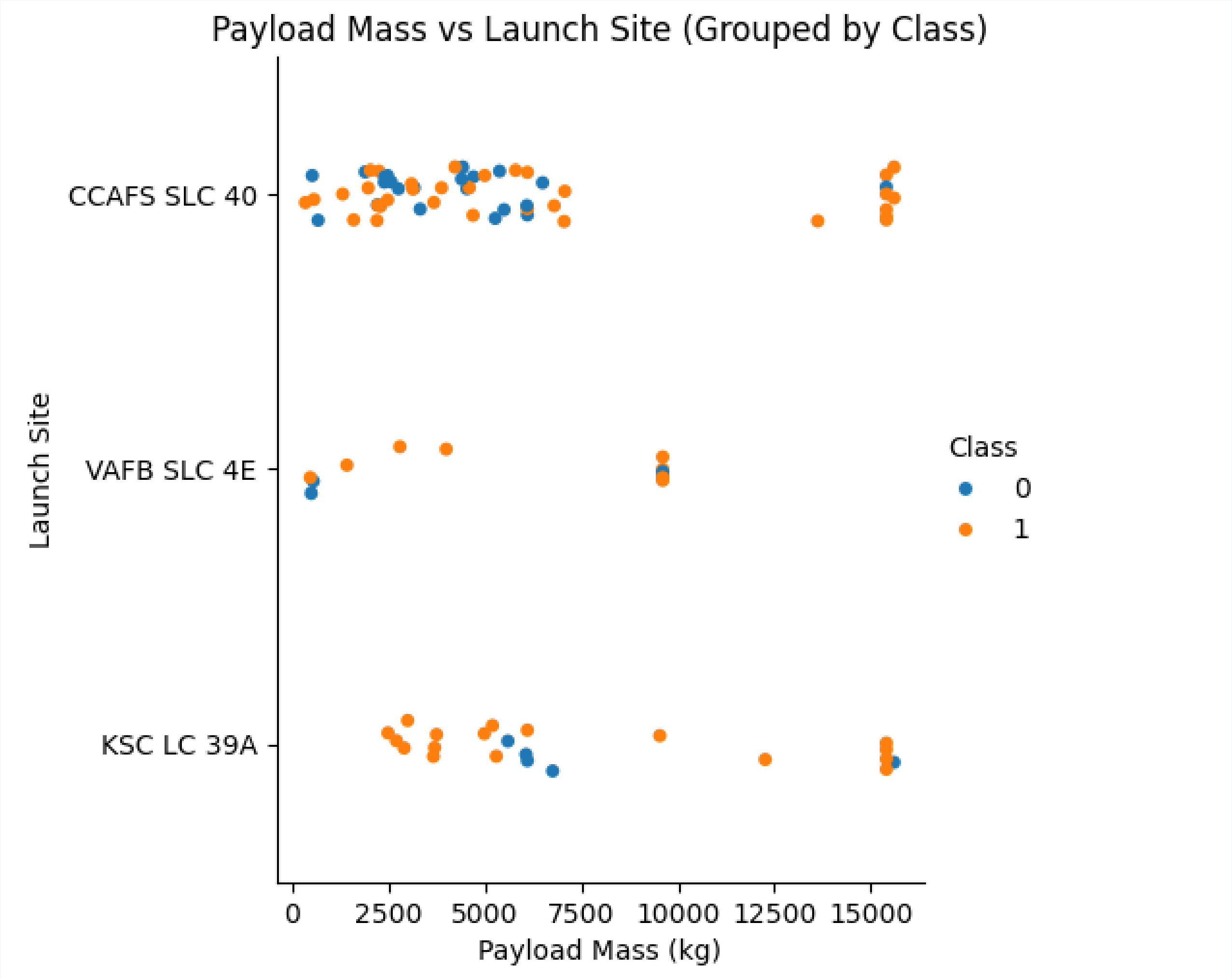
# Flight Number vs. Launch Site

- In the plot, the bigger flight number, the more success rate



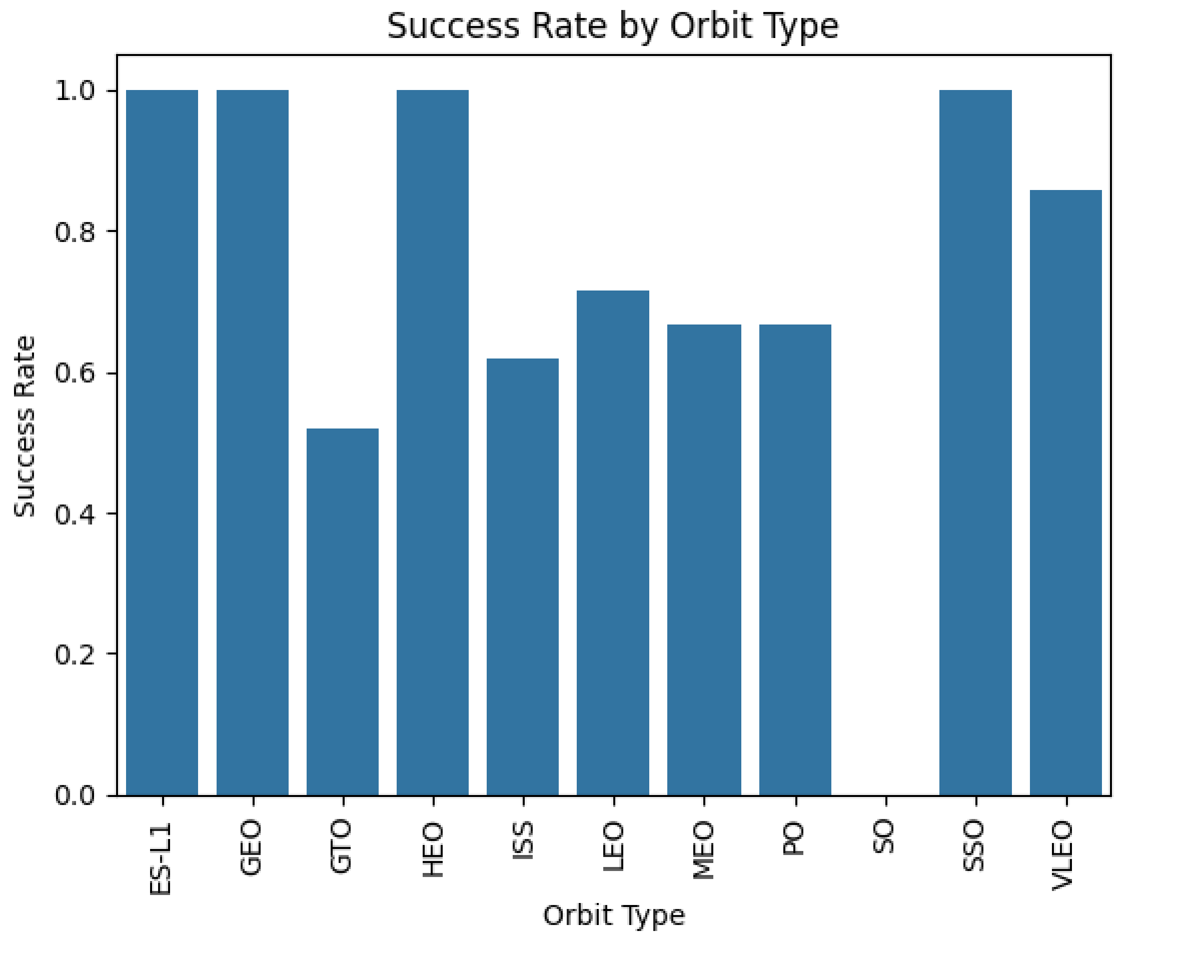
# Payload vs. Launch Site

- In the plot, the bigger payload mass, the higher success rate



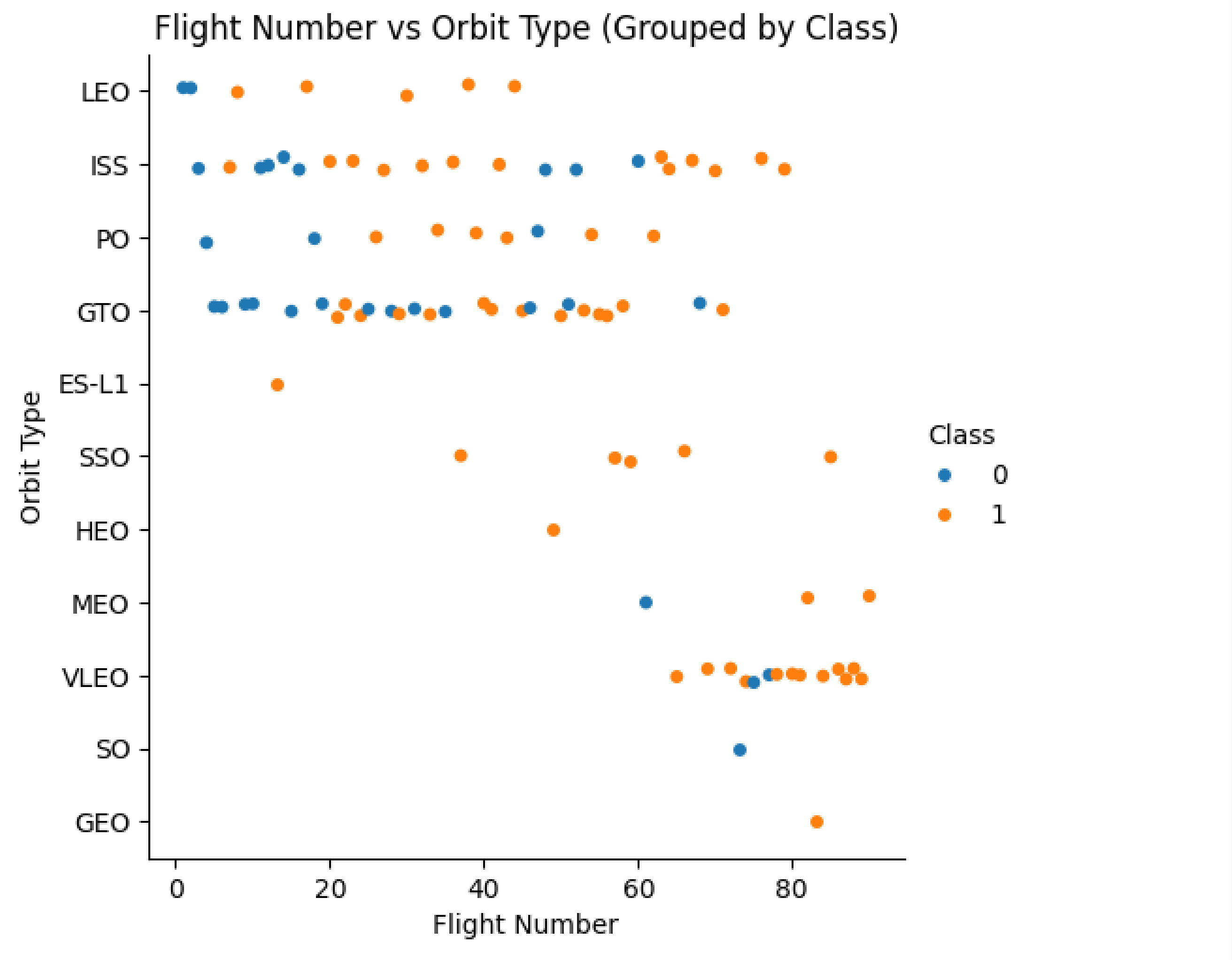
# Success Rate vs. Orbit Type

- In the plot, the most success rate for ES-L1, GEO, HEO, SSO



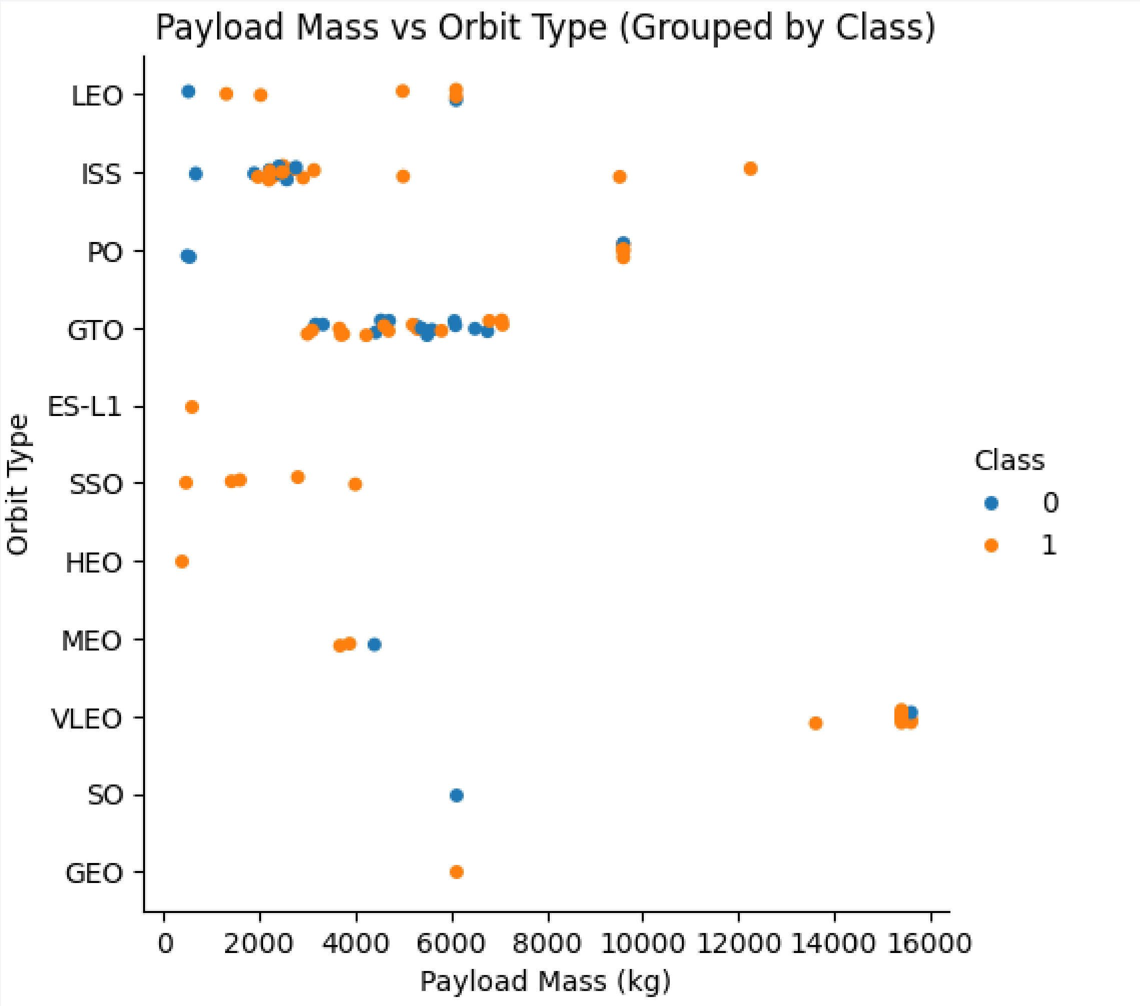
# Flight Number vs. Orbit Type

- In the plot, flight number vs orbit type, and success rate



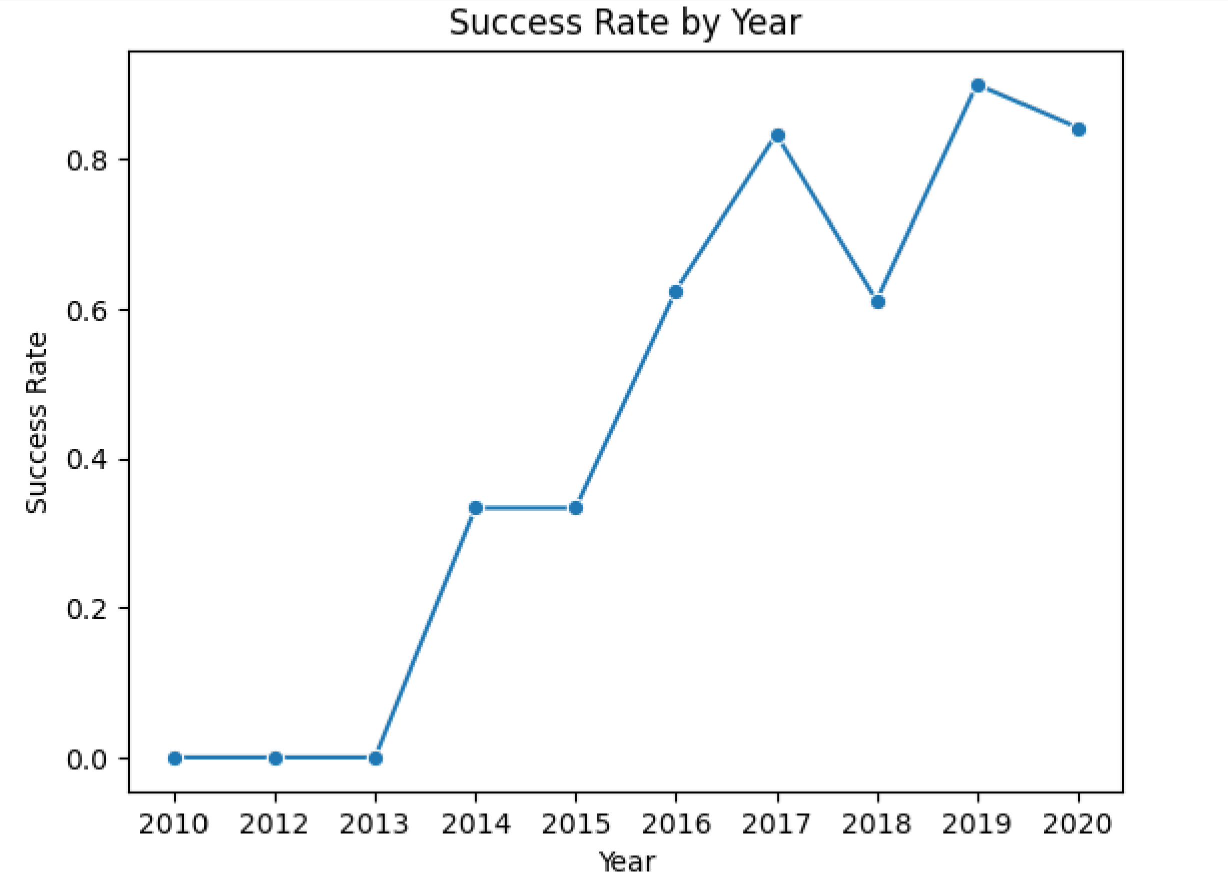
# Payload vs. Orbit Type

- In the plot, heavy payloads have high success rate for some orbit types



# Launch Success Yearly Trend

- In the plot, success rate is increasing since 2013



# All Launch Site Names

- Key word: DISTINCT

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

# Launch Site Names Begin with 'CCA'

- Key word: WHERE, LIMIT

| 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

- Key word: SUM, WHERE

| total_payload_mass |
| --- |
| 45596 |

# Average Payload Mass by F9 v1.1

- Key word: AVG, WHERE

| average_payload_mass |
|---|
| 2534.6666666666665 |

# First Successful Ground Landing Date

- Key word: MIN, WHERE

| first_successful_landing_date |
|---|
| 2018-07-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

- Key word: WHERE

| Booster_Version |
| --- |
| F9 B5 B1046.2 |
| F9 B5 B1047.2 |
| F9 B5 B1048.3 |
| F9 B5 B1051.2 |
| F9 B5B1060.1 |
| F9 B5 B1058.2 |
| F9 B5B1062.1 |

# Total Number of Successful and Failure Mission Outcomes

- Key word: WHERE

| Landing_Outcome | total_count |
| --- | --- |
| Failure | 3 |
| Success | 38 |

# Boosters Carried Maximum Payload

- Key word: MAX, WHERE

| 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

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

- Present your query result with a short explanation here

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

- Key word: WHERE, GROUP BY, ORDER BY

| 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 |

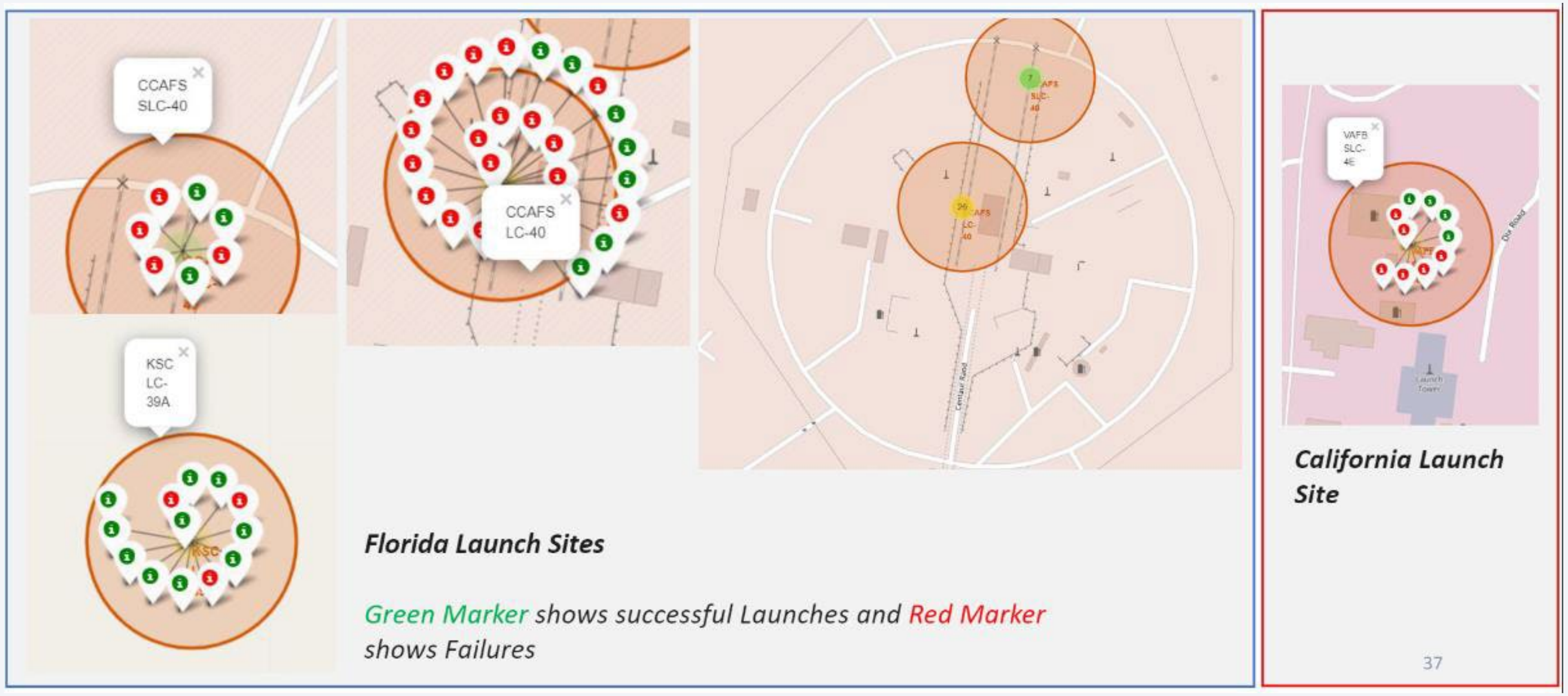
Section 3

# Launch Sites Proximities Analysis

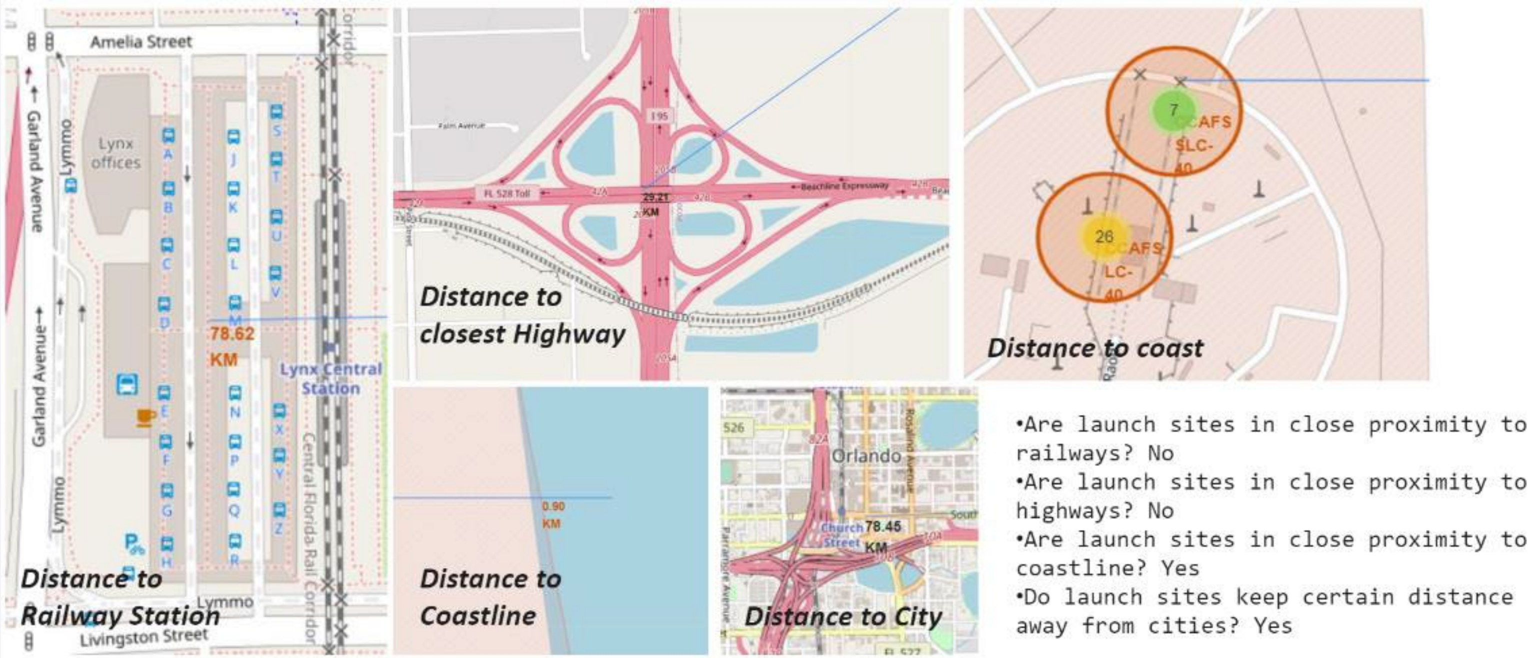
# All launch sites global map markers

- Launch sites in California and Florida

# Markers showing launch sites with color labels



# Launch Site distance to landmarks

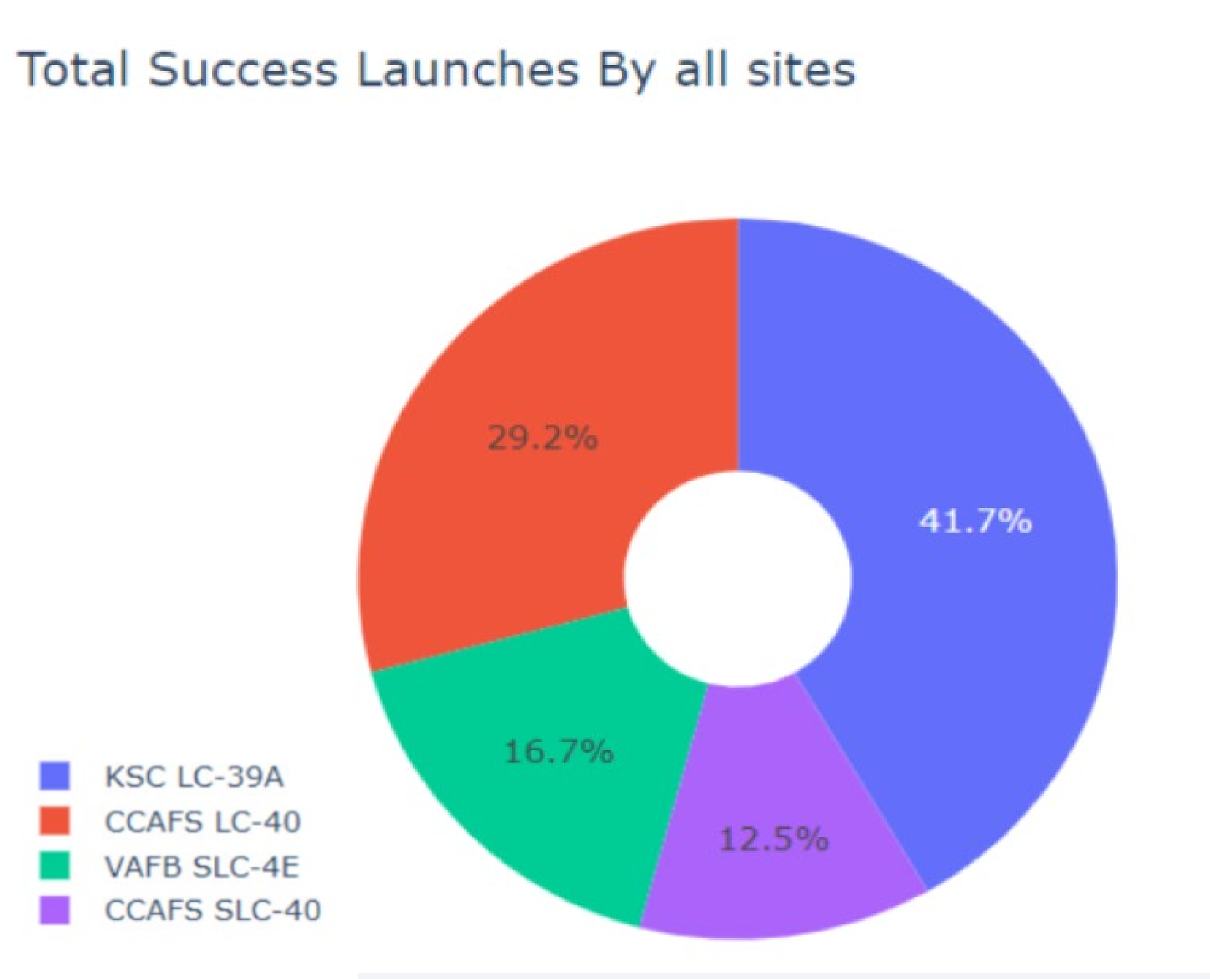


Section 4

# Build a Dashboard with Plotly Dash

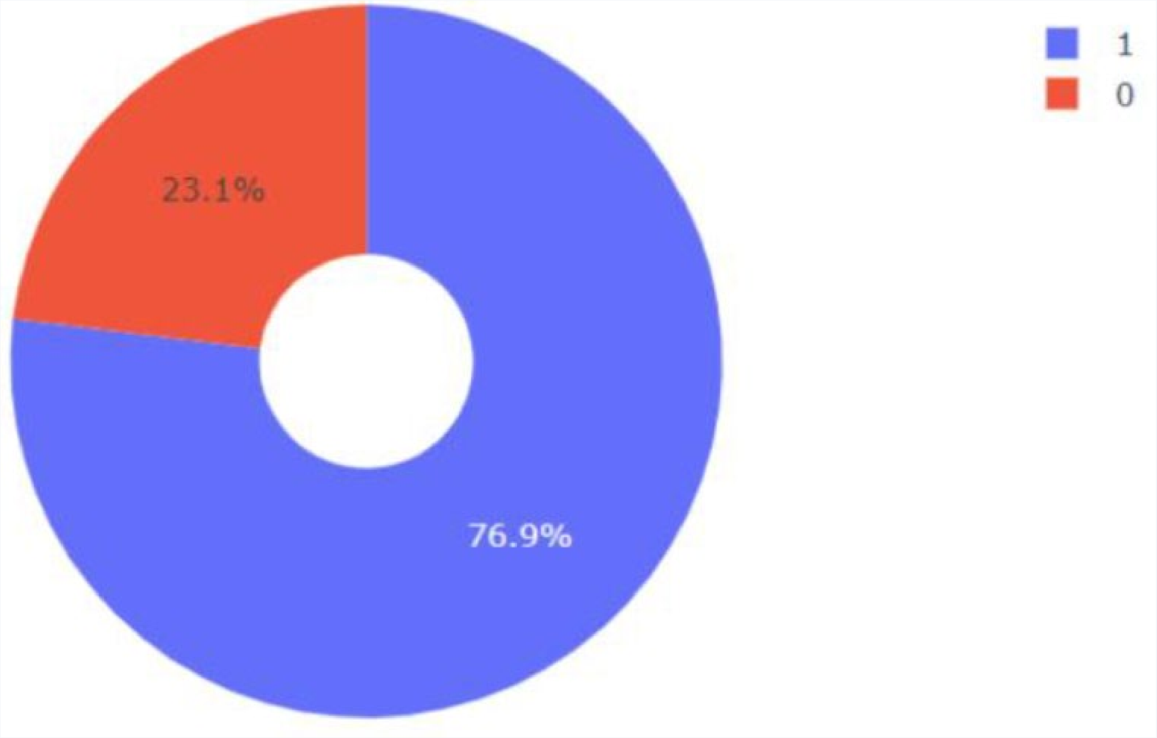
# The success percentage

- KSC LC-39A with most successful launches



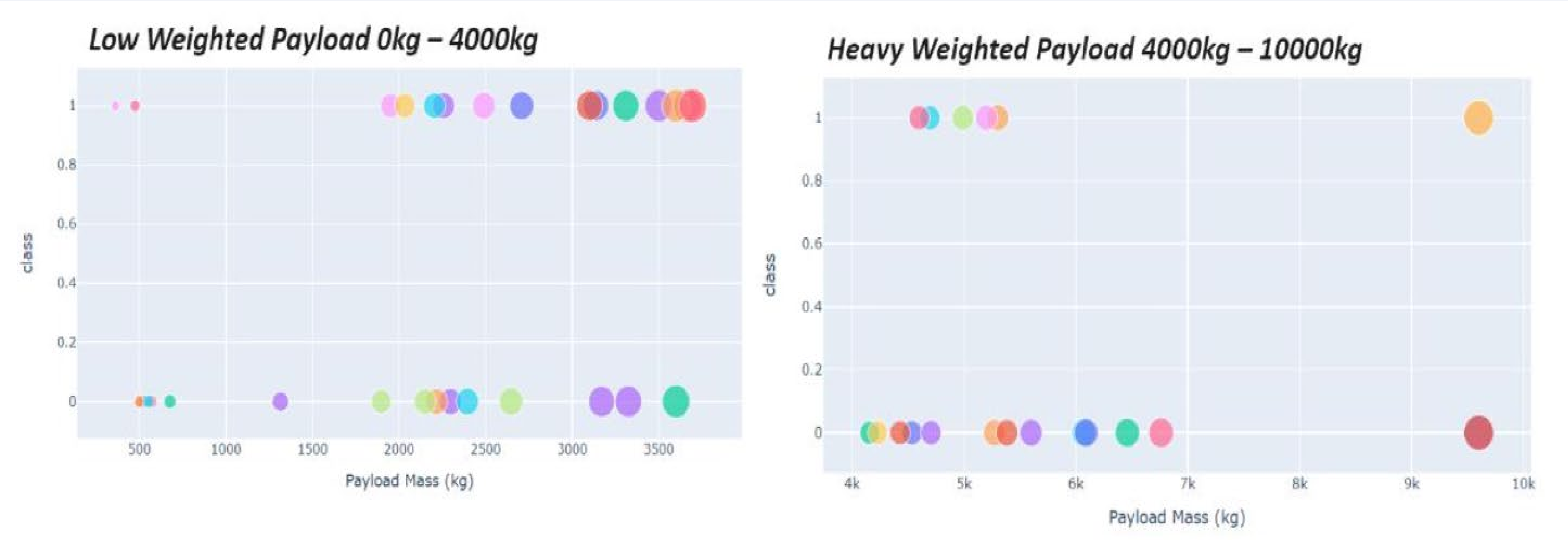
# The highest launch success ratio

- KSC LC-39A achieved 76.9% success rate



# Payload vs Launch Outcome for all sites

- Success rate for low weighted payloads is higher than the heavy weighted payloads

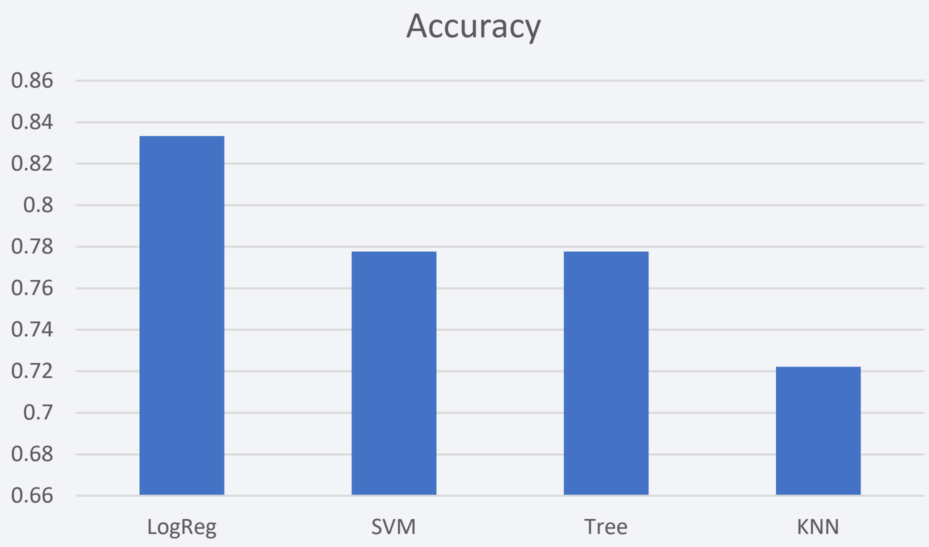


Section 5

# Predictive Analysis (Classification)

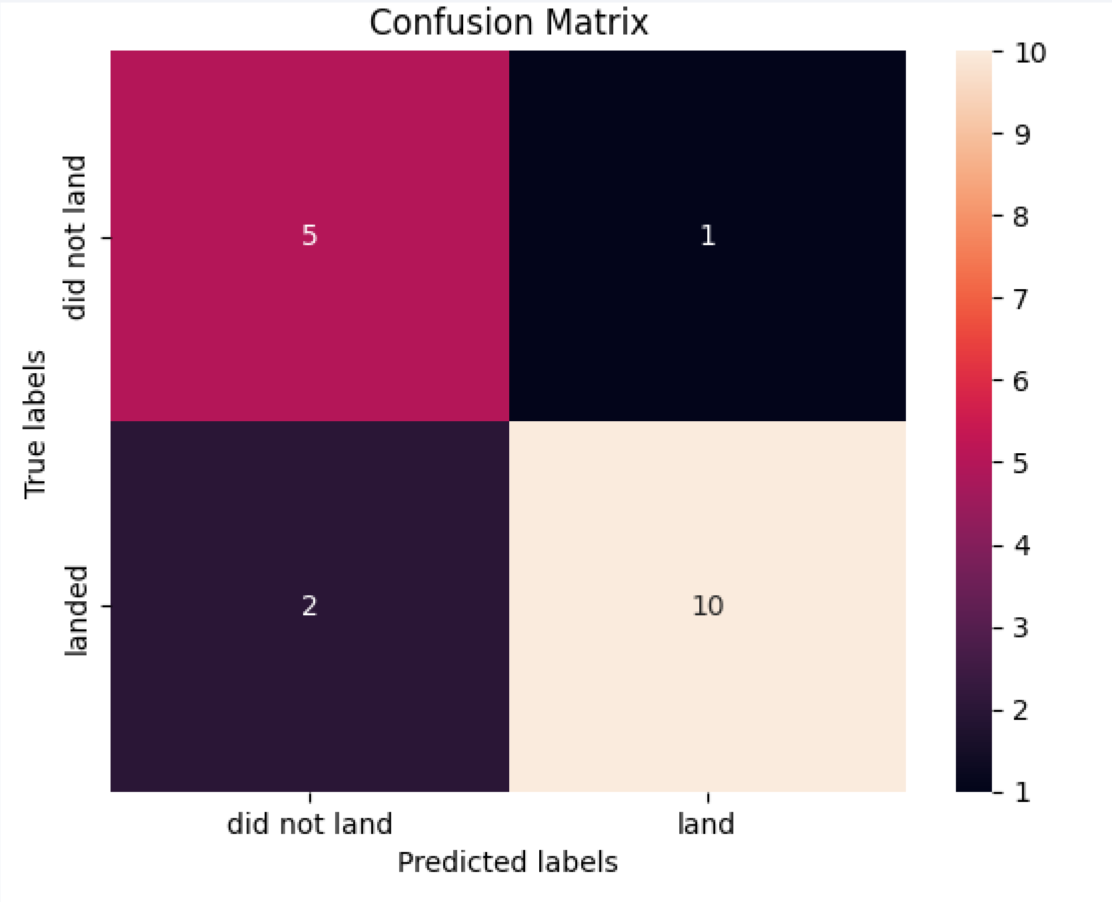
# Classification Accuracy

- In the plot, accuracy for each model



# Confusion Matrix

- Confusion matrix for logistic regression



# Conclusions

We can conclude that:

- A higher number of flights at a launch site is associated with a higher success rate at that site.
- The success rate of launches began to rise from 2013 and continued to improve through 2020.
- The orbits ES L1, GEO, HEO, SSO, and VLEO experienced the highest success rates.
- KSC LC 39A had the highest number of successful launches among all launch sites.
- The Decision Tree Classifier is the most effective machine learning algorithm for this task.

# Thank you!