IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Krzysztof Kuba
25.03.2025

Outline

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

Executive Summary

- Methodologies
  - Data Collection: Web scraping and API requests to gather data.
  - Data Wrangling: Cleaning and transforming datasets.
  - EDA and Visualization: Using Pandas, NumPy, Matplotlib, Seaborn, and SQL.
  - Machine Learning: Building pipelines for prediction, including preprocessing, model training, and evaluation.
- Key Results
  - Successfully collected additional data from websites.
  - Gathered and cleaned data from the SpaceX API.
  - Prepared the dataset for analysis and modeling.
  - Uncovered insights and visualized data trends.
  - Analyzed data using SQL queries and visualizations.
  - Mapped and analyzed SpaceX launch site locations.
  - Developed a machine learning model to predict landing success with high accuracy.

Introduction

Background and Context: The Capstone Project is the final course in the IBM Data Science Professional Certificate program, which aims to consolidate and apply the comprehensive skills and knowledge acquired throughout the program. The project focuses on analyzing and predicting the success of SpaceX Falcon 9 first stage landings. SpaceX, a private aerospace manufacturer and space transportation company, has been working on developing reusable rockets to reduce the cost of space travel. The success of these landings is crucial for the company's mission to make space exploration more affordable and sustainable.

Problem Statement: The primary objective of this project is to analyze various factors that influence the success of SpaceX Falcon 9 first stage landings and develop a predictive model to forecast the outcomes of future landings. By leveraging data science techniques, we aim to answer the following key questions:

1. What are the critical factors that determine the success of SpaceX Falcon 9 first stage landings?
2. How can we collect and preprocess data from multiple sources to build a comprehensive dataset for analysis?
3. What insights can be uncovered through exploratory data analysis (EDA) and visualization techniques?
4. How can we apply data wrangling techniques to clean and transform the dataset for modeling?
5. What machine learning models can be developed to predict the success of SpaceX landings, and how accurate are these models?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using web scraping and Specex API
- Perform data wrangling
  - Prepared the dataset for analysis and modeling.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The data sets for the Capstone Project were collected using two primary methods: web scraping and API requests. Here is a detailed description of each method:
    - The Webscraping.ipynb notebook demonstrates the process of collecting historical launch records of Falcon 9 and Falcon Heavy rockets from a Wikipedia page.
    - The Spacex_Data_Collection_Api.ipynb notebook covers the process of collecting data from the SpaceX API.

By combining data from web scraping and API requests, the project ensures a comprehensive dataset that includes historical launch records and additional details from the SpaceX API. This dataset is then used for exploratory data analysis, visualization, and machine learning model development to predict the success of SpaceX Falcon 9 first stage landings.

Data Collection – SpaceX API

- Steps:
  - Importing Required Libraries
  - Making API Requests
  - Parsing JSON Responses
  - Data Cleaning and Transformation:
  - Creating a DataFrame
  - Exporting to CSV

- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/Spacex_Data_Collection_Api.ipynb

Task 2: Filter the dataframe to only include `Falcon 9` launches

Finally we will remove the Falcon 1 launches keeping only the Falcon 9 launches. Filter the data dataframe using the `BoosterVersion` column to only keep the Falcon 9 launches. Save the filtered data to a new dataframe called `data_falcon9`.

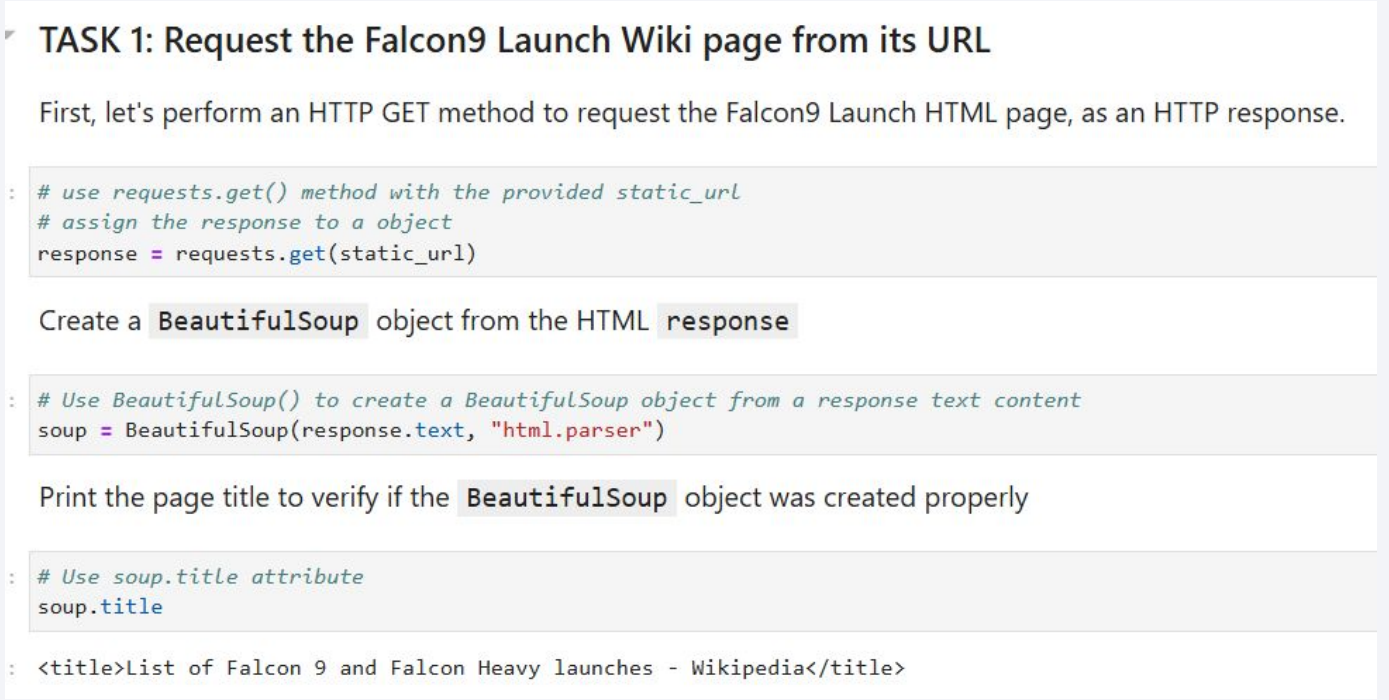
[35]:
```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = df[df['BoosterVersion']!='Falcon 1']
data_falcon9
```

[35]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None
5	8	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None
6	10	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None
7	11	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None

Data Collection - Scraping

- Steps
  - Importing Required Libraries
  - Requesting the HTML Page
  - Extracting Table Data.
  - Data Cleaning and Transformation
  - Creating a DataFrame
  - Exporting to CSV
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/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.

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

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

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, "html.parser")
```

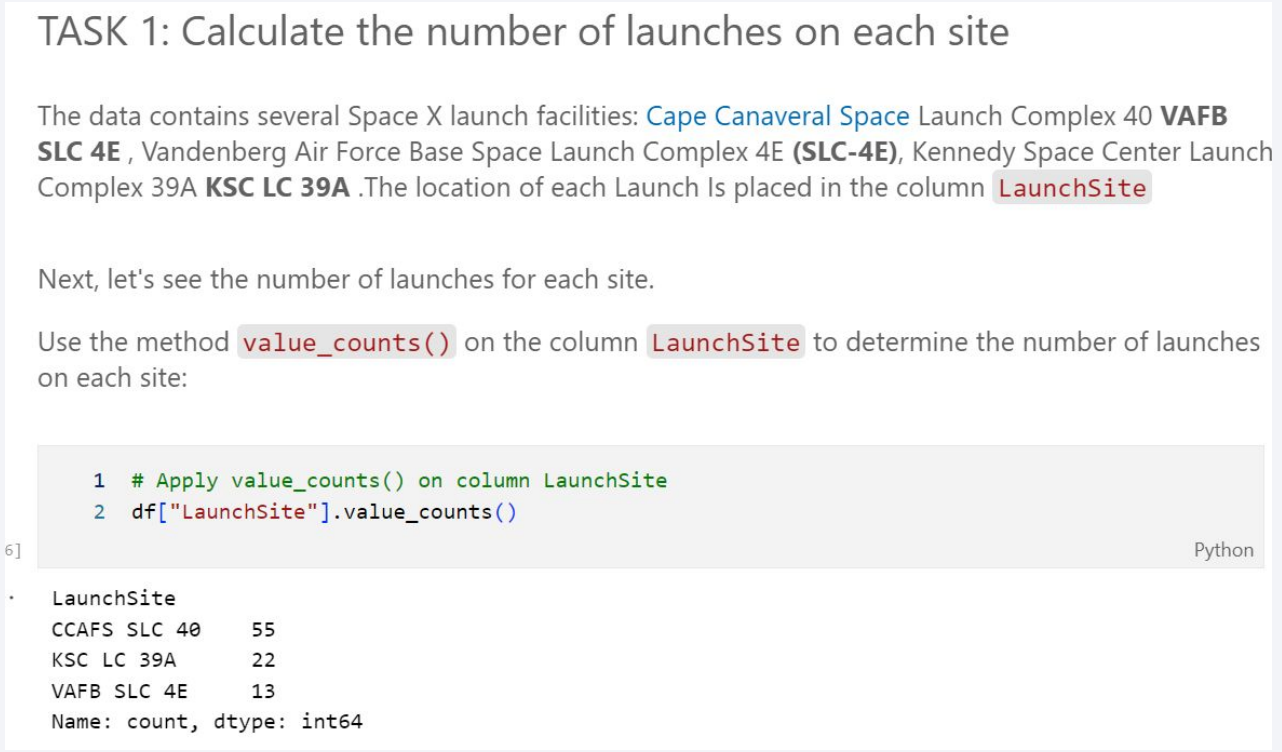
Print the page title to verify if the `BeautifulSoup` object was created properly

```
# Use soup.title attribute
soup.title
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Data Wrangling

- Importing Required Libraries:
- Loading the Data:
- Data Cleaning
  - Handling Missing Values
  - Removing Duplicates
  - Correcting Data Types
- Data Transformation
  - Feature Engineering
  - Normalization and Scaling
- Merging Datasets
- Saving the Processed Data
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/Spacex_Data_Wrangling.ipynb



TASK 1: Calculate the number of launches on each site

The data contains several Space X launch facilities: Cape Canaveral Space Launch Complex 40 **VAFB SLC 4E** , Vandenberg Air Force Base Space Launch Complex 4E **(SLC-4E)**, Kennedy Space Center Launch Complex 39A **KSC LC 39A** .The location of each Launch Is placed in the column `LaunchSite`

Next, let's see the number of launches for each site.

Use the method `value_counts()` on the column `LaunchSite` to determine the number of launches on each site:

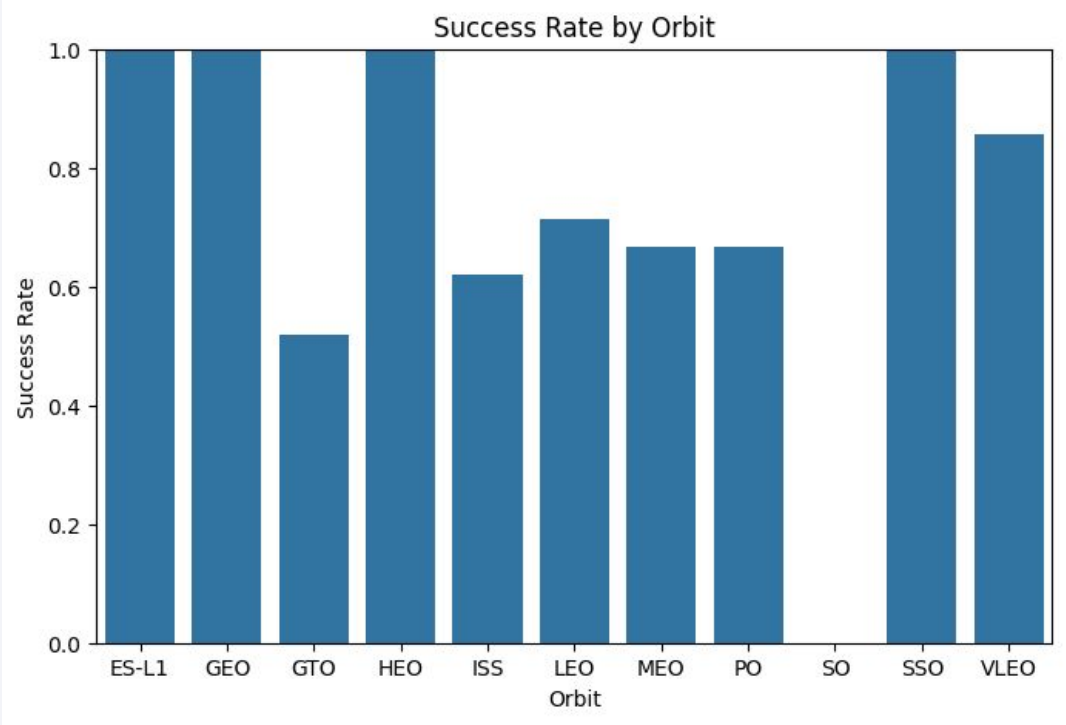
```
# Apply value_counts() on column LaunchSite
df["LaunchSite"].value_counts()
```

6] Python

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

EDA with Data Visualization

- Scatter Plot: Flight Number vs. Payload Mass
- Scatter Plot: Flight Number vs. Launch Site
- Scatter Plot: Payload Mass vs. Launch Site
- Bar Chart: Success Rate by Orbit
- Scatter Plot: Flight Number vs. Orbit
- Scatter Plot: Payload Mass vs. Orbit
- Line Chart: Yearly Success Rate
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/EDA_Data_Visualization.ipynb



EDA with SQL

- Display the names of the unique launch sites in the space mission.
- Display 5 records where launch sites begin with the string 'CCA'.
- Display the total payload mass carried by boosters launched by NASA (CRS).
- Display the average payload mass carried by booster version F9 v1.1.
- List the date when the first successful landing outcome on ground pad was achieved.
- List the names of the boosters which have success on drone ship and have payload mass greater than 4000 but less than 6000.
- List the total number of successful and failed mission outcomes.
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/EDA_SQL.ipynb

Build an Interactive Map with Folium

In the Launch_Site_Location.ipynb notebook, various map objects were created and added to a Folium map to visualize the locations and proximities of SpaceX launch sites. Here is a summary of the map objects added:

- Folium Map Object:
- Circles and Markers for Launch Sites:
- Marker Cluster for Launch Outcomes:
- Mouse Position Plugin:
- Distance Markers and Lines
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/Launch_Site_Location.ipynb

Build a Dashboard with Plotly Dash

- Dropdown List for Launch Site Selection:
- Pie Chart for Successful Launches:
- Slider for Payload Range Selection:
- Scatter Chart for Payload vs. Launch Success:
    - Purpose: To show the correlation between payload mass and launch success.
    - Interaction: The scatter chart updates based on the selected launch site from the dropdown list and the selected payload range from the slider, displaying the relationship between payload mass and launch success for the filtered data.
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/Spacex_Dash_App.py

Predictive Analysis (Classification)

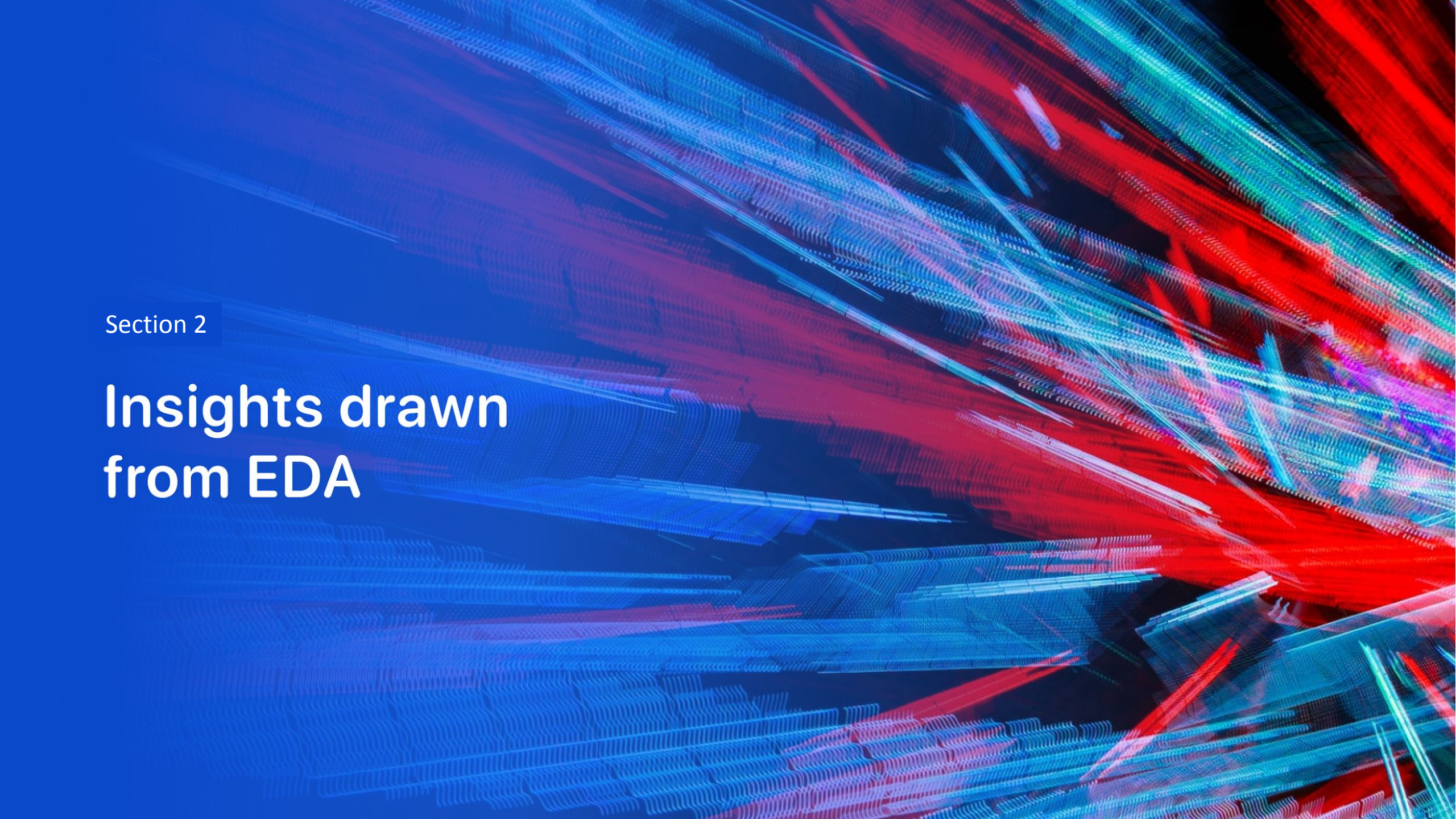
- Logistic Regression
- Support Vector Machine (SVM)
- Decision Tree
- K-Nearest Neighbors (KNN)
- Model Evaluation: Calculated accuracy and plotted confusion matrices for each model.
- https://github.com/krzysztofpk14/IMB-Data-Science-Proffesional-Certificate/blob/main/Data%20Science%20Project/SpaceX_Machine%20Learning%20Prediction.ipynb

Results

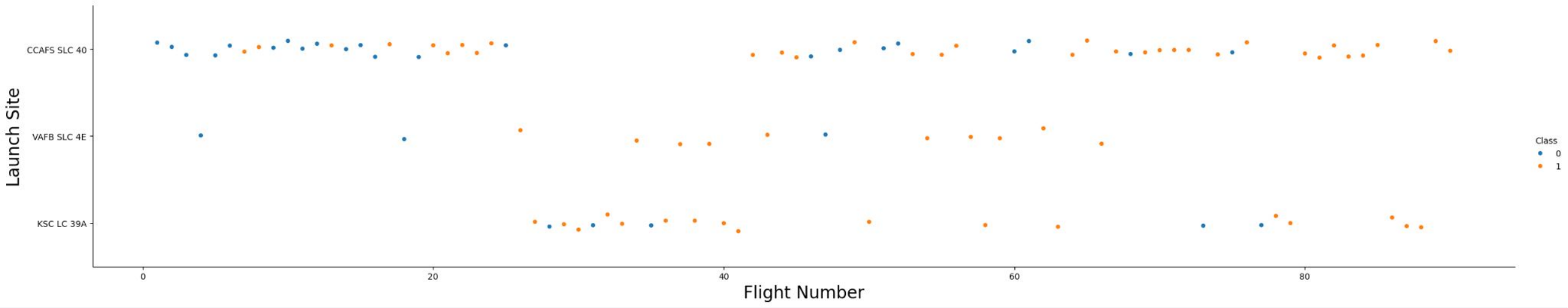
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

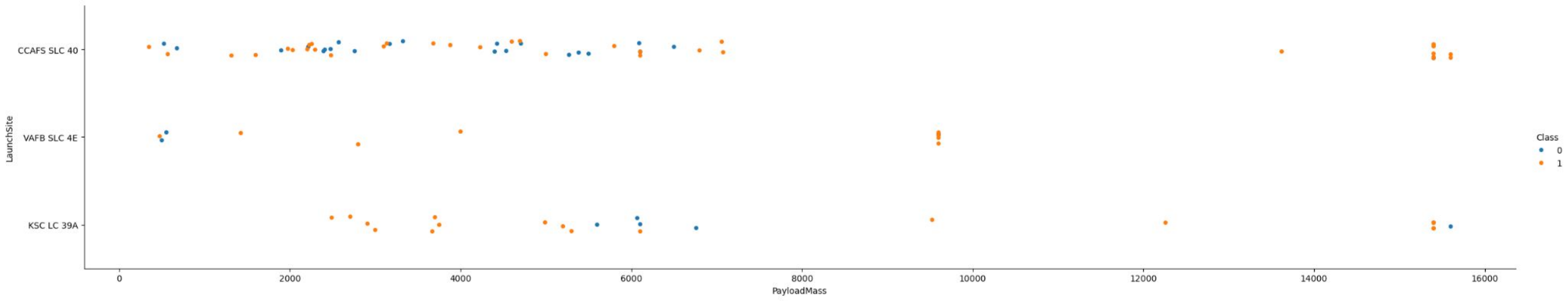
Insights drawn from EDA



Flight Number vs. Launch Site

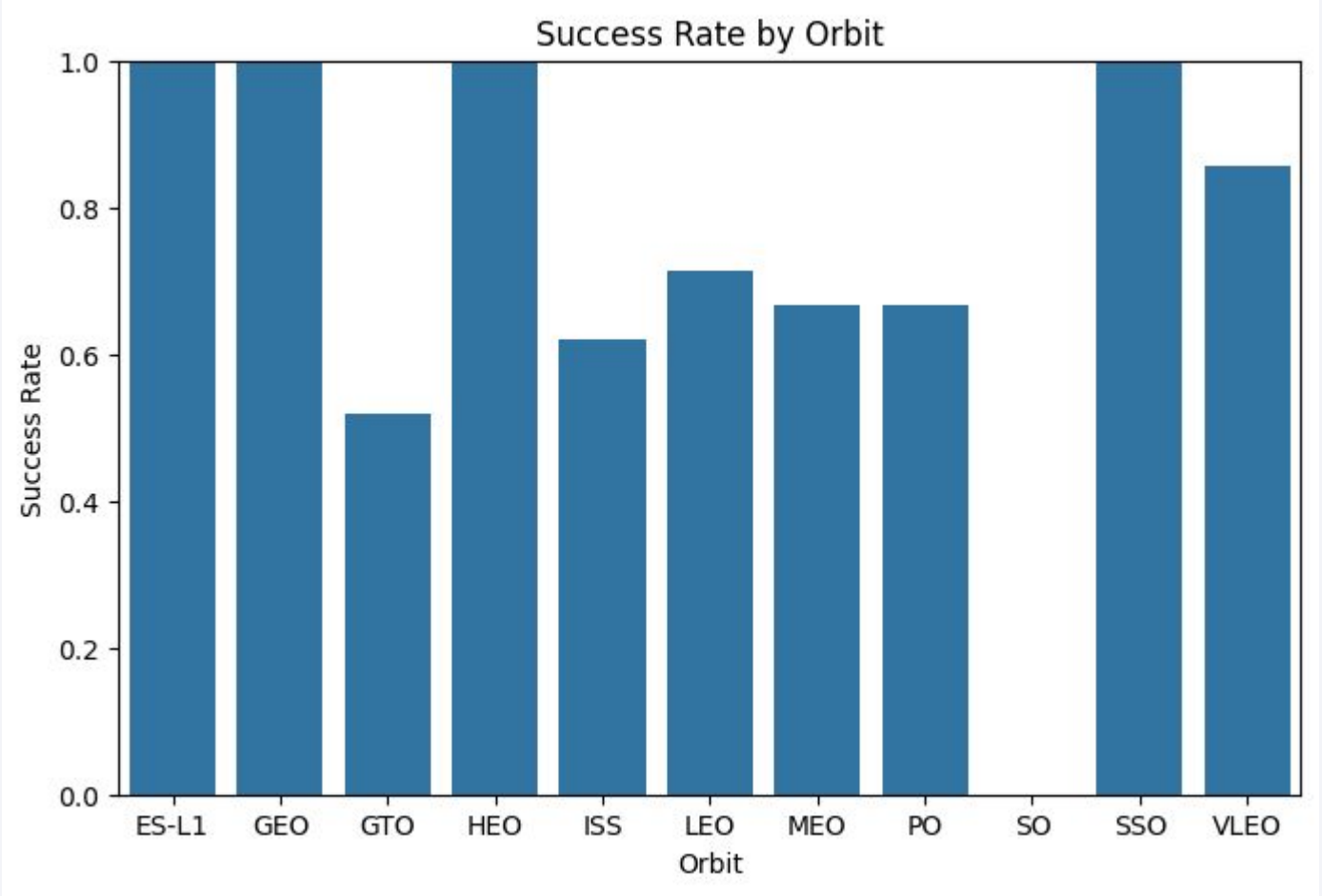


Payload vs. Launch Site



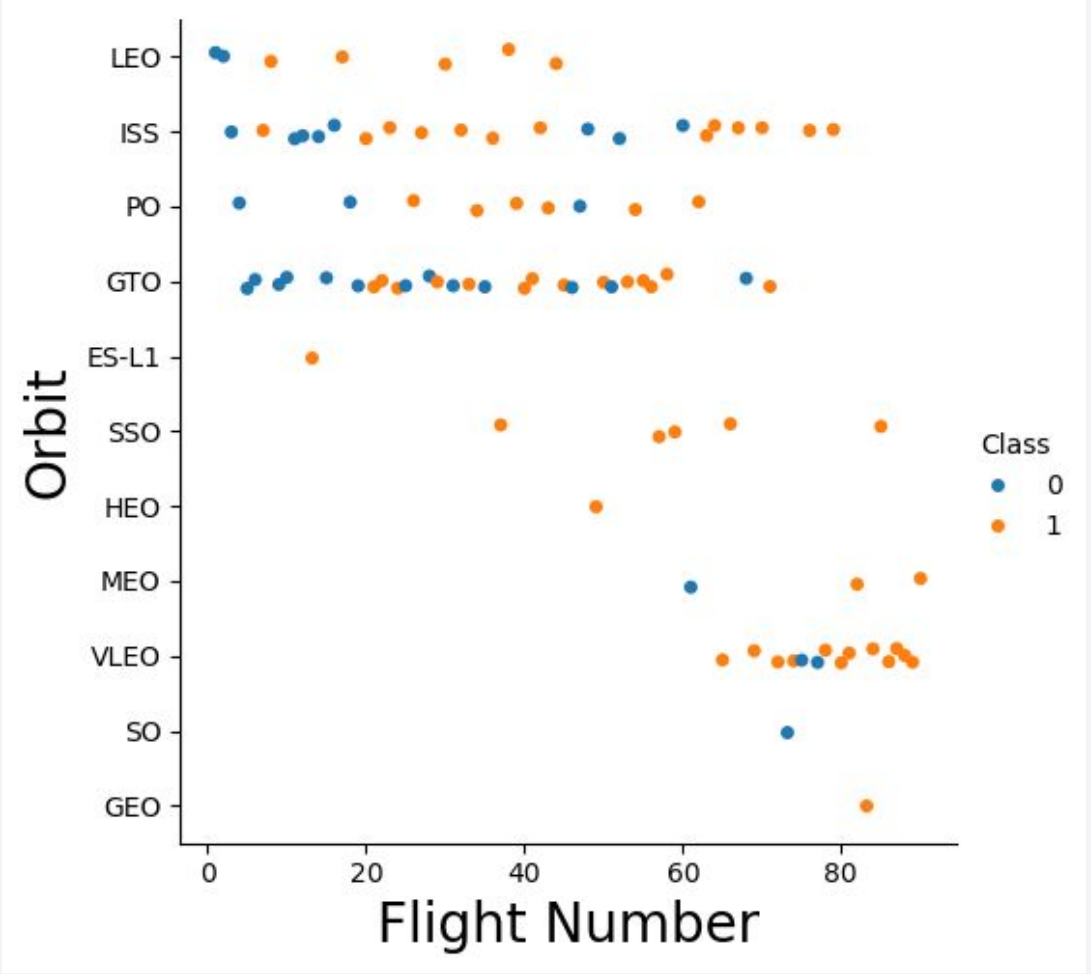
Success Rate vs. Orbit Type

- The biggest success rate have orbits:
  - ES-L1
  - GEO
  - HEO
  - SSO

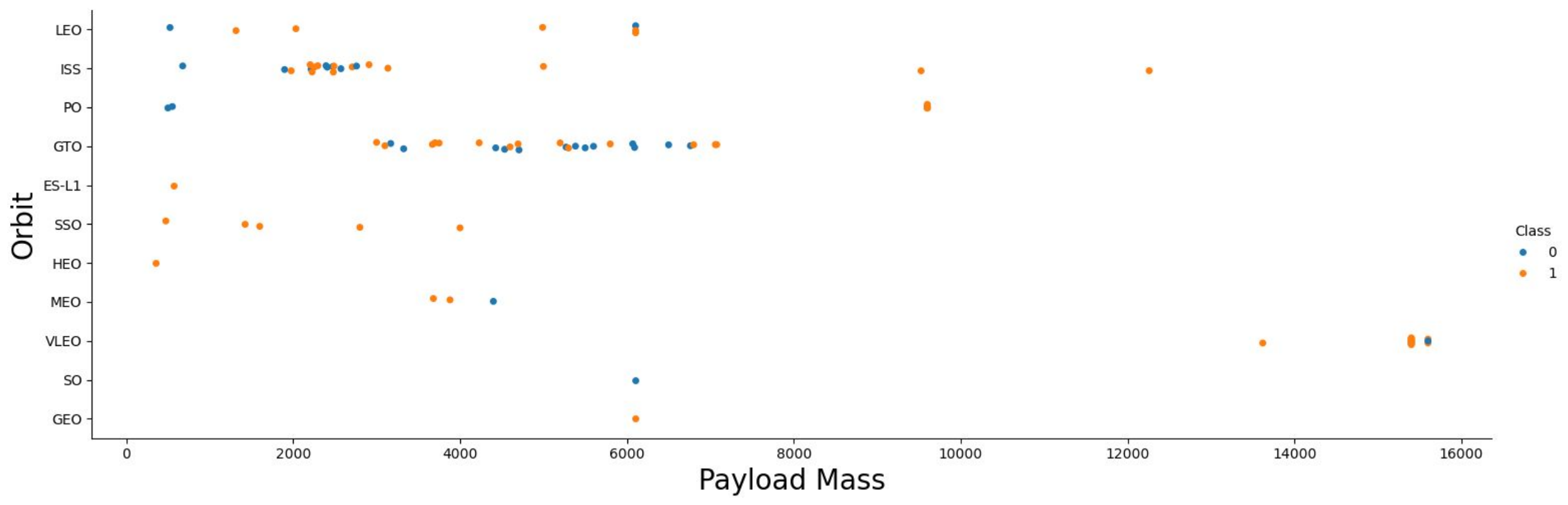


Flight Number vs. Orbit Type

- In almost all orbits, success rate increases as Flight Number increases

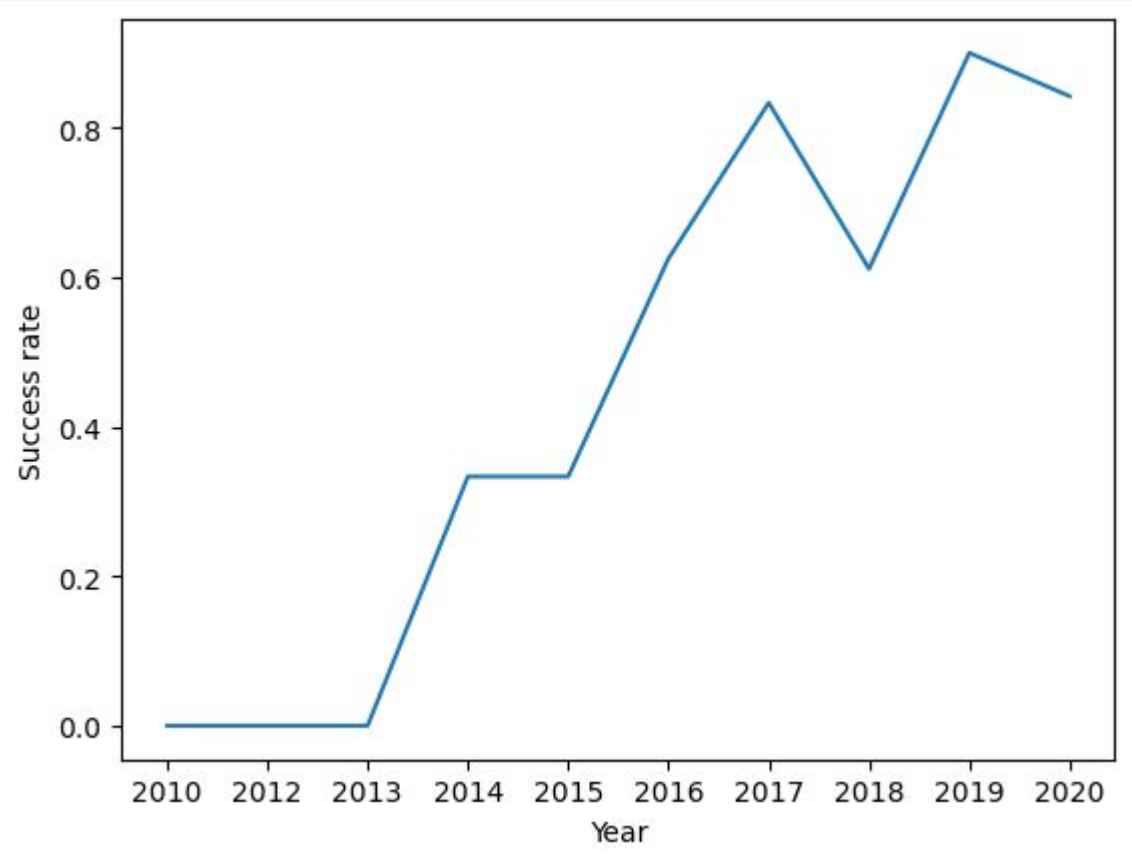


Payload vs. Orbit Type



Launch Success Yearly Trend

- Success rate increases in time. This may be due to experience gained with each new rocket launch.



All Launch Site Names

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

```
[10]  1  %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 Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- select * from spacextable where "Launch_Site" like 'CCA%' limit 5

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

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

```
1  %sql select sum("PAYLOAD_MASS__KG_") from spacextable where "Customer" == 'NASA (CRS)'
```

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

sum("PAYLOAD_MASS__KG_")
45596

Average Payload Mass by F9 v1.1

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

```
1  %sql select avg("PAYLOAD_MASS__KG_") from spacextable where "Booster_Version" like "F9 v1.1%"
```

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

avg("PAYLOAD_MASS__KG_")
2534.6666666666665

First Successful Ground Landing Date

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

Hint:Use min function

```
1  %sql select min("Date") from spacextable where "Landing_Outcome" = 'Success'
```

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

min("Date")
2018-07-22

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

```
1  %sql select "Booster_Version" from spacextable where (PAYLOAD_MASS__KG_ between 4000 and 6000)  and ("Landing_Outcome" =
```

Pytho

```
 * 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

List the total number of successful and failure mission outcomes

```
%%sql

SELECT
    COUNT(CASE WHEN "Mission_Outcome" = 'Success' THEN 1 END) AS Successful_Missions,
    COUNT(CASE WHEN "Mission_Outcome" like 'Failure%' THEN 1 END) AS Failed_Missions
FROM spacextable;
```

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

Successful_Missions	Failed_Missions
98	1

Boosters Carried Maximum Payload

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

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

```
1  %sql select "Booster_version" from spacextable where "PAYLOAD_MASS__KG_" = (select max(PAYLOAD_MASS__KG_) from spacextable)
```

Python

2015 Launch Records

- %sql select * from spacextable where substr(Date,0,5)='2015'

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2015-01-10	9:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)
2015-02-11	23:03:00	F9 v1.1 B1013	CCAFS LC-40	DSCOVR	570	HEO	U.S. Air Force NASA NOAA	Success	Controlled (ocean)
2015-03-02	3:50:00	F9 v1.1 B1014	CCAFS LC-40	ABS-3A Eutelsat 115 West B	4159	GTO	ABS Eutelsat	Success	No attempt
2015-04-14	20:10:00	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)
2015-04-27	23:03:00	F9 v1.1 B1016	CCAFS LC-40	Turkmen 52 / MonacoSAT	4707	GTO	Turkmenistan National Space Agency	Success	No attempt
2015-06-28	14:21:00	F9 v1.1 B1018	CCAFS LC-40	SpaceX CRS-7	1952	LEO (ISS)	NASA (CRS)	Failure (in flight)	Precluded (drone ship)
2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

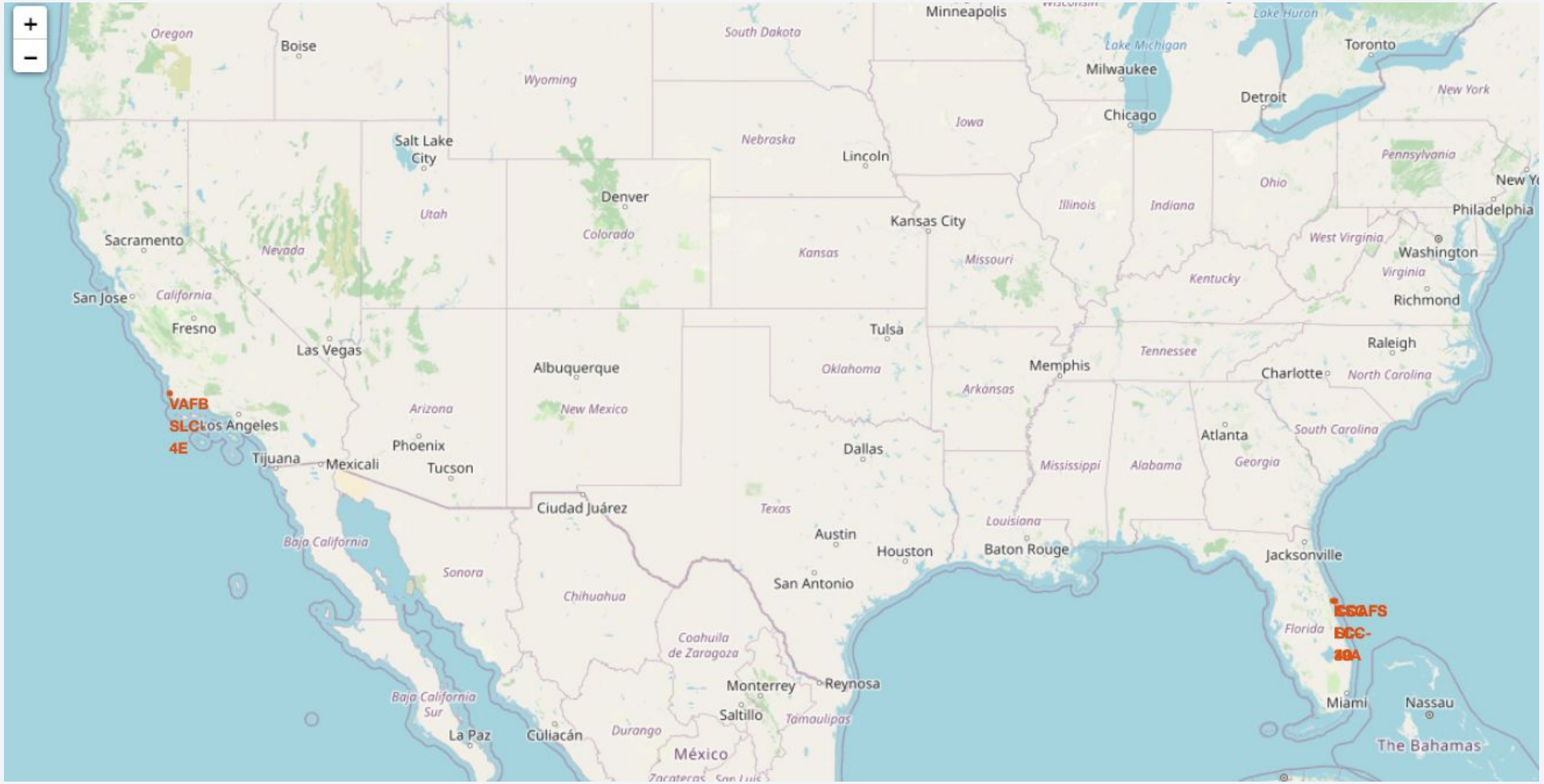
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- %sql select "Landing_Outcome", count("Landing_Outcome") from spacextable where Date between '2010-06-04' and '2017-03-20' group by "Landing_Outcome" order by count("Landing_Outcome") desc

Landing_Outcome	count("Landing_Outcome")
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

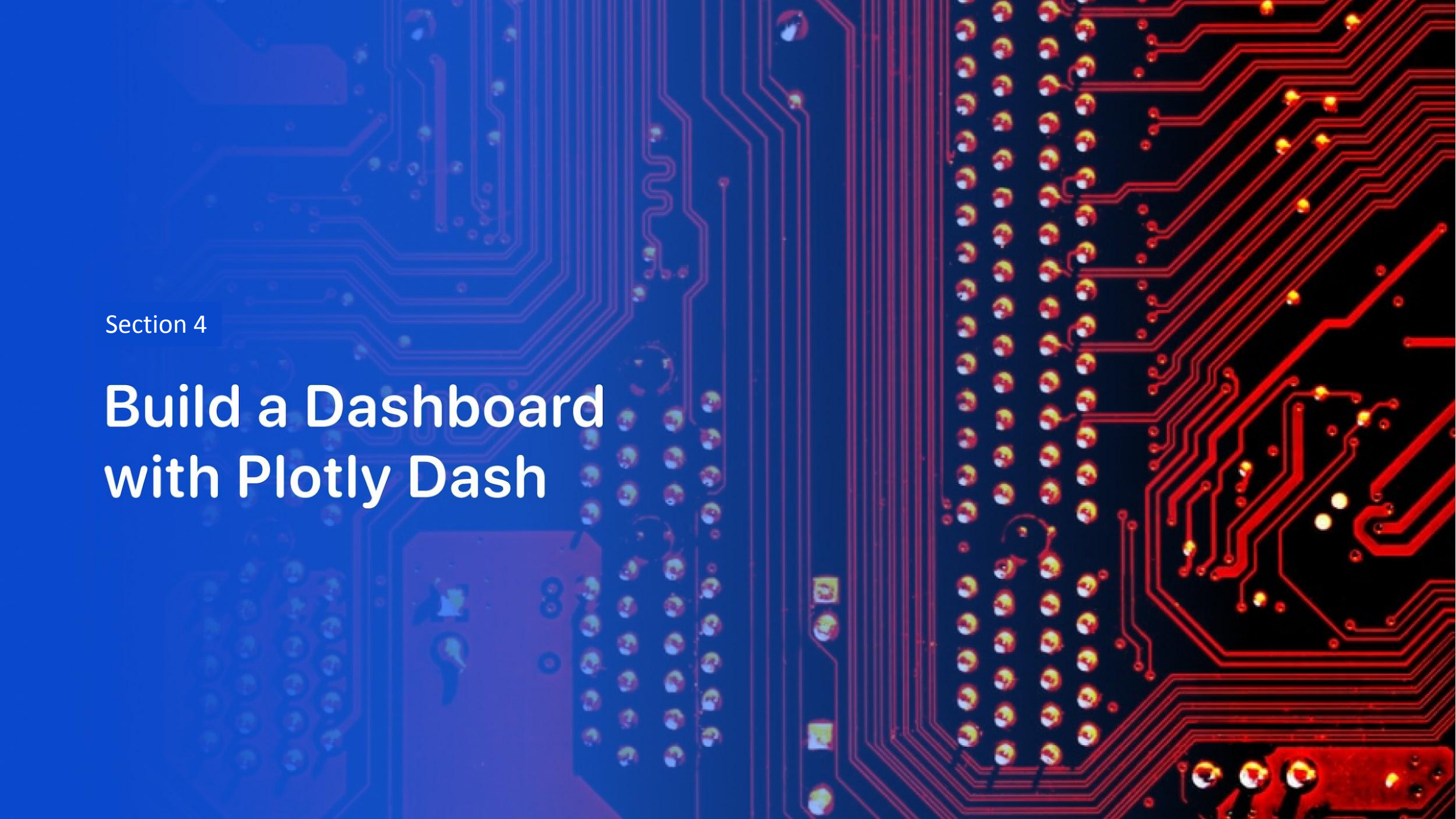

Section 3
Launch Sites
Proximities Analysis

All Launch Sites

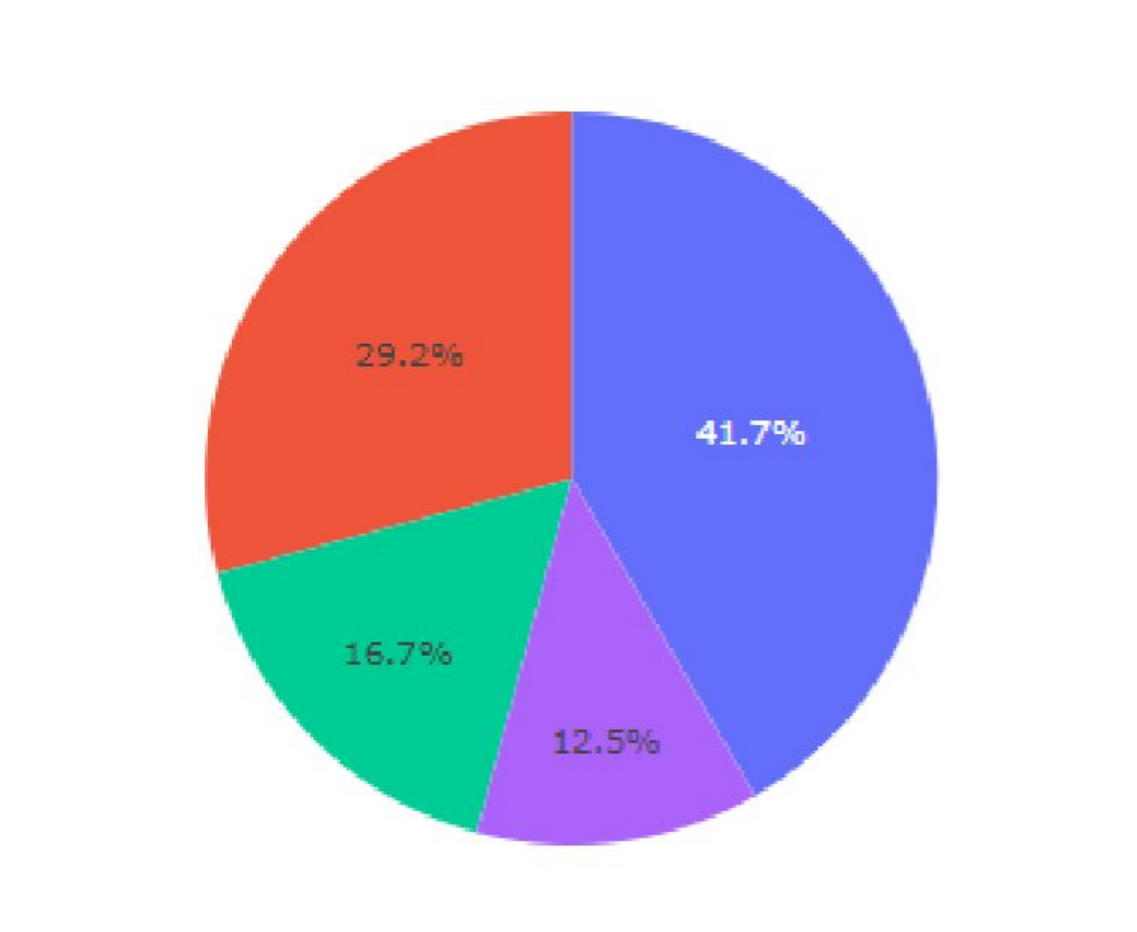


Launch Outcomes on Map

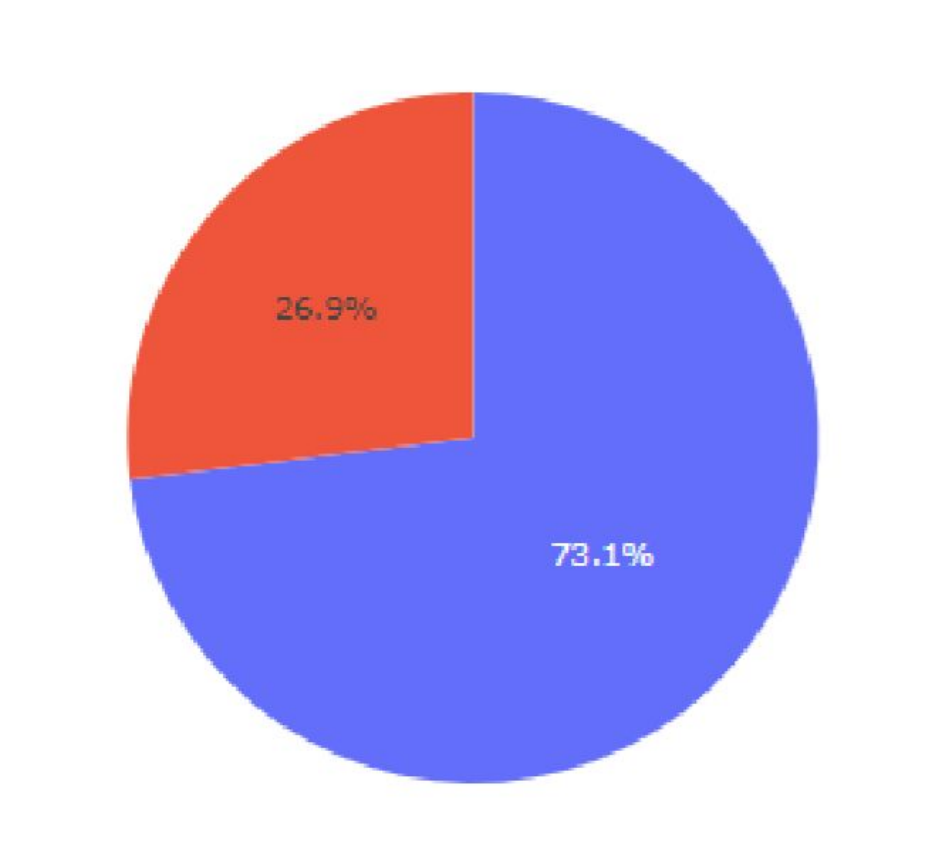
Distances from Coastline


Section 4
Build a Dashboard with Plotly Dash

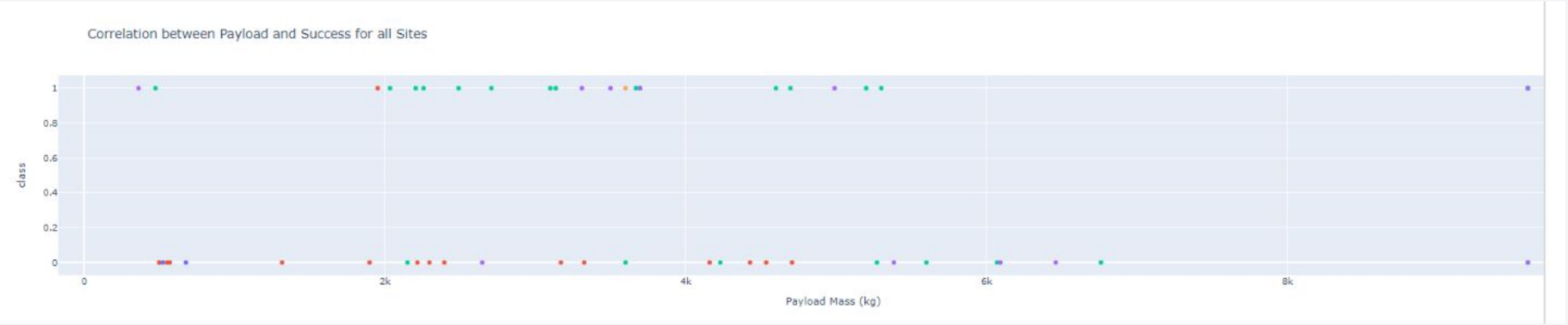
Success count for all sites



Highest lauch succes rate



Payload Mass vs. Outcome

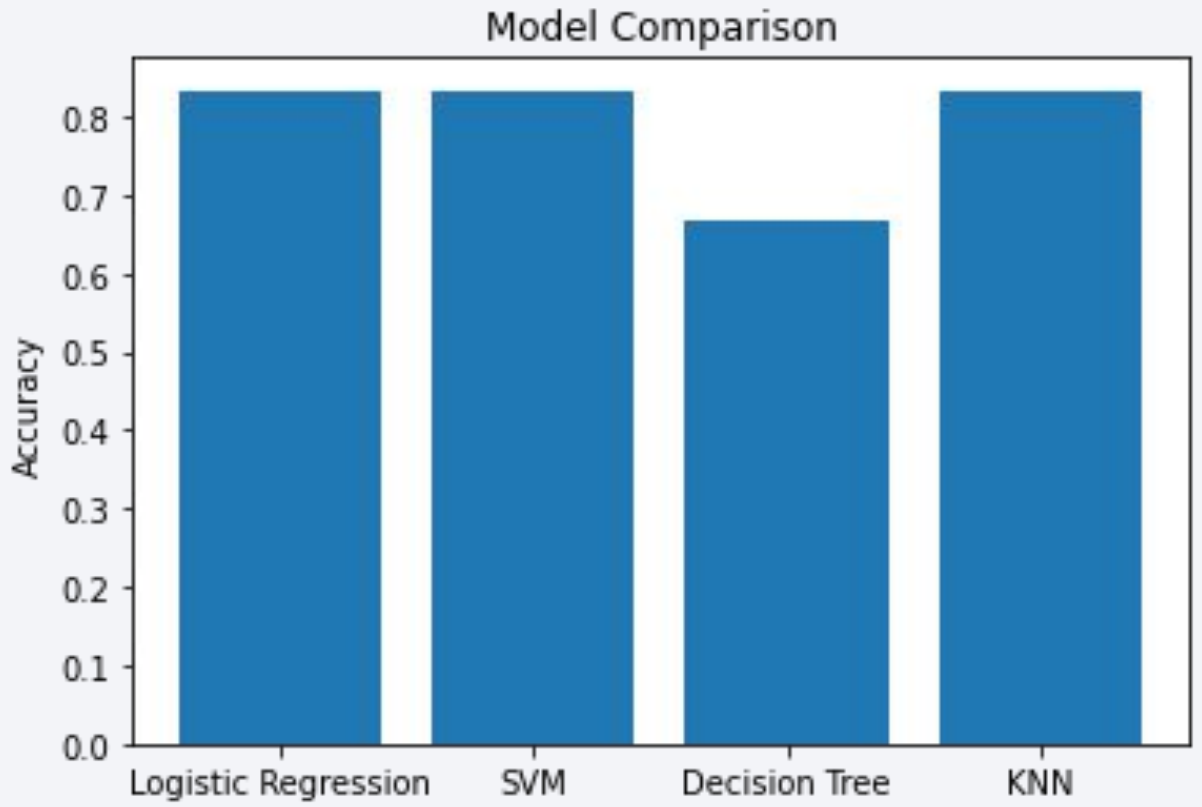


Section 5

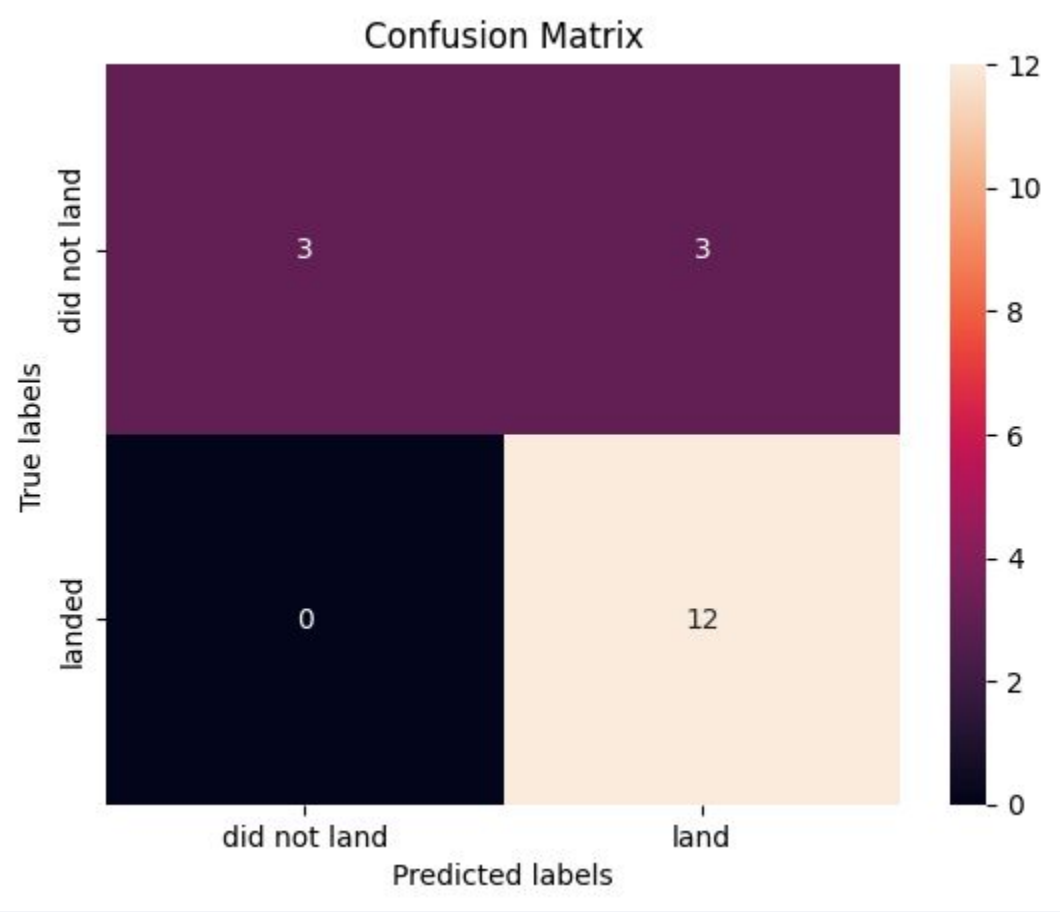
Predictive Analysis (Classification)

Classification Accuracy

- The highest accuracy had:
  - Logistics regression
  - SVM
  - KNN



Confusion Matrix



Conclusions

- Comprehensive Data Collection: Successfully collected and processed data from multiple sources, including web scraping and API requests, to build a robust dataset for analysis.
- Effective Data Visualization: Utilized various charts and maps to perform exploratory data analysis, uncovering key insights and patterns in the data.
- Interactive Dashboard: Developed an interactive dashboard using Dash to visualize SpaceX launch records, enabling dynamic exploration of launch site performance and payload impacts.
- Machine Learning Models: Built and evaluated multiple machine learning models (Logistic Regression, SVM, Decision Tree, KNN) to predict the success of SpaceX Falcon 9 landings, achieving similar accuracy scores of 83.3%.
- Practical Application: Demonstrated the practical application of data science techniques to solve real-world problems, showcasing the skills and knowledge gained throughout the IBM Data Science Professional Certificate program.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!