IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Edward Yeung
16 Oct 2024

Outline

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

Executive Summary

- **Summary of methodologies**
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- **Summary of all results**
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result

Introduction

- **Project background and context:**
  - SpaceX offers Falcon 9 rocket launches for $62 million, compared to $165 million fro m other providers, mainly because SpaceX can reuse the first stage. Predicting the fir st stage's landing success helps estimate launch costs, crucial for companies bidding against SpaceX. The project aims to create a machine learning pipeline to predict first stage landing success.
- **Problems you want to find answers:**
  - What factors influence a rocket's successful landing?
  - How do various features interact to determine landing success rates?
  - What operating conditions ensure a successful landing program?

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- 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

- **Data Collection and Preparation:**
- **API Requests**: We collected data using GET requests to the SpaceX API.
- **JSON Parsing**: The response content was decoded as JSON using the .json() function and converted into a pandas DataFrame with .json_normalize().
- **Data Cleaning**: We cleaned the data, checked for missing values, and filled them where n ecessary.
- **Web Scraping**

Data Collection – SpaceX API

- **Objective**: To gather comprehensive data on SpaceX launches.
- **Tool**: Utilized the SpaceX API for efficient data retrieval.
- **Process**:
  - API Requests: Sent GET requests to the SpaceX API to fetch the latest launch data.

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

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

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

Lab 1:https://github.com/EdwardYeung6/IBM-Applied-Data-Science-Capstone/blob/c3308c3b5542a299e24ac04c51910a3e089f55b3/01-spacex-data-collection-api.ipynb

Data Collection - Scraping

- **Objective**: To extract additional data from the Space X website for comprehensive analysis.
- **Tool**: Utilized BeautifulSoup for web scraping.
- **Process**:
  - **Initiated Requests**: Sent GET requests to the website to fetch HTML content.
  - **Parsed HTML**: Used BeautifulSoup to parse the HTML structure.
  - **Data Extraction**: Identified and extracted relevant data elements (such as launch records) from the HTML.
  - **Transformation**

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

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

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data)
```

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

```
# Use soup.title attribute
print(soup.title)
```

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

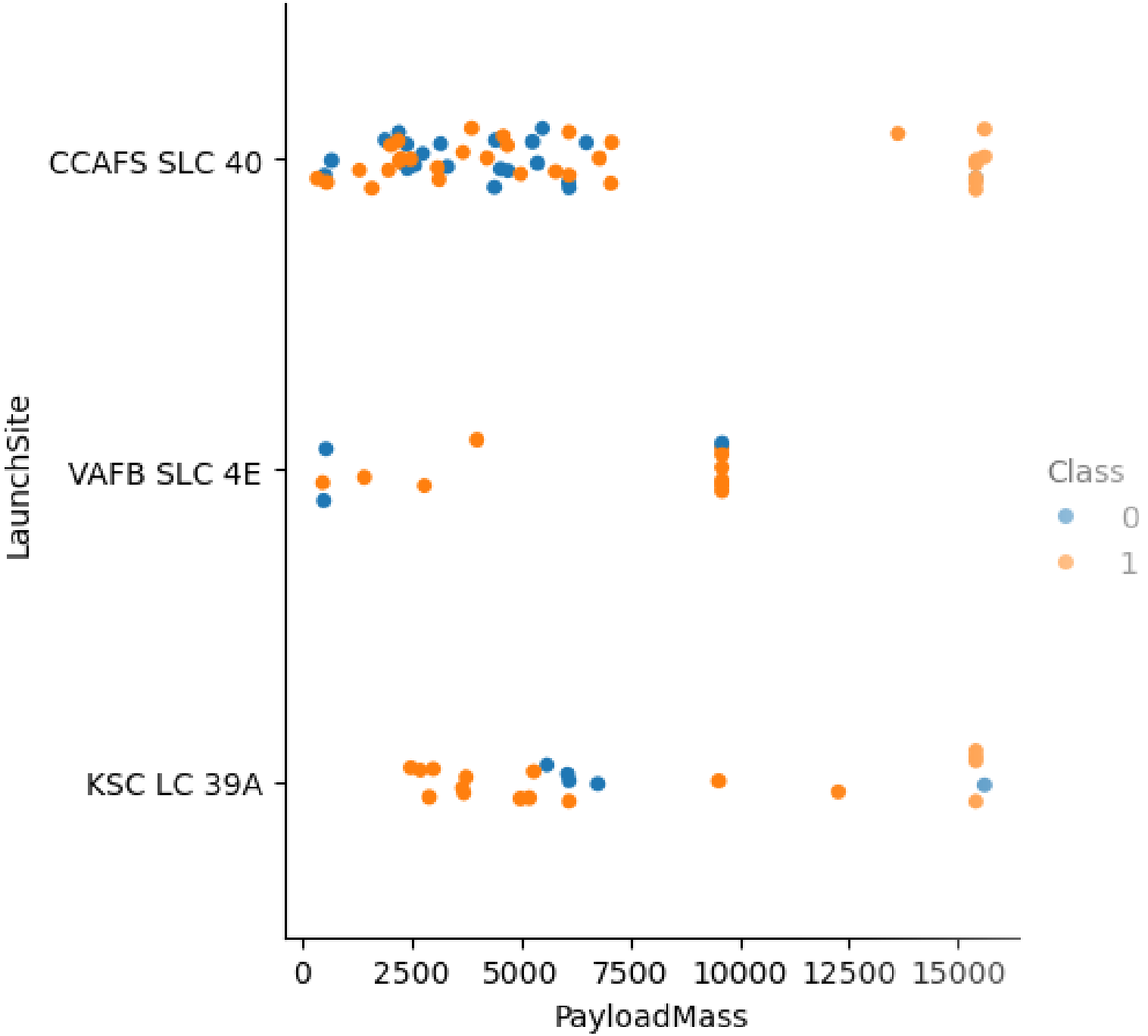
Data Wrangling

- **Objective**: To prepare and refine data for analysis.
- **Process**:
  - **Cleaning**: Removed irrelevant data, handled miss ing values, and ensured data consistency.
  - **Formatting**: Organized and formatted the data to make it suitable for further analysis.

```
# 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)
```

EDA with Data Visualization

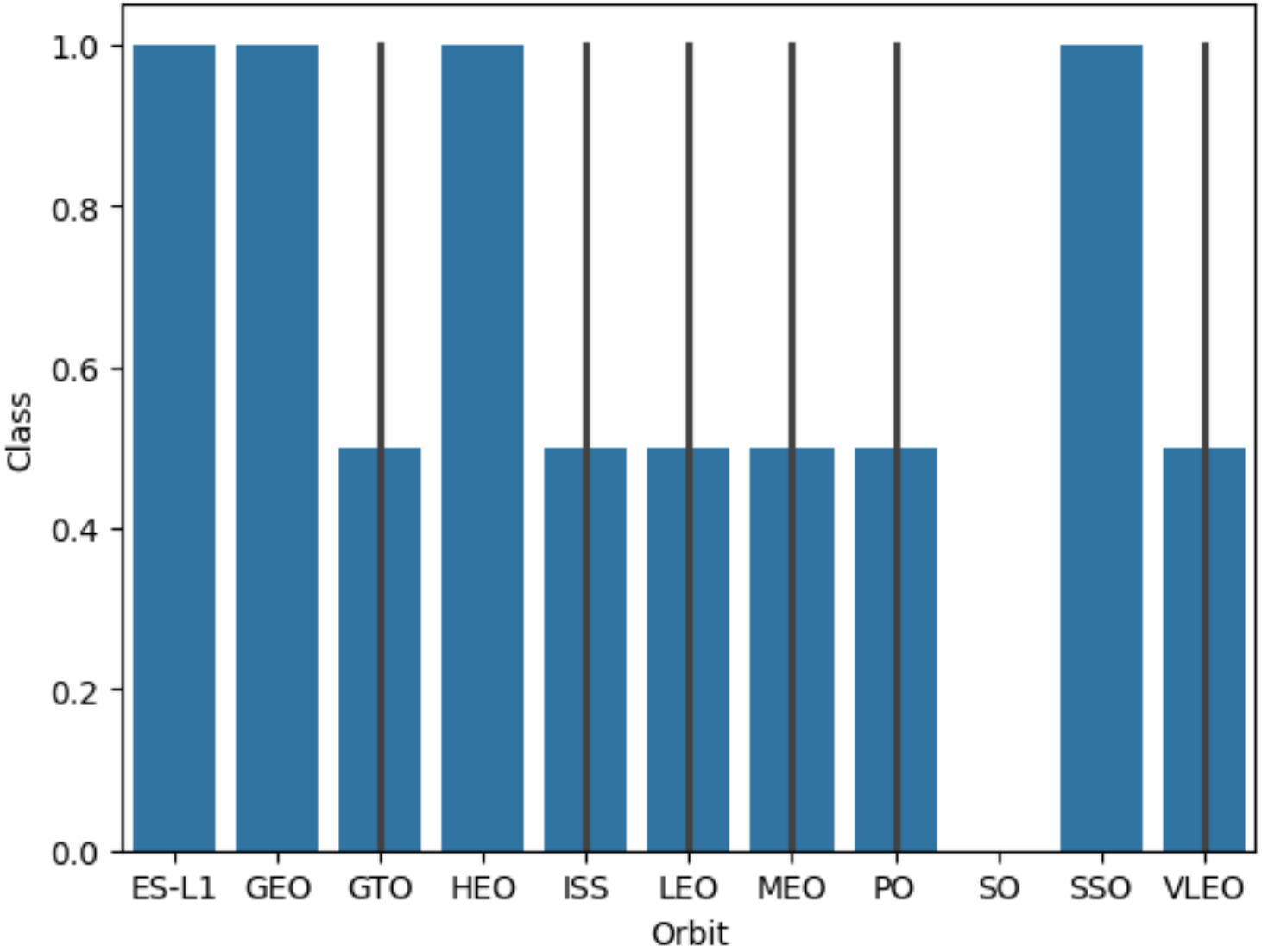
- **Objective**: To visually explore and understand the SpaceX API data.
- **Approach**:
  - **Data Plotting**: Used libraries like Matplotlib and Seaborn to create insightful graphs and chart s.
  - **Trends and Patterns**: Identified key trends and patterns in launch data, payloads, and success rates.
  - **Graph Types**: Created various visualizations such as bar charts, scatter plots, and histograms t o uncover insights.
  - **Insights**: Visual representations highlighted relationships between launch sites and payload mass, success rates over time, and more.
- **Results**: Derived critical insights, making complex data more understandable and actionable.



EDA with Data Visualization

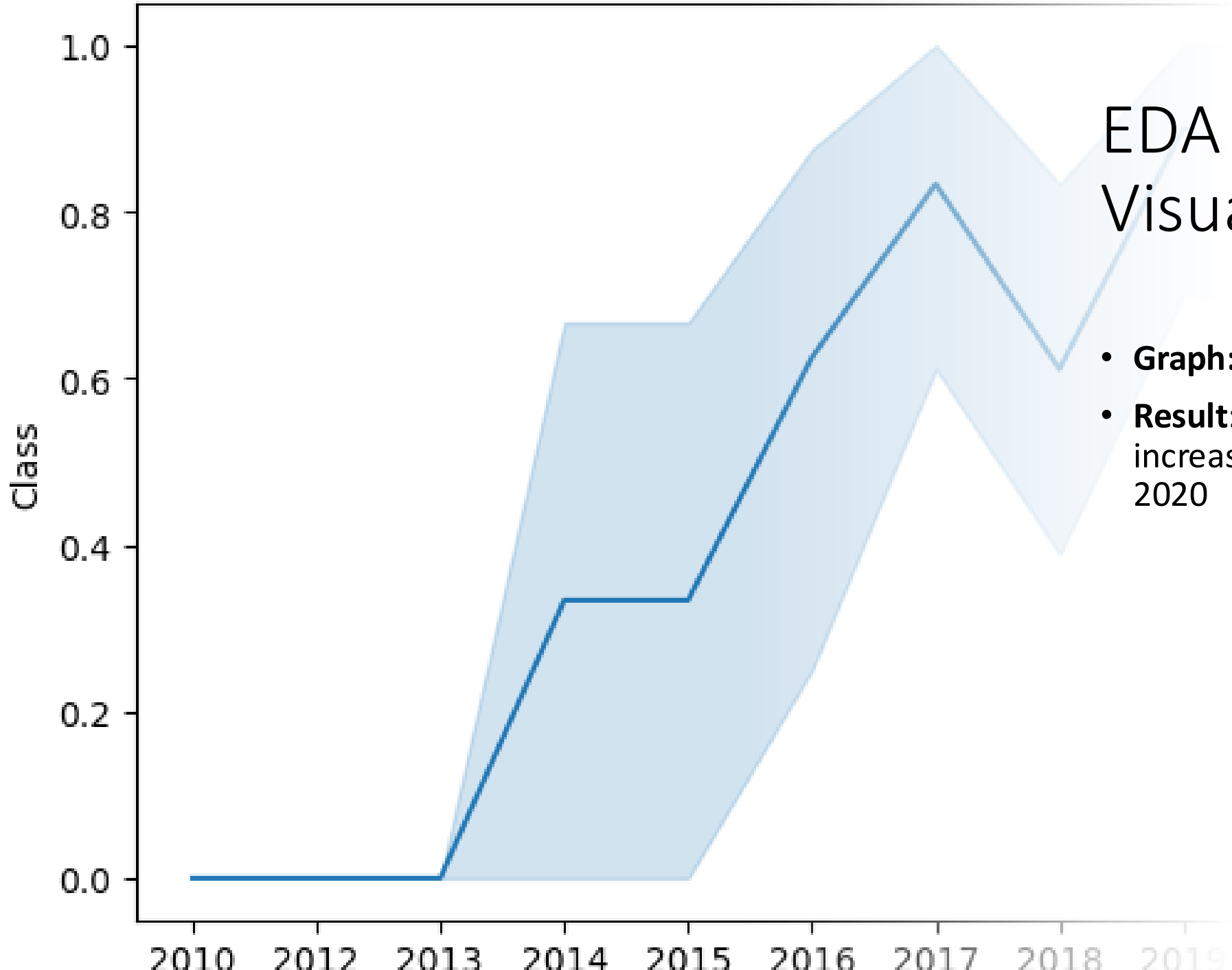
- **Graph: Launch Site VS Payload Mass**
- **Result**: No rockets were launched for payloads greater than 10,000 kg.

EDA with Data Visualization



- **Graph: Class VS Orbit**
- **Result**: ES-L1, GEO, HEO, SSO, and VLEO showed the highest success rates.

EDA with Data Visualization



- **Graph: Class VS Date**
- **Result**: Sucess rate were increasing since 2013 kept till 2020

EDA with SQL

- **Objective**: To explore and analyze the collected data.
- **SQL Queries**: Used SQL to perform exploratory data analysis (EDA) on the SpaceX API dat aset.
  - Queried for summary statistics and distribution patterns.
  - Identified key metrics like payload mass, launch success rates, and site-specific performance.
  - Aggregated and visualized data to uncover trends and insights.
- **Insights**: Derived critical information on launch outcomes, payload distribution, and oper ational efficiency.

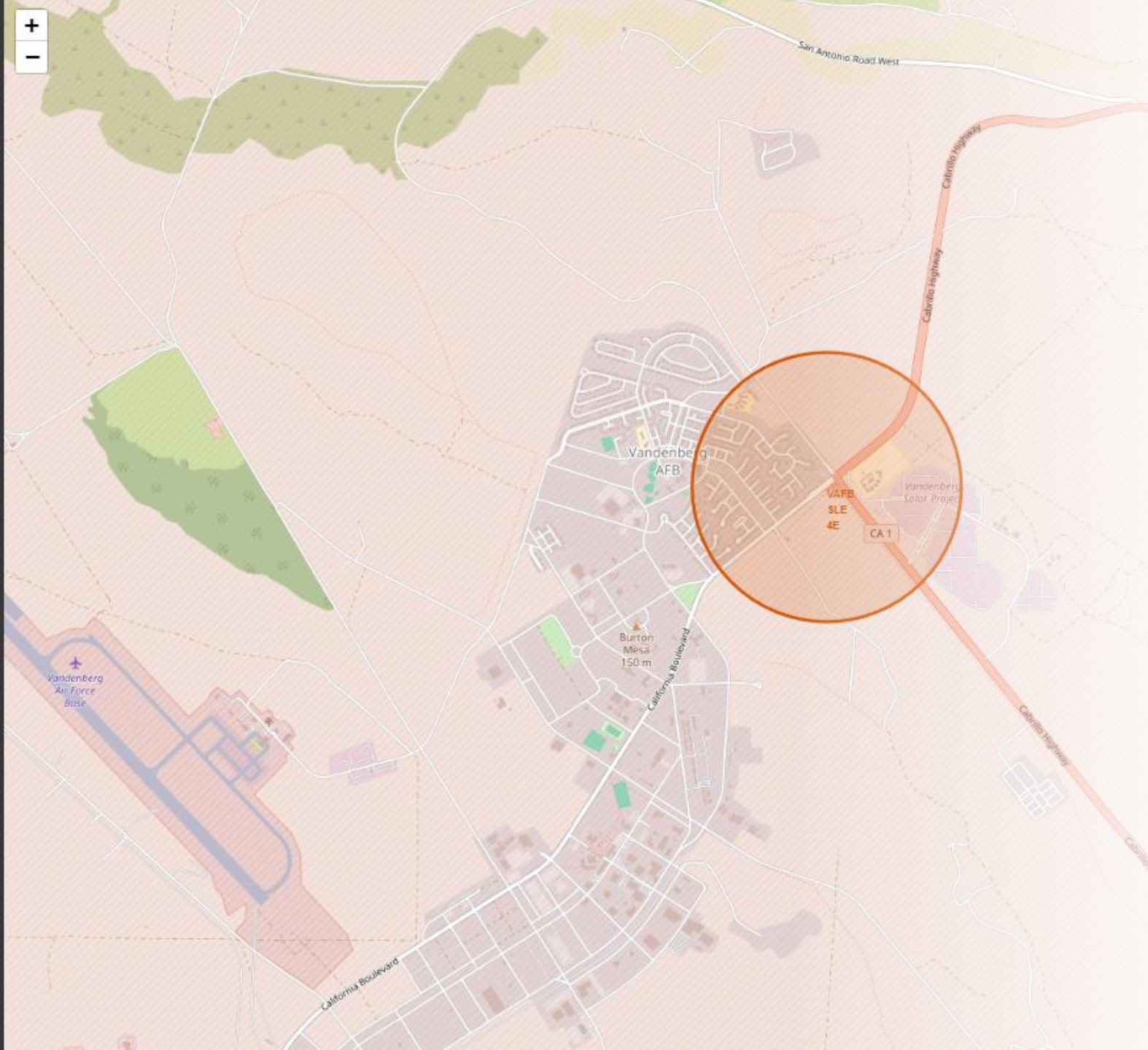
EDA with SQL

- **Objective**: To explore the dataset and identify unique features.
- **Task**: Displayed the names of the unique launch sites in the space mission using SQL.
  - **SQL Query**: Used a DISTINCT query to list all unique launch sites from the dataset.
  - **Result**: Successfully identified and displayed unique launch site names, aiding in further analysis of launch patterns.

```
%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



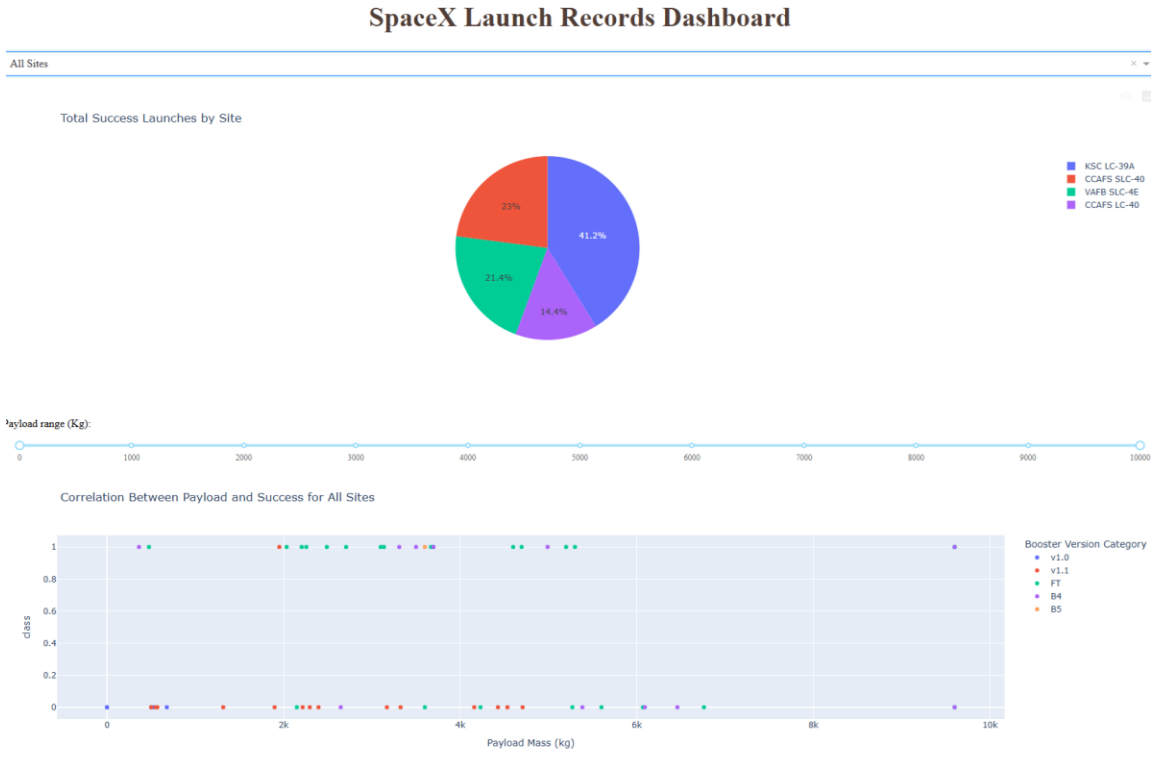
Build an Interactive Map with Folium

- **Objective**: Visualize SpaceX launch sites on an interactive map.
- **Tool**: Folium for geospatial visualization.
- **Steps**:
  - Extracted geospatial coordinates (la titude and longitude) of launch site s.
  - Created a Folium map centered aro und launch locations.
  - Added markers for each site with re levant info.
  - Made markers clickable to display l aunch details.

Build a Dashboard with Plotly Dash

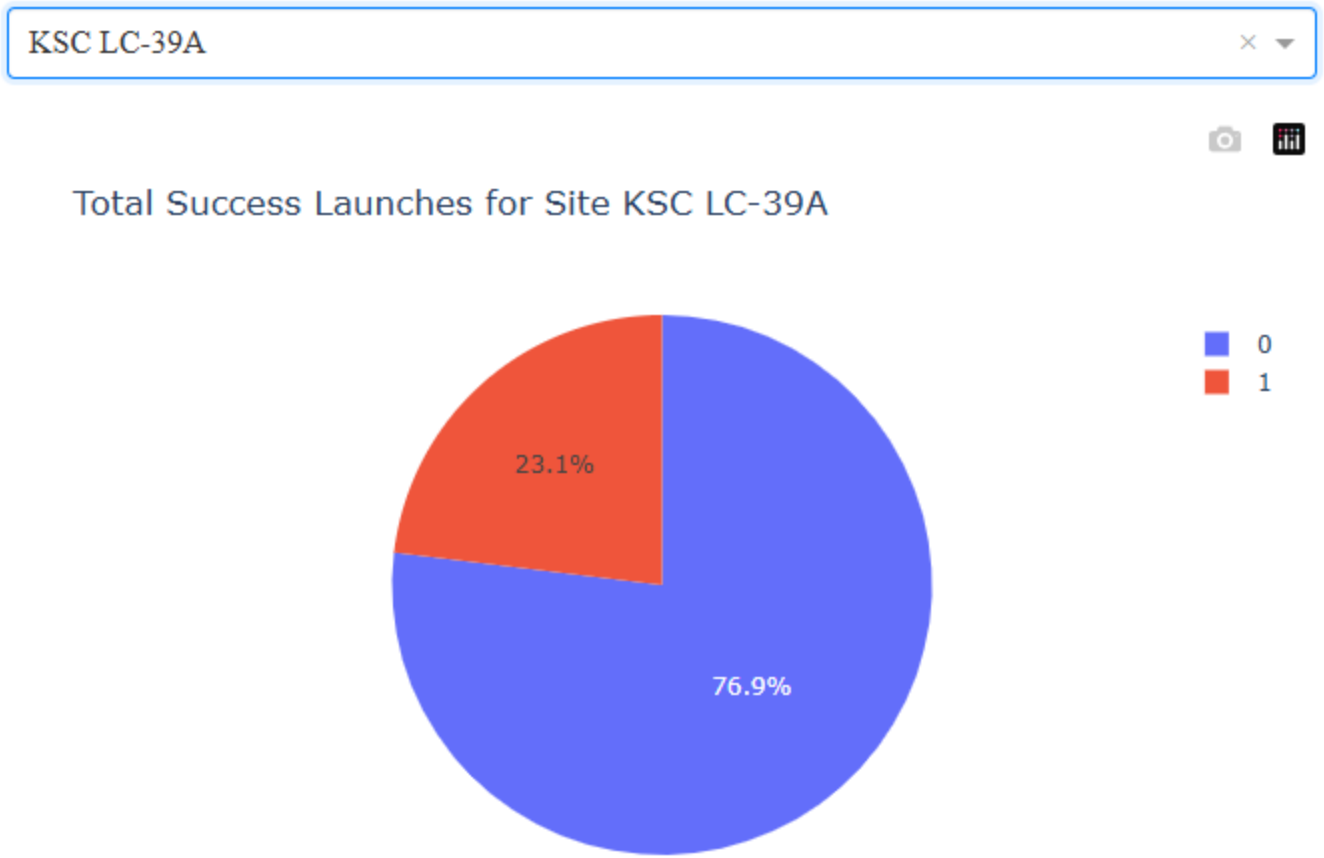
- **Objective**: Create an interactive dashboard for visualizing SpaceX launch data.
- **Tool**: Plotly Dash for building web-based visualizations.
- **Steps**:
  - Designed and implemented the dashboard layout using Dash components.
  - Integrated interactive charts and graphs to display key metrics.
  - Enabled user interactions like filtering and selecting data points.

- **Result:** KSCLC-39A had the most successful launch rate



Build a Dashboard with Plotly Dash

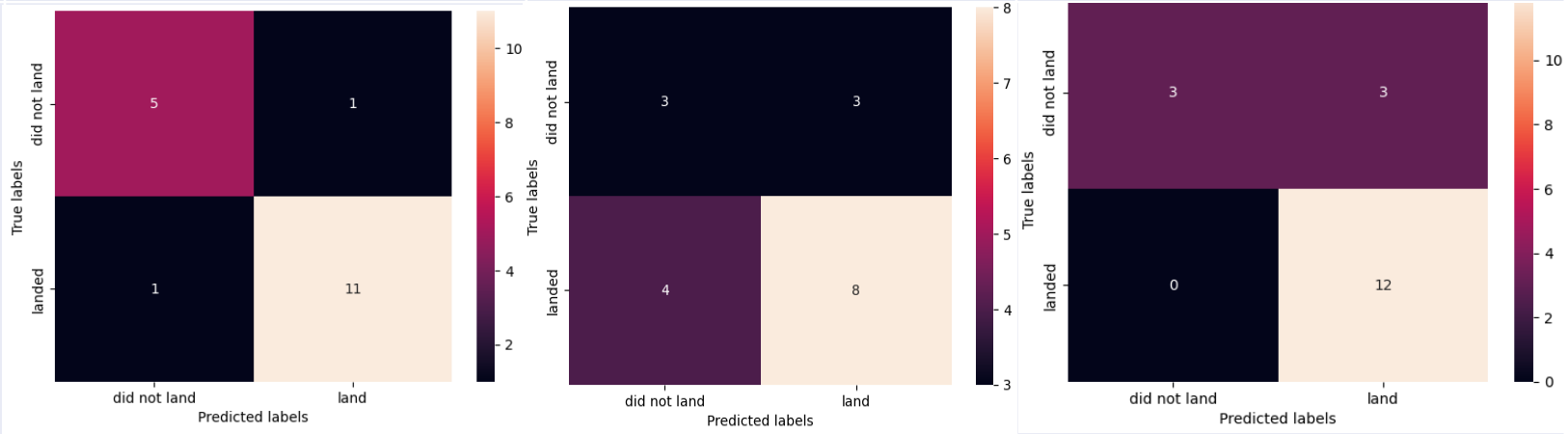
- **Dashboard Filter: KSCLC-39A**
- **Result:** KSCLC-39A achieved 76.9% success rate and 23.1% failure rate



Predictive Analysis (Classification)

- **Objective**: Assess the accuracy of machine learning models.
- **Models Evaluated**:
  - **DecisionTreeClassifier**
  - **KNeighborsClassifier**
  - **LogisticRegression**
- **Method:** Utilized the Accuracy Score, F1 Score, and Confusion Matrix to measure model accuracy on the test dataset.

Predictive Analysis (Classification)

	DecisionTree Classifier	KNeighbors Classifier	Logistic Regression
Accuracy Score (%)	88.8889	69.5652	83.333333
F1_Score	0.916667	0.695652	0.888889
Jaccard_Score	0.846154	0.533333	0.800000
Confusion matrix			

Results

- More flights at a launch site correlate with higher success rates.
- No rockets were launched for payloads greater than 10,000 kg.
- Orbits like ES-L1, GEO, HEO, SSO, and VLEO showed the highest success rates.
- Launch success rates increased steadily from 2013 to 2020.
- KSC LC-39A recorded the most successful launches among all sites.
- The DecisionTree Classifier proved to be the best algorithm for this task.

URL

- https://github.com/EdwardYeung6/IBM-Applied-Data-Science-Capstone.git

Thank you!