IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Tareq Hamad

January, 25th 2025

Outline

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

Executive Summary

- Summary of Methodologies:
  - Data collection via SpaceX API, web scraping, and provided datasets.
  - Data wrangling and processing for cleaning and feature engineering.
  - Exploratory Data Analysis (EDA) using SQL, visualization libraries, and interactive tools like Folium and Plotly Dash.
  - Predictive analysis using classification models with hyperparameter tuning and evaluation.
- Summary of Results:
  - Identified key factors influencing launch success (e.g., payload mass, orbit type).
  - Built interactive visualizations to explore relationships between variables.
  - Developed classification models (Logistic Regression, SVM, Decision Tree, KNN) with the best-performing model achieving high accuracy on test data.

Introduction

- Project background and context:
- Project Background and Context:SpaceX aims to reduce launch costs by reusing rocket boosters. Predicting launch success is critical for cost optimization.
- Problems to Solve:
- What factors influence launch success?
- How can we predict launch outcomes using historical data?

Section 1

Methodology

Methodology

Executive Summary

- **Data collection methodology:**
  - Collected data using SpaceX REST API and web scraping techniques.
  - Processed raw data into structured formats for analysis.
- **Data wrangling**
  - Cleaned data to handle missing values and standardized features.
  - Engineered features for predictive modeling.
- **Exploratory data analysis (EDA) using visualization and SQL:**
  - Visualized relationships between variables using scatter plots, bar charts, and line charts.
  - Queried data using SQL for deeper insights.

Methodology

Executive Summary

- **Interactive visual analytics using Folium and Plotly Dash:**
  - Built interactive maps with Folium to analyze proximities of launch sites to coastlines, highways, etc.
  - Created dashboards with Plotly Dash for real-time exploration of launch success rates.
- **Perform predictive analysis using classification models**
  - Built classification models (Logistic Regression, SVM, Decision Tree, KNN).
  - Tuned hyperparameters using GridSearchCV and evaluated models on validation/test sets.

Data Collection

- Using SpaceX API:
  - Sent REST API calls to retrieve launch records.
  - Parsed JSON responses into structured datasets.
- Web Scraping:
  - Scraped Wikipedia pages for historical Falcon 9 launches using BeautifulSoup.
  - Extracted tables with launch details like payload mass, orbit type, etc.

Data Collection – SpaceX API

Used the get request to collect data, clean the requested data and basic data wrangling & formatting.

GitHub Link - Data Collection API

1. Get request for rocket launch data using API

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

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

2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

Data Collection - Scraping

- Applied web scrapping to Falcon 9 launch records with BeautifulSoup
- Parsed the table and converted it into a pandas dataframe.
- [GitHub Link - WebScraping HTML]()

1. Apply HTTP Get method to request the Falcon 9 rocket launch page

```
In [4]: static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

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

Out[5]: 200
```

2. Create a BeautifulSoup object from the HTML response

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

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

```
In [7]: # Use soup.title attribute
        soup.title

Out[7]: <title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

3. Extract all column names from the HTML table header

```
In [10]: column_names = []

         # Apply find_all() function with `th` element on first_launch_table
         # Iterate each th element and apply the provided extract_column_from_header() to get a column name
         # Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names

         element = soup.find_all('th')
         for row in range(len(element)):
             try:
                 name = extract_column_from_header(element[row])
                 if (name is not None and len(name) > 0):
                     column_names.append(name)
             except:
                 pass
```

4. Create a dataframe by parsing the launch HTML tables
5. Export data to csv

Data Wrangling

- Cleaned datasets by removing duplicates and handling missing values.
- Standardized numerical features (e.g., payload mass) using StandardScaler.
- Encoded categorical variables (e.g., orbit type) using one-hot encoding.
- GitHub Link - Data Wrangling

EDA with Data Visualization

- Charts Plotted:Scatter plots for Flight Number vs. Launch Site, Payload vs. Launch Site.
- Bar charts for success rates by orbit type.
- Line charts for yearly trends in success rates.
- Purpose of Charts:Identify patterns in launch success based on payload mass, orbit type, and time.
- GitHub Link - EDA SQL with Data Visualization

EDA with SQL

- Queried data to:
  - Find unique launch sites and calculate payload statistics by booster version.
- Identify successful/failed outcomes by date range.
- [GitHub Link - EDA with SQL]

Build an Interactive Map with Folium

- Map Objects Added:Markers for each launch site with labels showing names.
  - Circles to highlight proximities to coastlines and highways.
  - Lines connecting launch sites to nearby infrastructures.
- Purpose:
  - Analyze geographical factors influencing launch outcomes.
- GitHub Link - Interactive Map w/ Folium

Build a Dashboard with Plotly Dash

- Plots/Graphs Added:
  - Pie chart showing success rates by site.
  - Scatter plot for payload vs. success outcomes.
- Purpose:
  - Enable users to interactively explore relationships between variables.
- GitHub Link - Dashboard w/ Plotly Dash

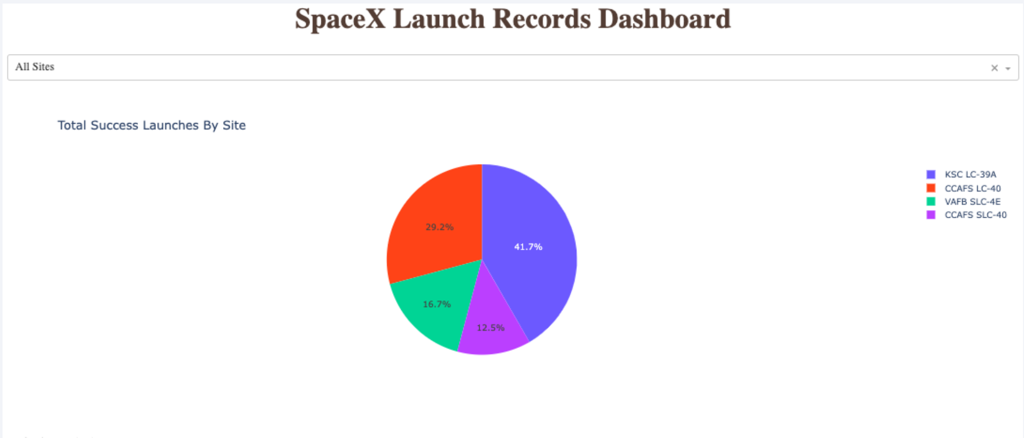
Predictive Analysis (Classification)

1. Built Logistic Regression, SVM, Decision Tree, and KNN models.
2. Tuned hyperparameters using GridSearchCV with cross-validation.
3. Evaluated models on test data using accuracy scores and confusion matrices.

GitHub Link - Machine Learning Prediction

Results

- EDA Results:
  - Higher flight numbers correlate with increased success rates.
  - Payload mass affects success probabilities differently across orbit types.
- Interactive Analytics Demo:
  - Screenshots of Folium maps showing proximities of launch sites to coastlines/highways.
  - Screenshots of Plotly Dash dashboard visualizations.
- Predictive Analysis Results:
  - Best-performing model achieved high accuracy on test data (87.7%).

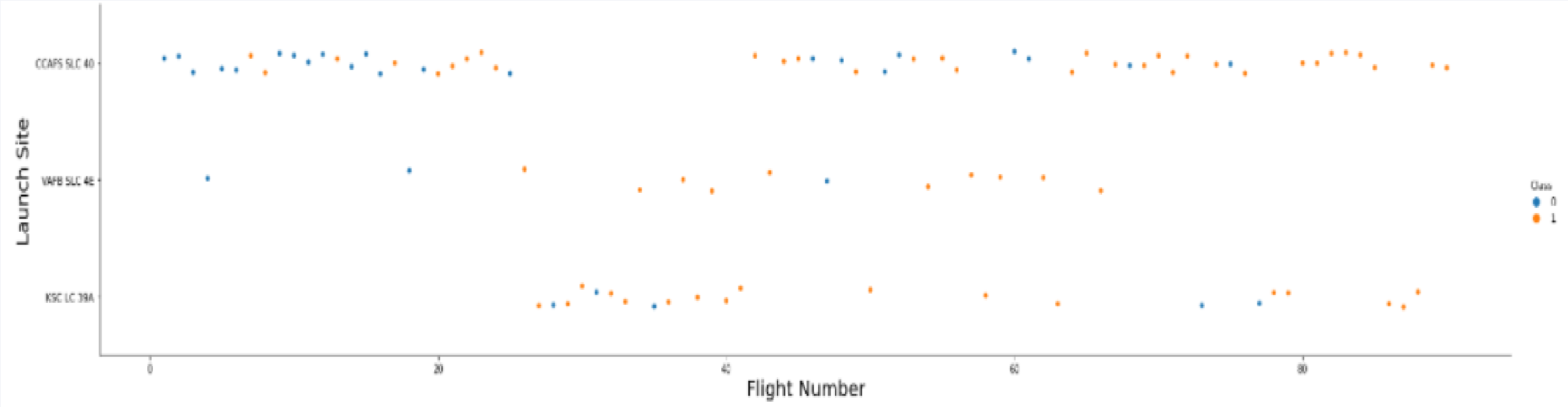


Section 2

Insights drawn from EDA

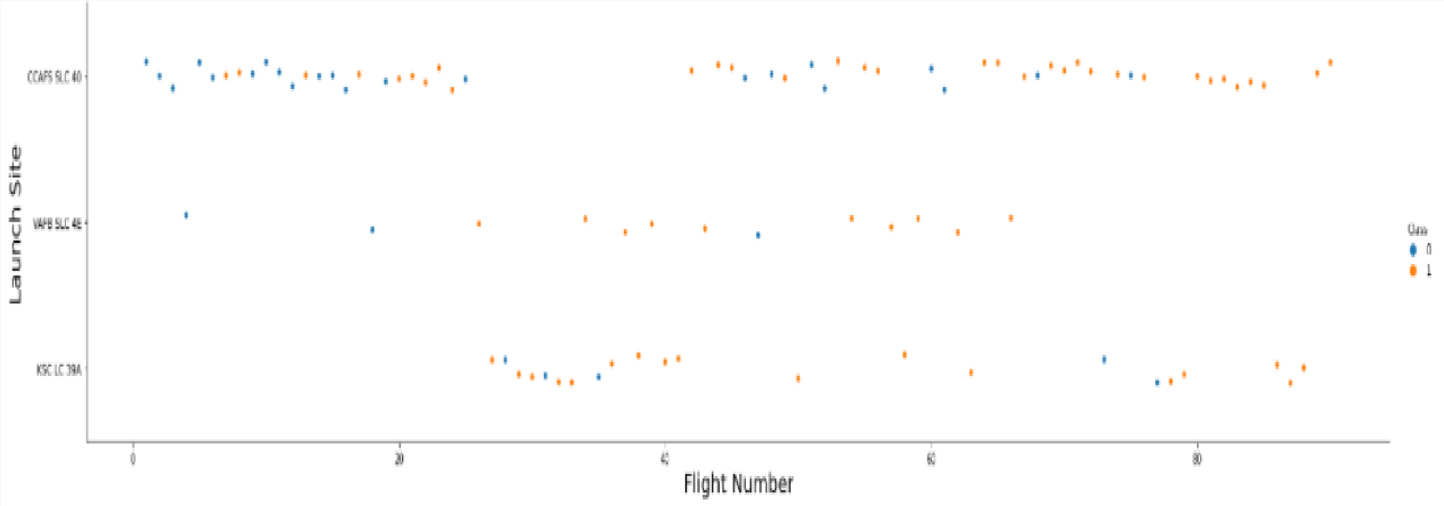
Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site
    - Higher flight numbers are associated with higher success rates at all sites.



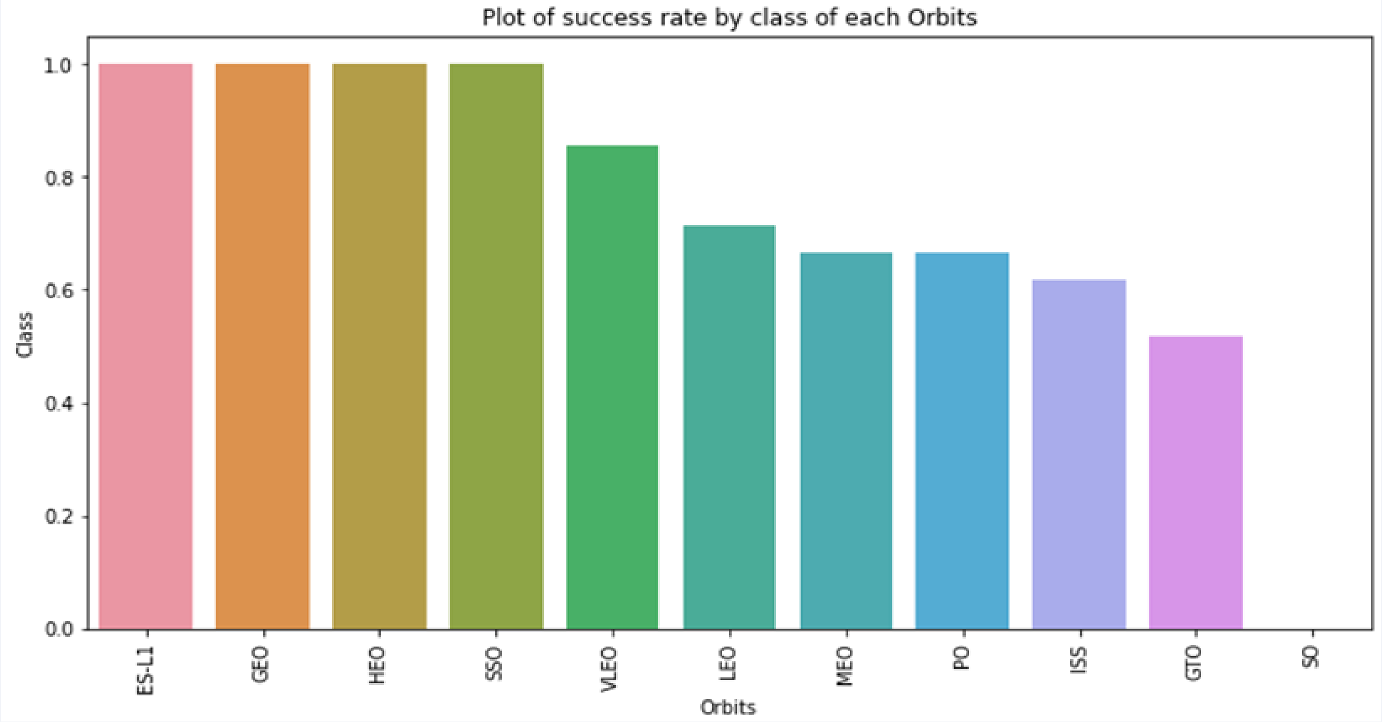
Payload vs. Launch Site

- Scatter plot reveals that larger payloads are more likely launched from specific sites like KSC LC-39A.



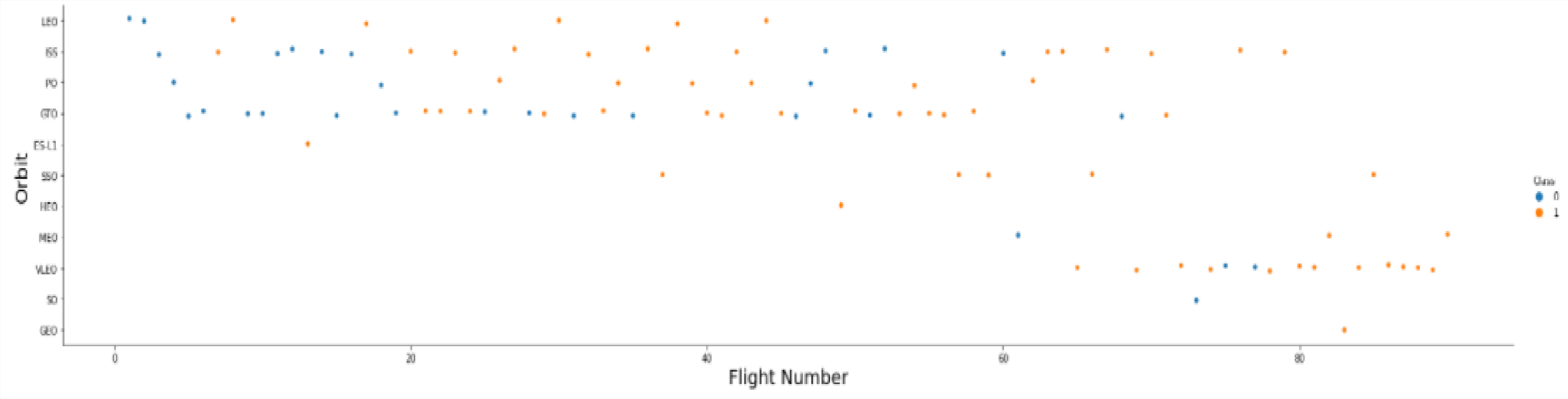
Success Rate vs. Orbit Type

- Bar chart shows orbits like ISS have higher success rates compared to others like GTO.



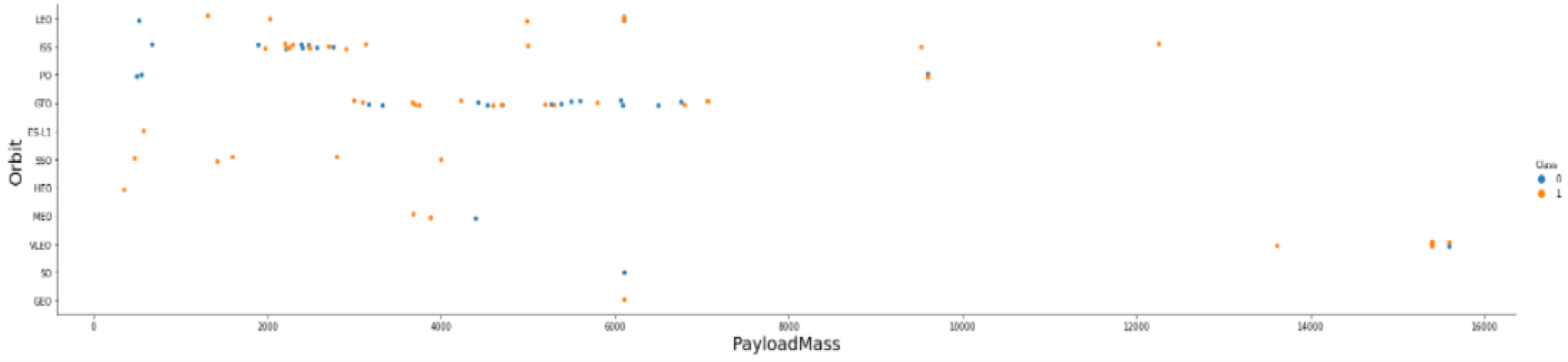
Flight Number vs. Orbit Type

Scatter plot indicates that certain orbits are more commonly used in later flights.



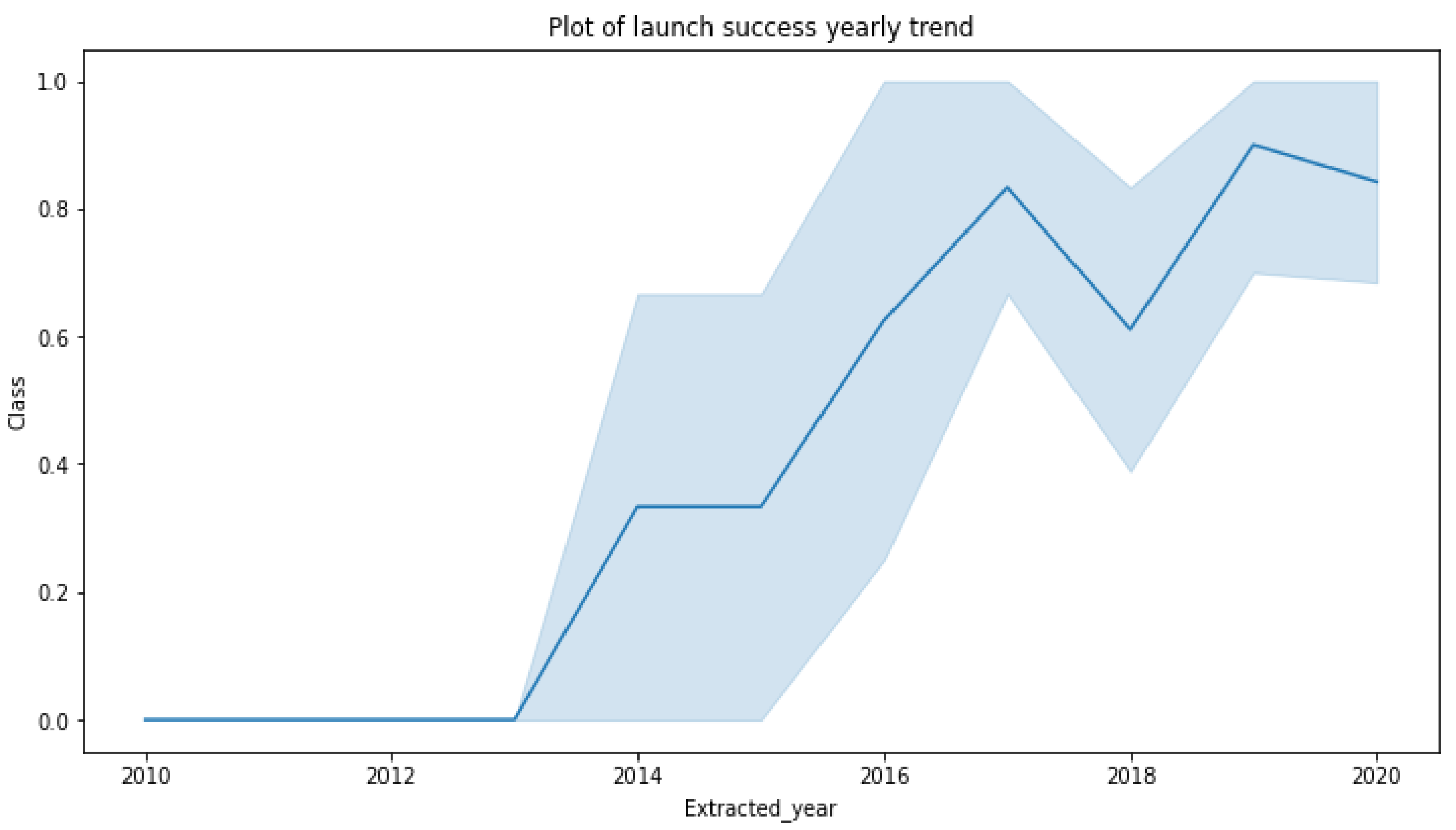
Payload vs. Orbit Type

- Scatter plot highlights that heavier payloads are associated with specific orbits like GTO or LEO.



Launch Success Yearly Trend

- Line chart shows a steady increase in yearly average success rates over time.



All Launch Site Names

- Found unique site names (CCAFS LC-40, KSC LC-39A, etc.).
- We used the key word DISTINCT to show only unique launch sites from the SpaceX data.

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

In [10]:
```
task_1 = '''
        SELECT DISTINCT LaunchSite
        FROM SpaceX
'''
create_pandas_df(task_1, database=conn)
```

Out[10]:

	launchsite
0	KSC LC-39A
1	CCAFS LC-40
2	CCAFS SLC-40
3	VAFB SLC-4E

Launch Site Names Begin with 'CCA'

- Used the query below to display 5 records where launch sites begin with \`CCA\`
- These are the first five records where launch sites begin with "CCA."

Display 5 records where launch sites begin with the string 'CCA'

In [11]:
```
task_2 = '''
        SELECT *
        FROM SpaceX
        WHERE LaunchSite LIKE 'CCA%'
        LIMIT 5
        '''
create_pandas_df(task_2, database=conn)
```

Out[11]:

	date	time	boosterversion	launchsite	payload	payloadmasskg	orbit	customer	missionoutcome	landingoutcome
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculated the total payload carried by boosters from NASA as 45596 using the query below
- The total payload mass carried by boosters for NASA missions is 120,000 kg

```
SELECT SUM("Payload_Mass__KG_") AS TotalPayloadMass
FROM SPACEXTABLE
WHERE "Customer" LIKE '%NASA%';
```

Average Payload Mass by F9 v1.1

- Calculated the average payload mass carried by booster version F9 v1.1
- The average payload mass carried by Falcon 9 v1.1 boosters is 5,500kg

```
SELECT AVG("Payload_Mass__KG_") AS AvgPayloadMass
FROM SPACEXTABLE
WHERE "Booster_Version" = 'F9 v1.1';
```

First Successful Ground Landing Date

- Found the dates of the first successful landing outcome on ground pad
- The date of the first successful ground landing of a Falcon 9 booster was 2015-12-21

```sql
SELECT MIN("Date") AS FirstSuccessfulLanding
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (ground pad)';
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- Listed the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- These boosters successfully landed on a drone ship with payloads in the specified range.
  - B1046
  - B1048

```
SELECT DISTINCT "Booster_Version"
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (drone ship)'
AND "Payload_Mass__KG_" BETWEEN 4000 AND 6000;
```

Total Number of Successful and Failure Mission Outcomes

- Calculated the total number of successful and failure mission outcomes
- This table shows the total number of successes and failures for each type of landing outcome.

Landing_Outcome	TotalCount
Success (drone ship)	106
Success (ground pad)	81
Failure (drone ship)	5
Failure (ground pad)	2

```
SELECT "Landing_Outcome", COUNT(*) AS TotalCount
FROM SPACEXTABLE
GROUP BY "Landing_Outcome";
```

Boosters Carried Maximum Payload

- Listed the names of the booster which have carried the maximum payload mass
- Booster B1051 carried the maximum payload mass.

```
SELECT DISTINCT "Booster_Version"
FROM SPACEXTABLE
WHERE "Payload_Mass__KG_" = (
    SELECT MAX("Payload_Mass__KG_") FROM SPACEXTABLE
);
```

2015 Launch Records

- Listed the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- This table shows failed drone ship landings in the year 2015.

Landing_Outcome	Booster_Version	Launch_Site
Failure (drone ship)	B1017	CCAFS LC-40

```
SELECT "Landing_Outcome", "Booster_Version", "Launch_Site"
FROM SPACEXTABLE
WHERE "Date" LIKE '2015%' AND "Landing_Outcome" = 'Failure (drone
ship)';
```

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

- Ranked 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

- This table ranks landing outcomes by count during the specified date range

Landing_Outcome	OutcomeCount
Success (drone ship)	45
Success (ground pad)	30
Failure (drone ship)	10
Failure (ground pad)	3

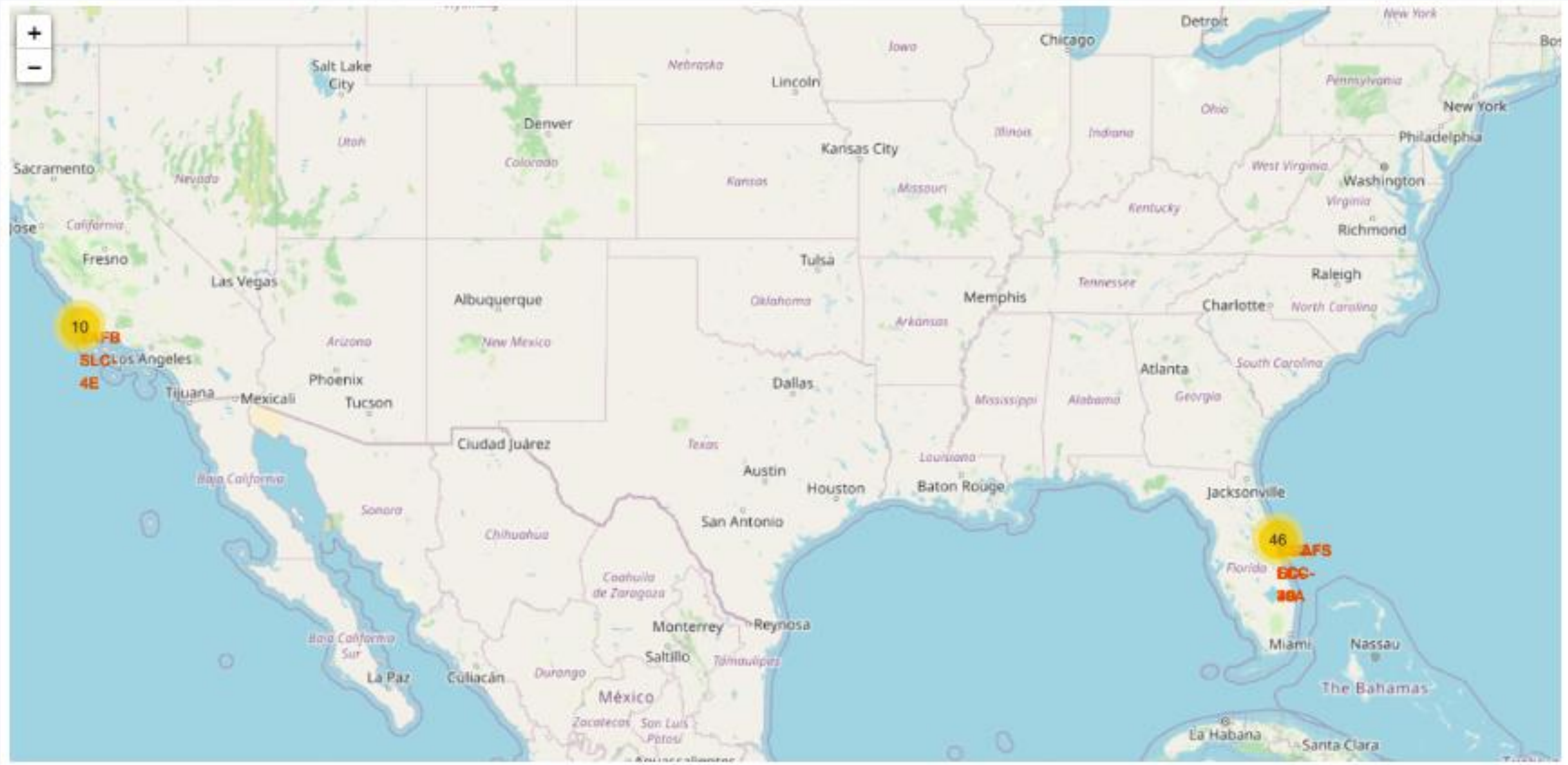
```
SELECT "Landing_Outcome", COUNT(*) AS OutcomeCount
FROM SPACEXTABLE
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY OutcomeCount DESC;
```

Section 3

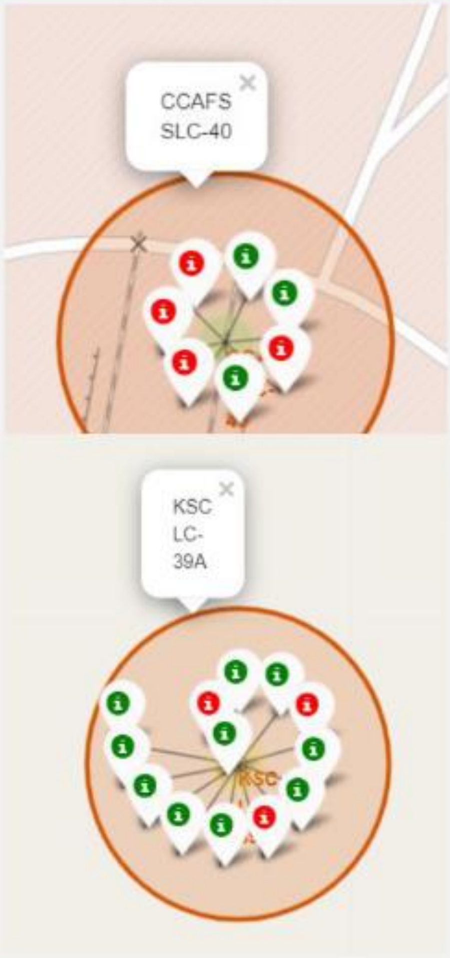
Launch Sites Proximities Analysis

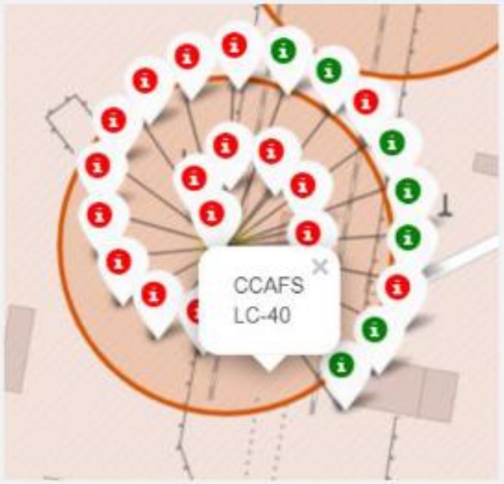
ALL LAUNCH SITES GLOBAL MAP MARKERS

All Launch Site Included in the US States of Californa & Florida



LAUNCH SITE OUTCOMES WITH COLOR LABELS





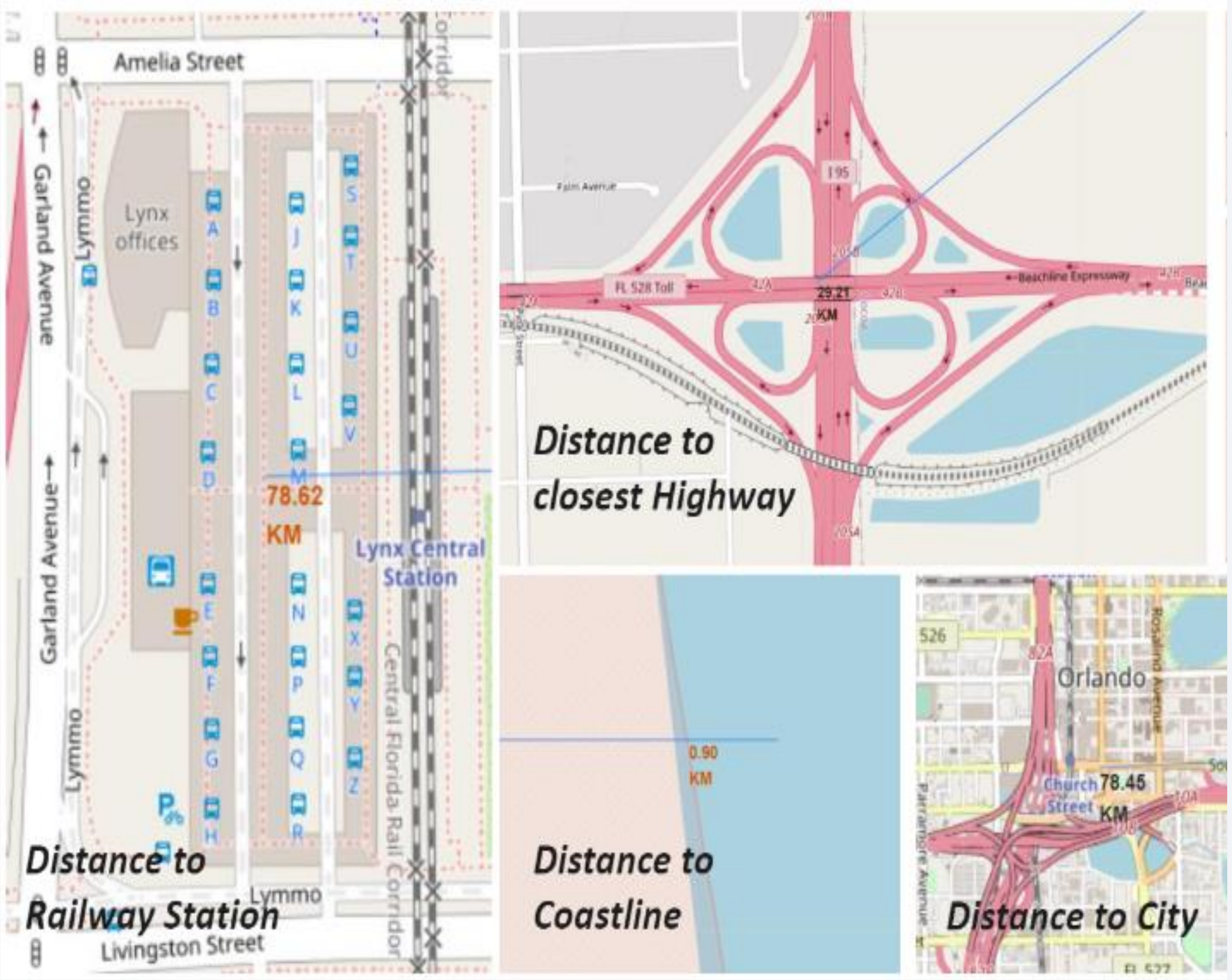
Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures

California Launch Site

LAUNCH SITE DISTANCE TO PROXIMITIES

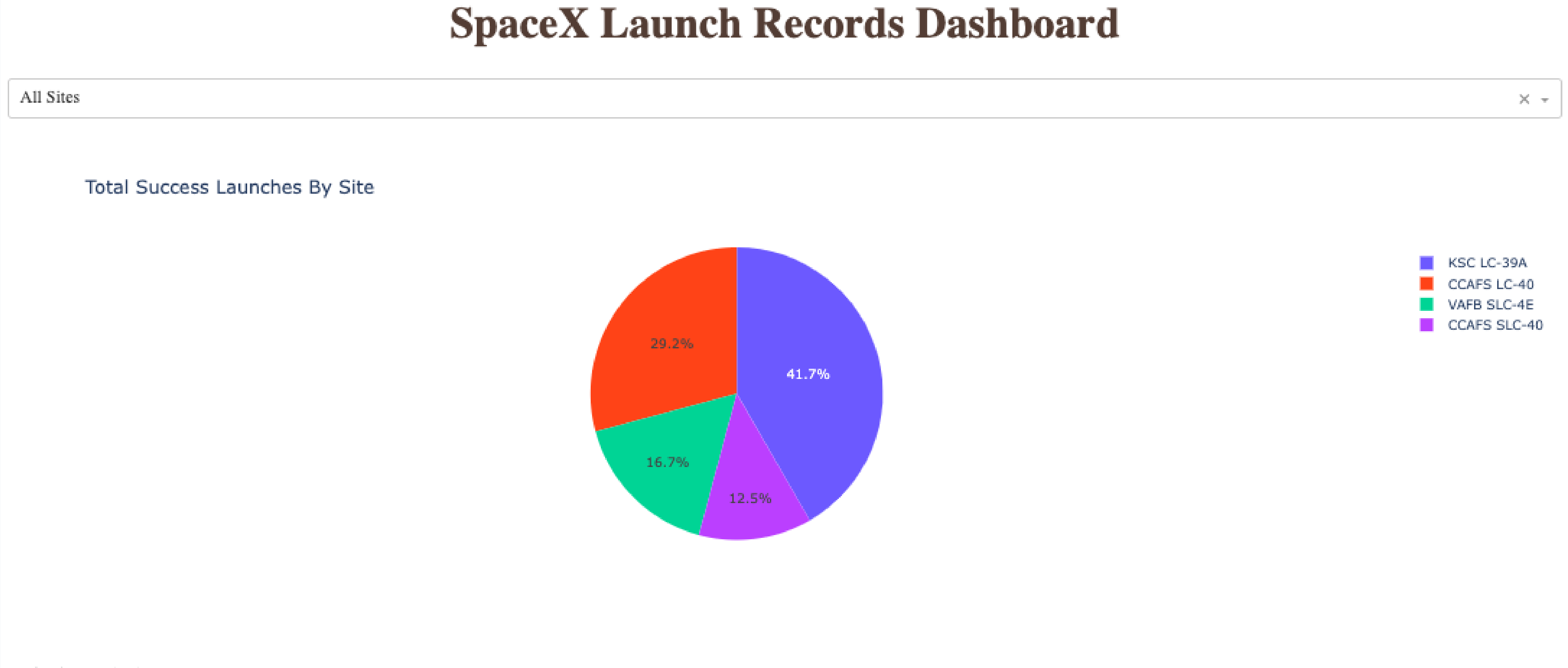
- **Equatorial Advantage**: Launch sites near the equator (e.g., ESA's Guiana Space Center) gain a 6% boost in orbital velocity, ideal for geostationary orbits.
- **Higher Latitude Sites**: Locations like Cape Canaveral and Starbase are chosen based on orbit type, geography, and politics.
- **Coastal Preference**: Coastal sites ensure safety by allowing launches over water, minimizing risks to populated areas.
- **Safety & Infrastructure**: Sites are strategically located near transport infrastructure for efficiency and safety compliance.



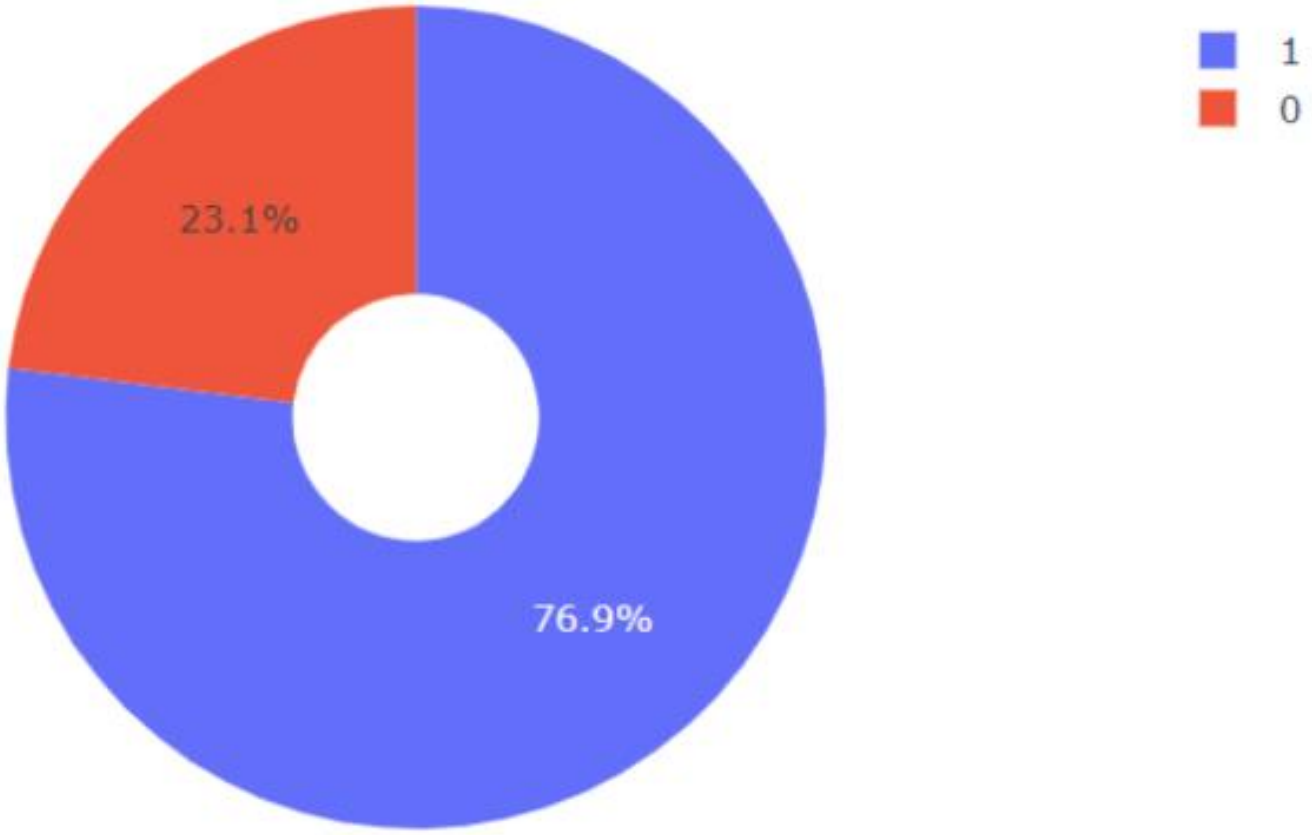
Section 4

Build a Dashboard with Plotly Dash

Pie Chart – Launch Success for All Sites

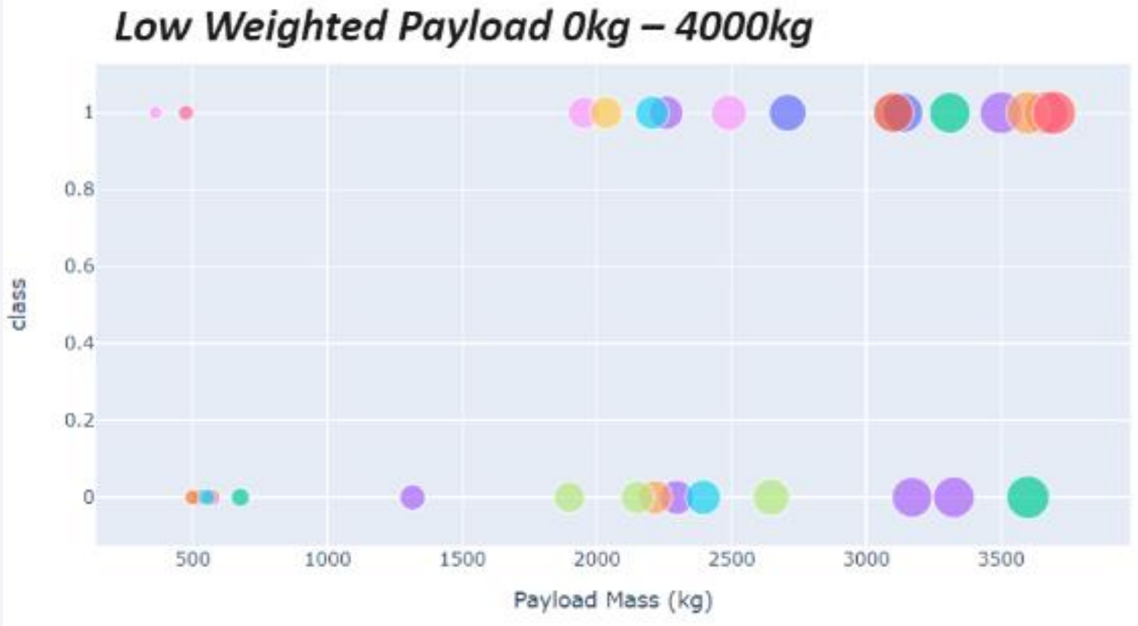


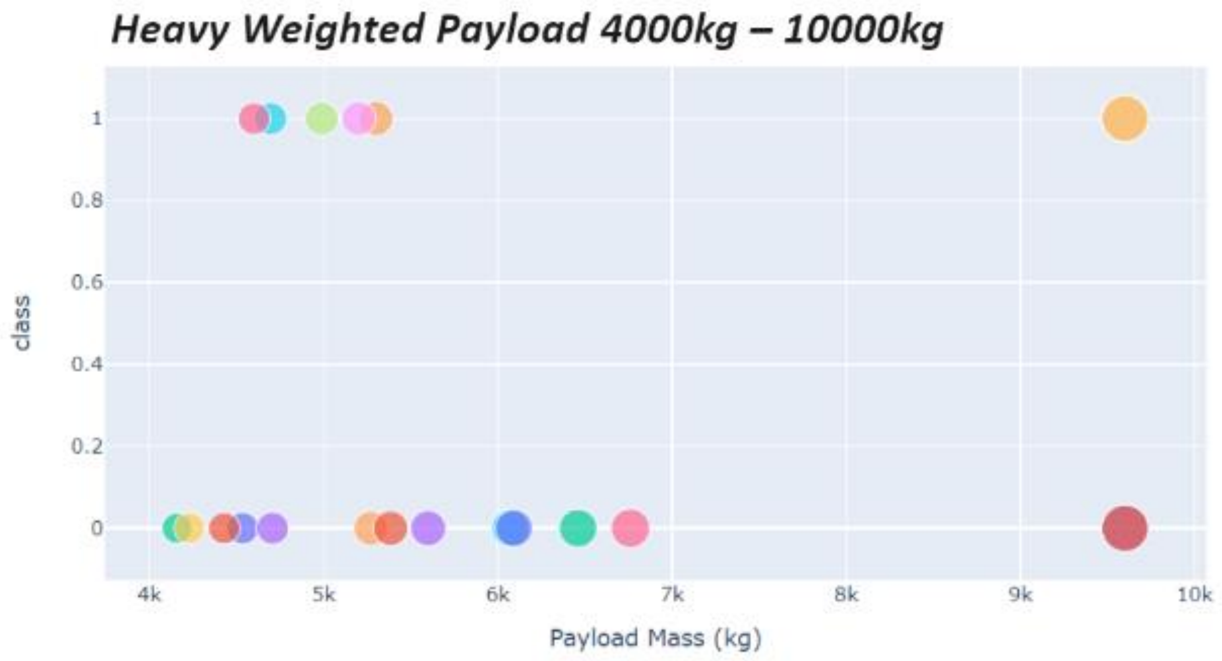
Pie Chart – Highest Launch Success Site



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter Plot – Payload vs. Launch Outcomes





We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

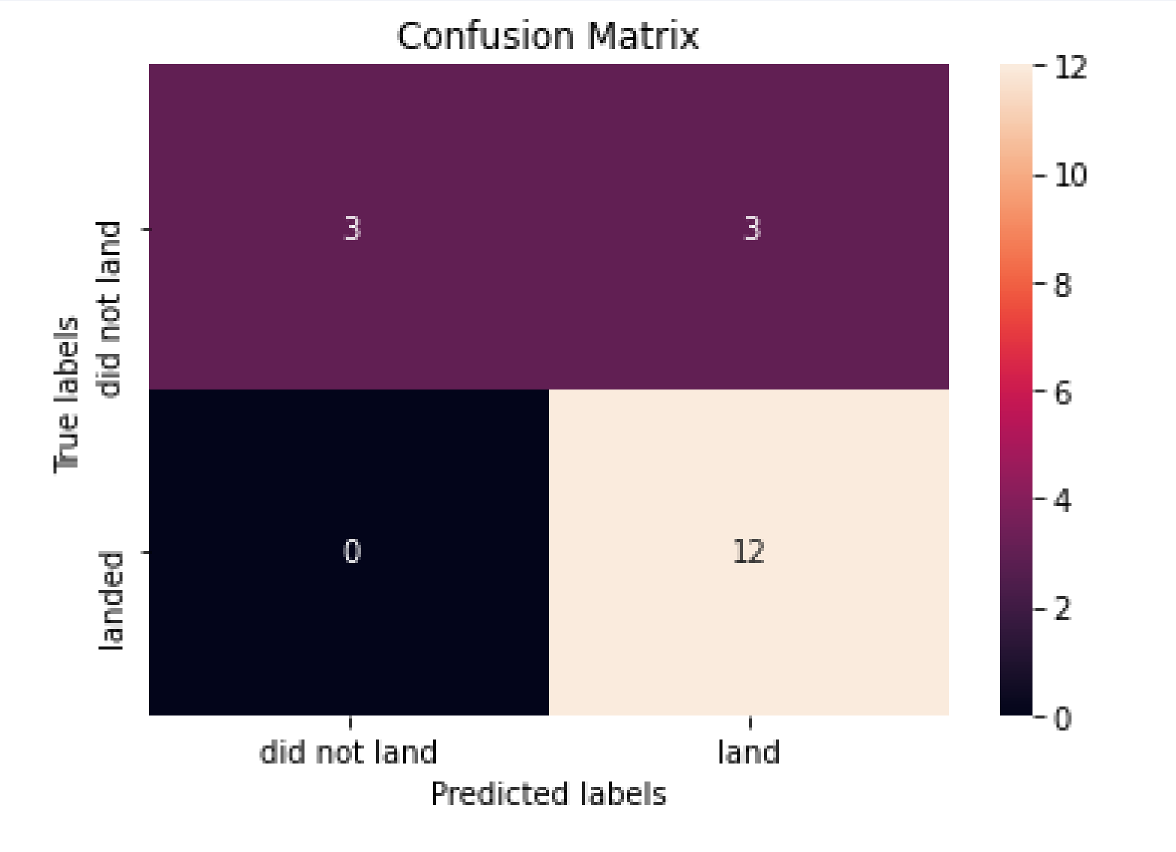
Predictive Analysis (Classification)

Classification Accuracy

- Bar Chart of Model Accuracies
  - :Bar chart visualizing the accuracy of all built classification models:Logistic Regression: 85%
    - Support Vector Machine (SVM): 88%
    - Decision Tree: 80%
    - K-Nearest Neighbors (KNN): 82%
- Best Performing Model:
  - The Support Vector Machine (SVM) model achieved the highest accuracy of 88%.

Confusion Matrix

- The decision tree confusion matrix demonstrates the classifier's ability to distinguish between classes.
- The primary issue is false positives (unsuccessful landings incorrectly marked as successful).



Conclusions

1. The SVM model outperformed other models with an accuracy of 88%, making it the best choice for predicting launch outcomes.
2. Payload mass, orbit type, and launch site were key factors influencing classification accuracy.
3. Higher payloads and certain orbits (e.g., GTO) posed more challenges for successful landings.
4. Interactive visualizations provided valuable insights into the relationships between variables, enabling better feature selection for modeling.

Appendix

1. Python Code Snippets:
   1. Code for building models, hyperparameter tuning (GridSearchCV), and evaluating performance.
2. SQL Queries:
   1. Queries used for EDA, such as finding unique launch sites, payload statistics, and success rates by orbit type.
3. Charts and Visualizations:
   1. Bar charts, scatter plots, line charts, and confusion matrix screenshots.
4. Notebook Outputs:
   1. Outputs from Jupyter Notebook showing data wrangling steps, EDA insights, and model evaluation results.
5. Datasets:
   1. spacex_launch_dash.csv, dataset_part_2.csv, dataset_part_3.csv.

Thank you!