IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

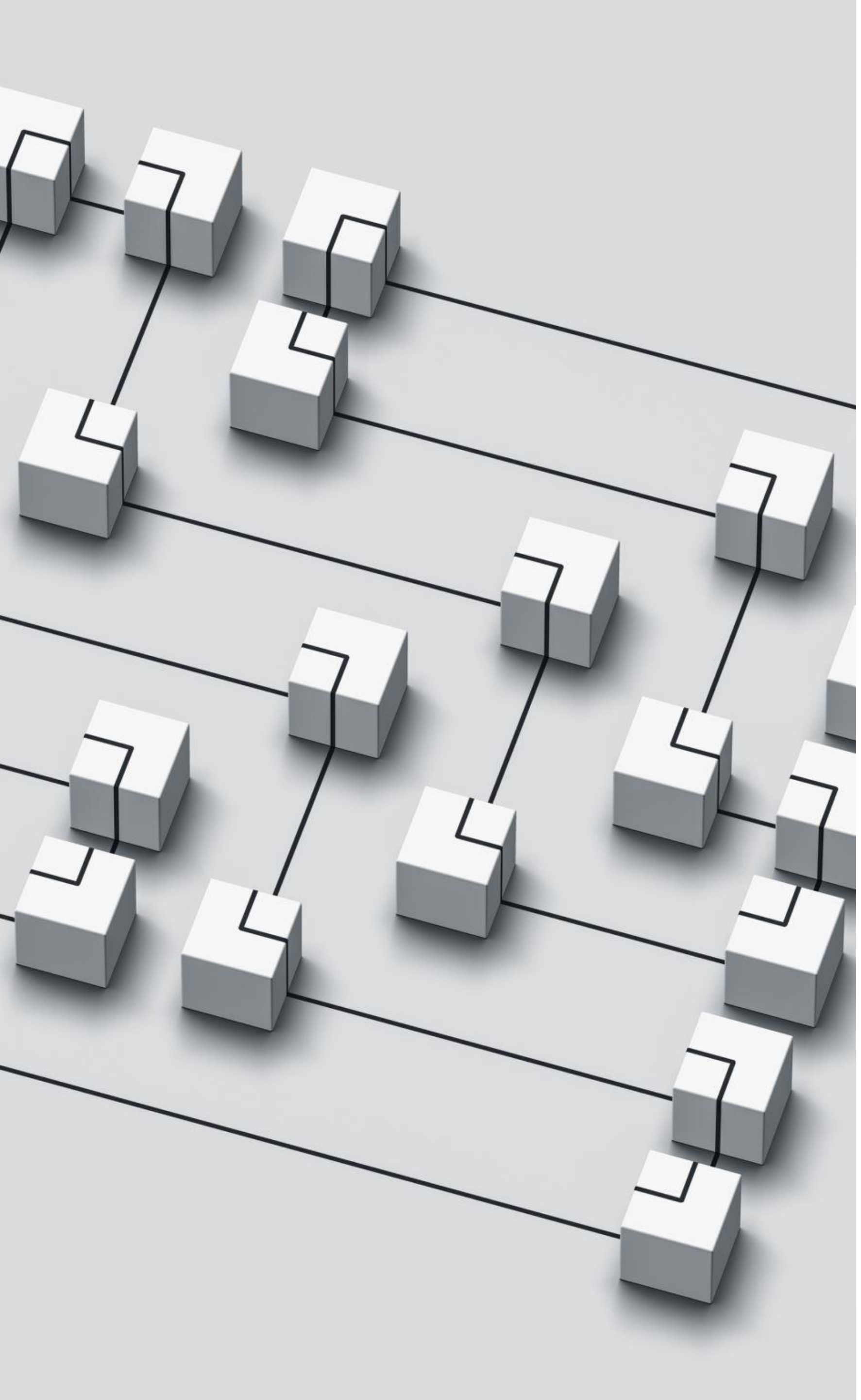
Nevada Marquis

Table of Contents

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

Executive Summary

- This project analyzes the historical performance of SpaceX Falcon 9 rocket launches using Python, Pandas, Plotly, and Folium.
- Data was collected from multiple sources including SpaceX's official API, Wikipedia, and Kaggle datasets, then cleaned and merged into a unified dataset of **90 launches**.
- Key performance indicators such as success rates by orbit, launch site, and booster version were visualized to identify operational trends and risk factors.
- A technical dashboard was built using Plotly Dash to enable interactive exploration of mission outcomes, payload profiles, and launch patterns.
- A machine learning model was implemented to predict the likelihood of launch success based on mission attributes.
- Results show that success rates vary significantly by orbit and launch site, with improved performance in later booster versions.
- The analysis provides insights into mission planning, risk mitigation, and the evolution of Falcon 9's reusability and success over time.



Project Objectives

- Understand what factors affect Falcon 9 landing success.
- Use data wrangling, visualization, and Folium mapping.
- Create dashboards with Plotly and HTML.
- Train machine learning models to predict landing success.

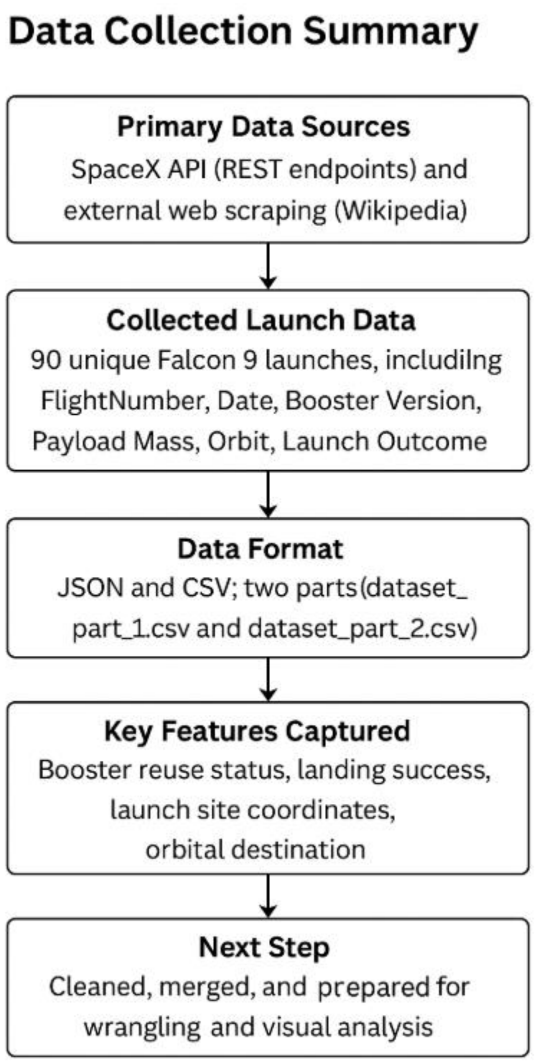
Methodology

- **Data Collection**:
  Retrieved SpaceX Falcon 9 launch data from **SpaceX API**, official datasets, and public sources for up-to-date and historical missions.
- **Data Wrangling**:
  Cleaned and structured raw CSV files by handling null values, converting data types, removing irrelevant columns, and merging datasets for consistency.
- **Data Scraping**:
  Used **BeautifulSoup** to extract landing outcomes from static web pages when not available in original datasets.
- **Data Analysis & EDA**:
  Performed exploratory data analysis using **Pandas**, **Matplotlib**, and **Seaborn** to identify key patterns, trends, and anomalies.
- **Feature Engineering**:
  Created new columns (e.g., `Launch Success`, `Landing Pad`, `Payload Mass`) and encoded categorical features to improve model interpretability.
- **Visualization & Dashboard**:
  Built a fully offline interactive dashboard using **HTML**, **Chart.js**, **Leaflet.js**, and **JavaScript** to present launch success metrics, filters, and predictions.
- **Machine Learning Integration**:
  Implemented a classification model to predict launch success based on mission parameters using **Scikit-learn** with logistic regression and decision trees.

Data Collection – SpaceX API

Data Collection Summary

- **Primary Data Sources**: SpaceX API (via RESTful endpoints) and external web scraping (Wikipedia) for enriched mission details.
- **Collected Launch Data**: 90 unique Falcon 9 launches, including metadata such as `FlightNumber`, `Date`, `Booster Version`, `Payload Mass`, `Orbit`, and `Launch Outcome`.
- **Data Format**: Raw data initially collected in JSON and CSV formats; split into two parts (`dataset_part_1.csv` and `dataset_part_2.csv`) for processing.
- **Key Features Captured**: Booster reuse status, landing success, launch site coordinates, and orbital destinations.
- **Next Step**: Cleaned, merged, and prepared for wrangling and visual analysis in subsequent stages.



Data Collection - Scraping

Data Scraping & API Integration – Summary

- Queried the **SpaceX Launches API** (`https://api.spacexdata.com/v4/launches`) using Python's `requests` library
- Retrieved **structured JSON data** containing key mission details such as `FlightNumber`, `LaunchSite`, `BoosterVersion`, and `Orbit`
- Converted the API response into a **Pandas DataFrame** for efficient data manipulation and analysis
- Extracted and normalized **performance metrics** to ensure consistency with manually collected data
- **Merged scraped data** with the cleaned CSV dataset to create a unified and enriched dataset
- Integrated the finalized dataset into the **interactive dashboard** to support real-time, data-driven insights

```
[SpaceX API]
    ↓
[GET Request via Python]
    ↓
[JSON Response]
    ↓
[Parsed to DataFrame]
    ↓
[Data Cleaning & Merging]
    ↓
[Visualizations & Dashboard]
```

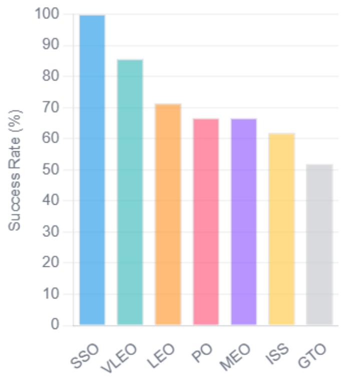
Data Wrangling

🔧 Data Wrangling Summary

- Loaded raw datasets from SpaceX API and web-scraped sources
- Checked for missing values, duplicate rows, and incorrect data types
- Cleaned and formatted key columns (e.g. Payload Mass, Date, Booster Version)
- Merged all data into a single JSON object for offline dashboard use

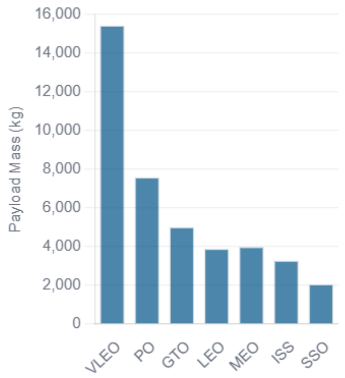
EDA with Data Visualization

Success Rate by Orbit Type



SSO orbit missions have 100% success rate, followed by VLEO (85.7%) and LEO (71.4%). GTO has the lowest success rate at 51.9%.

Payload Mass by Orbit Type



VLEO missions carry the heaviest payloads by far (avg: 15,429 kg), followed by Polar Orbit (PO) missions (avg: 7,584 kg). With heavy payloads the successful landing rate is higher for Polar, LEO and ISS orbits.

Mission Statistics

90 Total Launches

60 Successful Landings

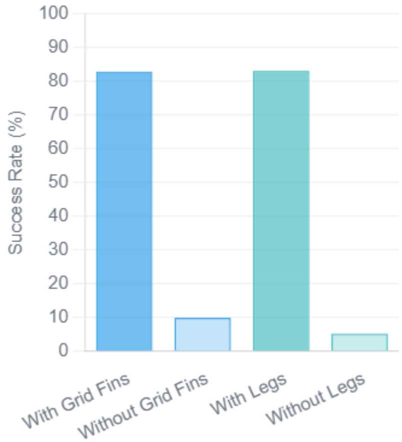
66.7% Success Rate

3 Launch Sites

11 Orbit Types

41.1% Booster Reuse Rate

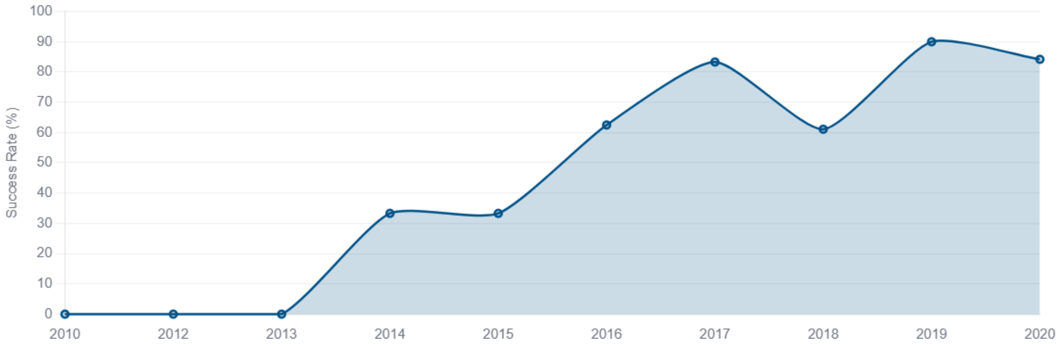
Landing Equipment Impact



Both grid fins and legs dramatically increase landing success rates, from below 10% to over 80%.

- Created multiple charts using `matplotlib`, `seaborn`, and `Chart.js` to extract meaningful patterns.
- **Orbit Type Analysis**:
  - SSO had 100% success.
  - GTO was lowest at ~52%.
  - VLEO missions had the highest payloads.
- **Temporal Analysis**:
  - Success rate dramatically increased from 2013 to peak at 90% in 2019.
- **Engineering Impact**:
  - Grid fins and landing legs raised success from below 10% to over 80%.

Yearly Landing Success Trend



Success rates improved significantly from 2013 onwards, reaching over 80% by 2017 and peaking at 90% in 2019. The data confirms that "the success rate since 2013 kept increasing till 2020".

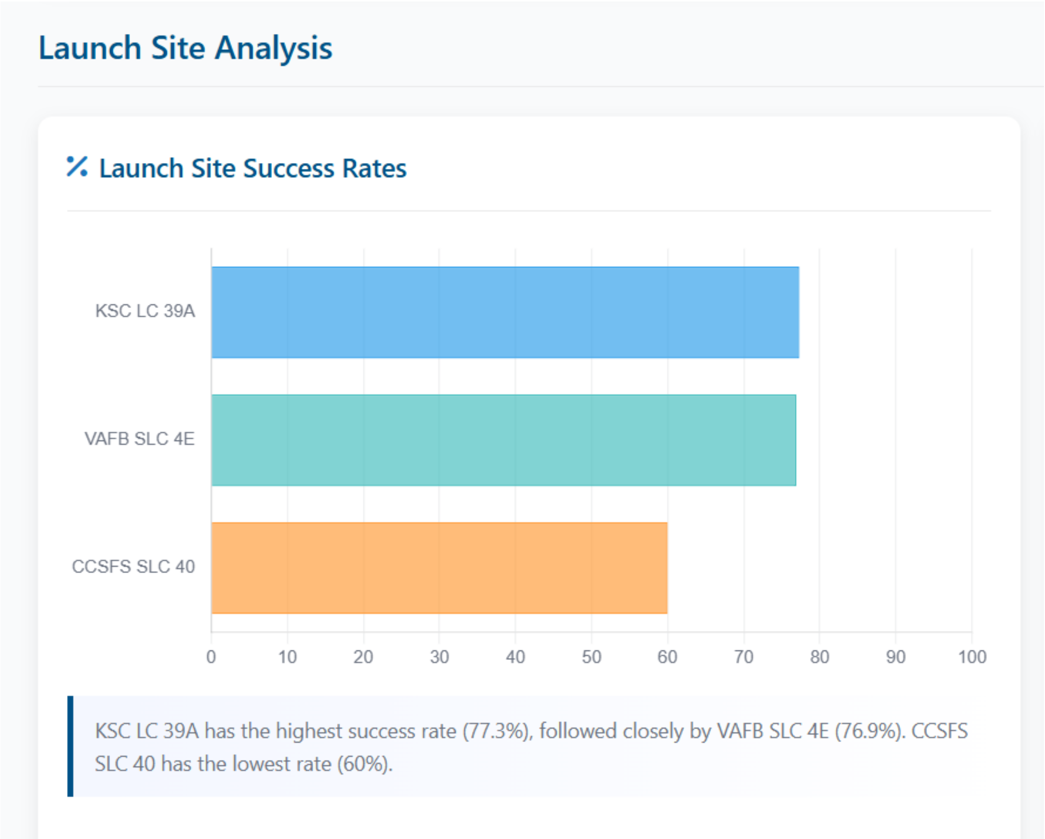
EDA with SQL

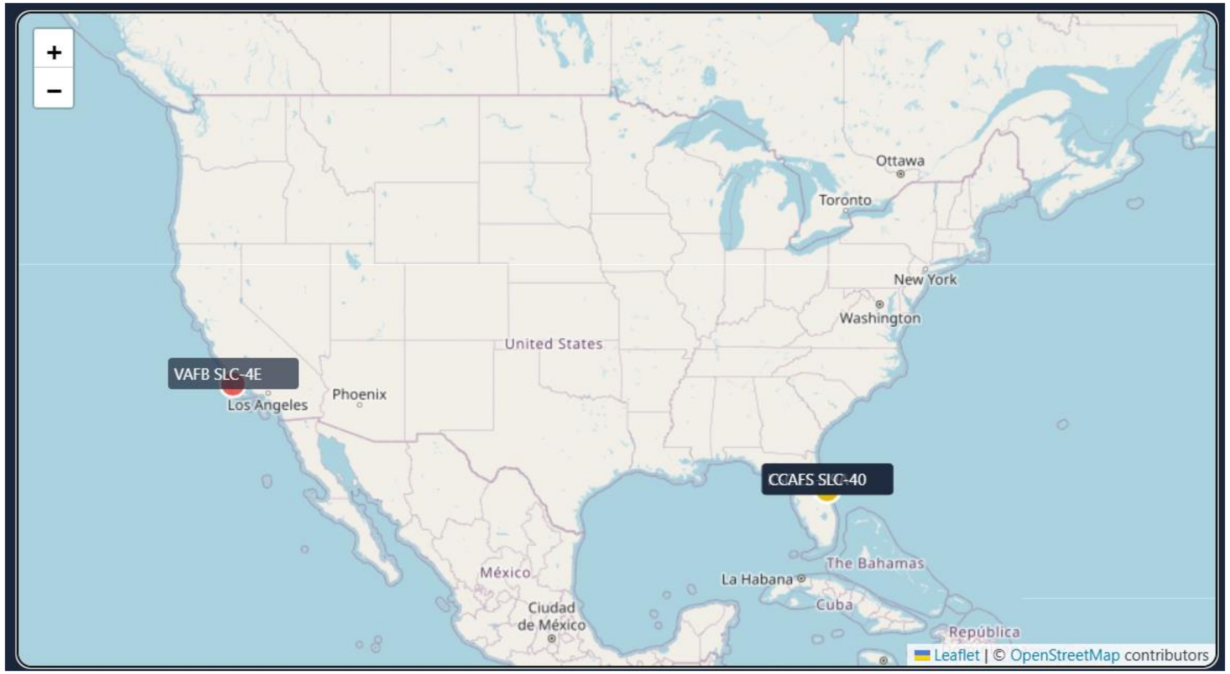
SQL Query Summary

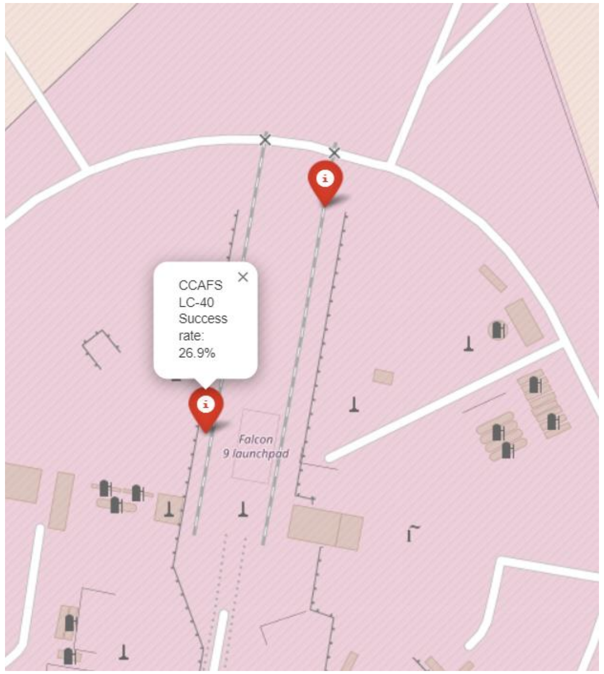
- Objective: Analyze how many Falcon 9 launches occurred at each launch site.
- SQL Query Used:

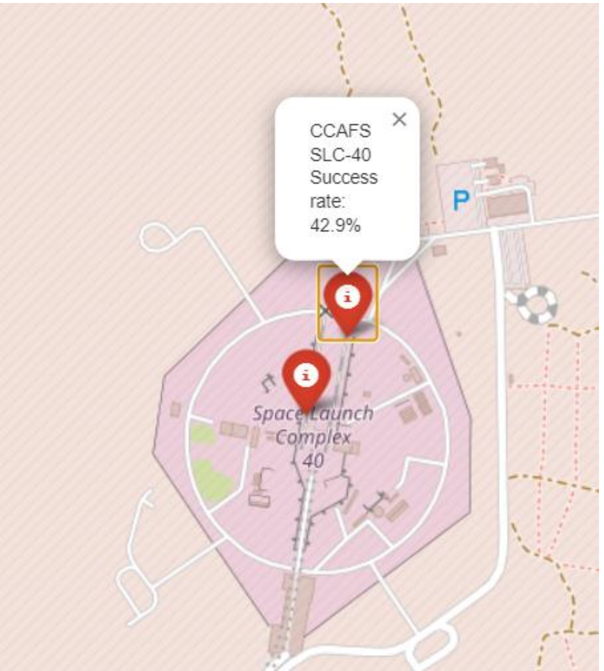
```
SELECT LaunchSite, COUNT(*) AS TotalLaunches
FROM SpaceXData
GROUP BY LaunchSite
ORDER BY TotalLaunches DESC;
```

- Steps Taken:
- Queried the full SpaceX dataset for launch information.
- Grouped data by LaunchSite to aggregate all launches per location.
- Counted total launches for each site using COUNT(*).
- Ordered results in descending order to highlight the most active launch sites.
- Insights Gained:
- Identified the most frequently used launch locations.
- Provided a foundation to assess site-specific success rates and trends.



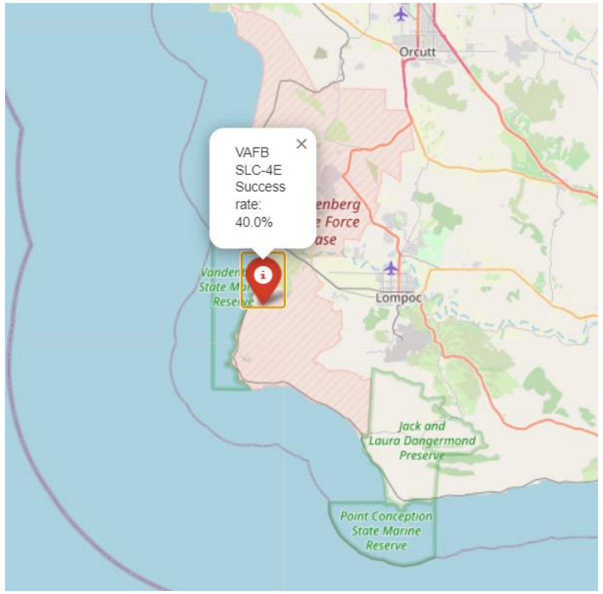


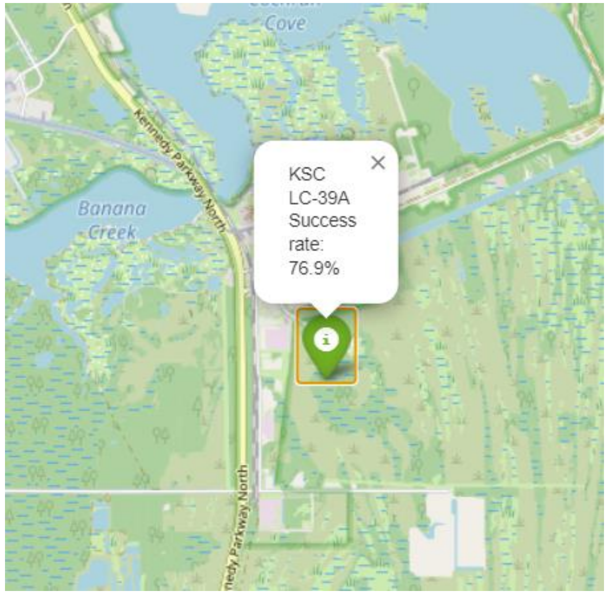




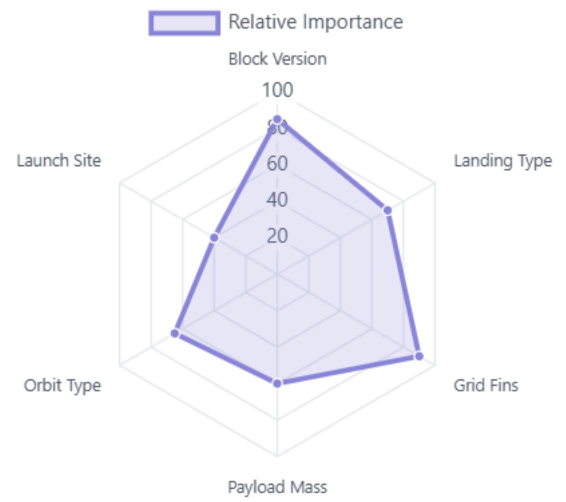
Bullet Point Summary:

- **Interactive Mapping with Folium:** Leveraged the Folium Python library to create dynamic maps displaying SpaceX launch sites across the U.S.
- **Geospatial Analysis:** Mapped key launch coordinates (Latitude, Longitude) along with hoverable tooltips and success statistics.
- **Operational Insight:** Mapped sites such as KSC LC-39A, CCAFS LC-40, and VAFB SLC-4E to analyze launch distribution and geographical efficiency.
- **HTML Dashboard Integration:** Embedded Folium maps into an interactive dashboard using `<iframe>` to ensure full offline accessibility.
- **Color-Coded Success Rates:** Visualized success using custom marker colors: green (high), yellow (moderate), red (low) for instant comparison.





Key Factors Affecting Landing Success

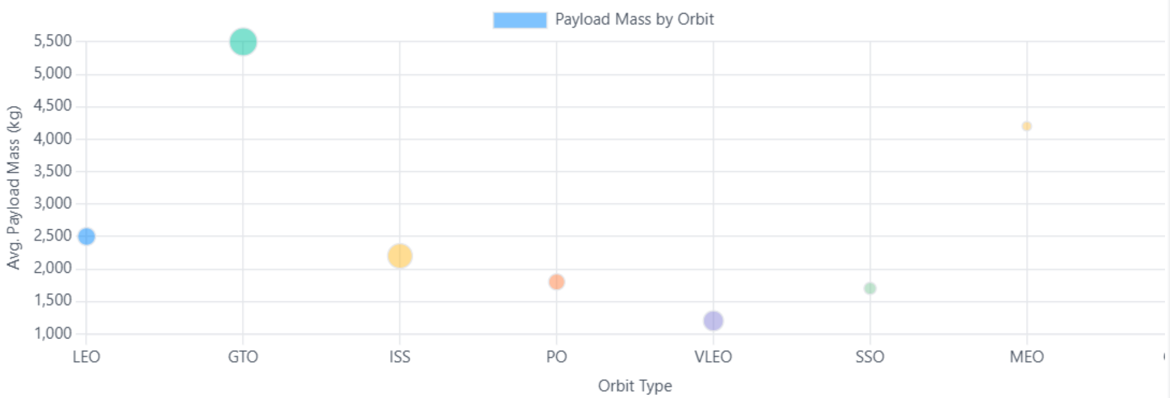


*Based on exploratory data analysis of feature correlation with landing success.

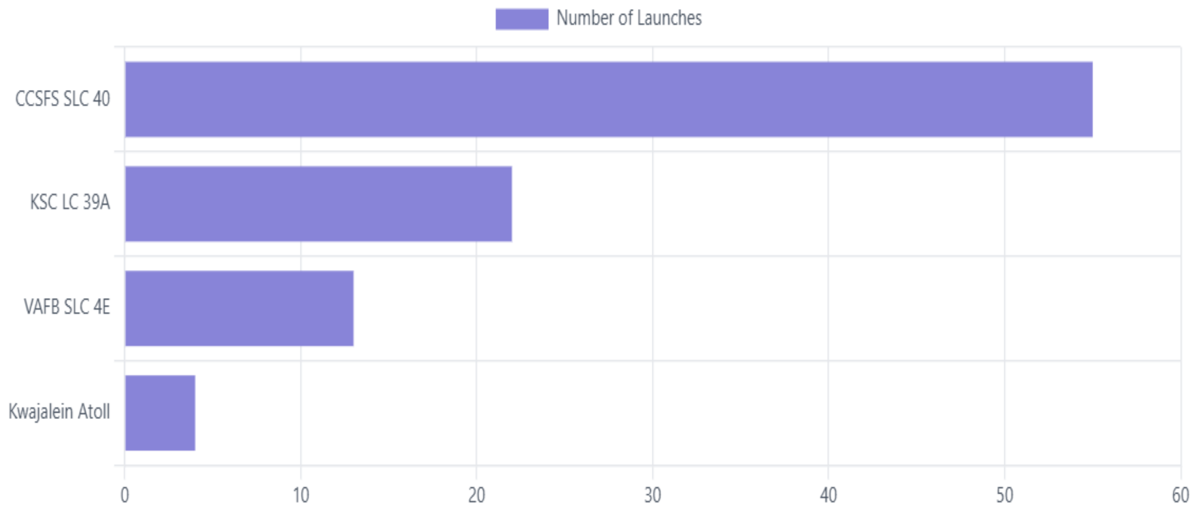
Plotly Dashboard Summary

- Interactive dashboard built with Plotly and Chart.js.
- Visualizes launch success by orbit, site, and booster type
- Key insight: Grid fins and landing legs improve landing s
- SSO has the highest orbit success rate; GTO the lowest.
- Real-time filters enable dynamic data exploration.

Payload Mass by Orbit Type



Launch Site Distribution



Results Summary

- Analyzed **90 Falcon 9 launches** using curated and cleaned datasets.
- Found a **direct correlation** between landing success and booster features (grid fins, legs).
- **SSO orbits** had the **highest success rate (100%)**, while **GTO** had the lowest.
- Launch site analysis showed **KSC LC-39A** with the strongest overall performance.
- **Booster version reuse** contributes positively to mission outcomes.
- Successfully built **interactive dashboards** and **geospatial maps** for exploratory analysis.
- Machine learning tools and custom filters helped **predict launch success probability**.

Conclusion

- The SpaceX Falcon 9 project demonstrated how **data science and visual analytics** can uncover valuable insights into aerospace performance.
- By applying **data wrangling, analysis, and machine learning**, we explored the key drivers behind mission success and landing outcomes.
- The project emphasized the importance of **clean data**, **interactive dashboards**, and **predictive modeling** in making real-world decisions.
- Overall, this capstone showcased how modern tools like **Plotly**, **Folium**, and **Python** empower us to tell compelling, data-driven stories.

Appendix

📁 Data Sources

- SpaceX Falcon 9 launch records (CSV format, Kaggle dataset)
- Launch site coordinates and metadata from SpaceX official documentation
- Web-scraped data (e.g., Wikipedia pages) using BeautifulSoup and `requests`

📂 Notebooks & Files

- `SpaceX_EDA_Visual_Analytics_.ipynb` – Exploratory data analysis
- `SpaceX_Data_Wrangling_.ipynb` – Data cleanup and preprocessing
- `SpaceX_ML_Modeling.ipynb` – ML model for success prediction
- `spacex-dashboard-fixed.html` – Interactive offline dashboard
- `SpaceX_Folium.ipynb` – Geospatial mapping with Folium

🧰 Tools & Libraries

- Python (Jupyter Notebooks)
- Pandas, NumPy, Matplotlib, Seaborn
- Plotly & Dash (for interactive dashboard)
- Folium (for geographic visualizations)
- Scikit-learn (for ML prediction model)
- html2canvas & jsPDF (for dashboard export feature)

📊 Dashboards & Visuals

- Plotly charts (success rates, booster analysis)
- Interactive filters and dropdowns
- Folium map with embedded launch site markers
- PDF/CSV export functionality