Winning Space Race with Data Science

Millicent Goodwin
IBM Data Science Capstone Project

Outline

- Introduction
- Data Collection
- Exploratory Data Analysis (EDA)
- Geospatial Analysis with Folium
- Machine Learning
- SQL Analysis
- Conclusion

Introduction

SpaceX has revolutionized the space industry with its Falcon 9 rocket, which costs significantly less to launch compared to other providers due to its reusability. As part of this project, we aim to predict whether the Falcon 9 first stage will successfully land using machine learning models, utilizing both historical data and geospatial analysis. By predicting landing success, SpaceX could optimize launch costs, benefiting both the company and potential competitors.

This report covers the following steps:

- **Data Collection**: How data was gathered from APIs and web scraping.
- **Exploratory Data Analysis (EDA)**: Analyzing the data for insights.
- **Feature Engineering**: Preparing the data for machine learning.
- **Geospatial Analysis**: Mapping and analyzing launch site locations.
- **Machine Learning**: Developing models to predict landing success.
- **SQL Analysis**: SQL queries to gain insights from the database.

Data Collection

The dataset used in this project was gathered in two main ways:

1. **SpaceX API**: The SpaceX API provides detailed information about past rocket launches. The data includes crucial features such as payload mass, booster versions, landing success, and more. API calls were made to retrieve launch-related data, and auxiliary functions were defined to collect additional information about rocket boosters, launch sites, payloads, and landing cores.
2. **Web Scraping**: The second was gathered by scraping historical Falcon 9 launch records from the Wikipedia page. Using BeautifulSoup, relevant columns such as flight numbers, payload masses, and landing outcomes were extracted into a clean dataset.

Exploratory Data Analysis (EDA)

- **Missing Data**: Various columns contained missing values, especially in landing outcomes and payload masses. The missing values were addressed by replacing them with mean values or handling them through imputation.
- **Launch Site Success**: The success rate of landing varied by launch site. This information is crucial for understanding where SpaceX performs its most successful landings.
- **Payload Mass**: Payload mass showed a strong relationship with landing success, with larger payloads more likely to result in unsuccessful landings.

Data Distribution

- **Landing Outcomes**: A classification task was defined where the landing outcome was either "successful" (1) or "unsuccessful" (0). This binary classification formed the target variable for machine learning models.

- **Orbit Types**: Different orbit types such as LEO (Low Earth Orbit), GEO (Geostationary Orbit), and SSO (Sun-Synchronous Orbit) were observed, with their respective success rates analyzed.

Feature Engineering

Feature engineering was performed to create meaningful features for the predictive model. Key steps included:

- **Converting categorical variables**: Features such as orbit type, booster version, and launch site were transformed into numerical values using encoding methods.
- **Handling Time Variables:** Date-related features were processed, with the launch date being converted into a datetime format. The time of day was extracted to analyze the correlation between landing success and time of launch.
- **Target Variable**: A binary variable landing_class was created, indicating whether the Falcon 9 booster successfully landed or not based on the outcome.

Geospatial Analysis with Folium

Using the **Folium** package, the geographic locations of launch sites were visualized on an interactive map. This analysis helped identify potential geographical patterns influencing launch success:

Task 1: Marking all launch sites on the map showed the proximity of different launch sites to coastal areas or major cities.

Task 2: Success and failure rates for each launch site were plotted, which helped identify whether proximity to certain locations affected the likelihood of a successful landing.

Task 3: The distances between launch sites and other major facilities were calculated, providing insights into logistical factors influencing the success rate.

Machine Learning

A series of machine learning models were tested to predict landing success, including:

Logistic Regression

K-Nearest Neighbors (KNN)

Random Forest

Support Vector Machines (SVM)

- The models were trained using features such as payload mass, launch site, and booster version. Performance metrics like accuracy, precision, recall, and the F1-score were used to evaluate the models.
- **Confusion Matrix**: The confusion matrix was plotted to assess the performance of the model. True positives and false negatives were crucial to understanding how often the model correctly predicted landing success versus failure.

After evaluating the models, **Random Forest** performed best in predicting landing success, with a high F1-score and balanced performance across both classes (successful and unsuccessful landings).

SQL Analysis

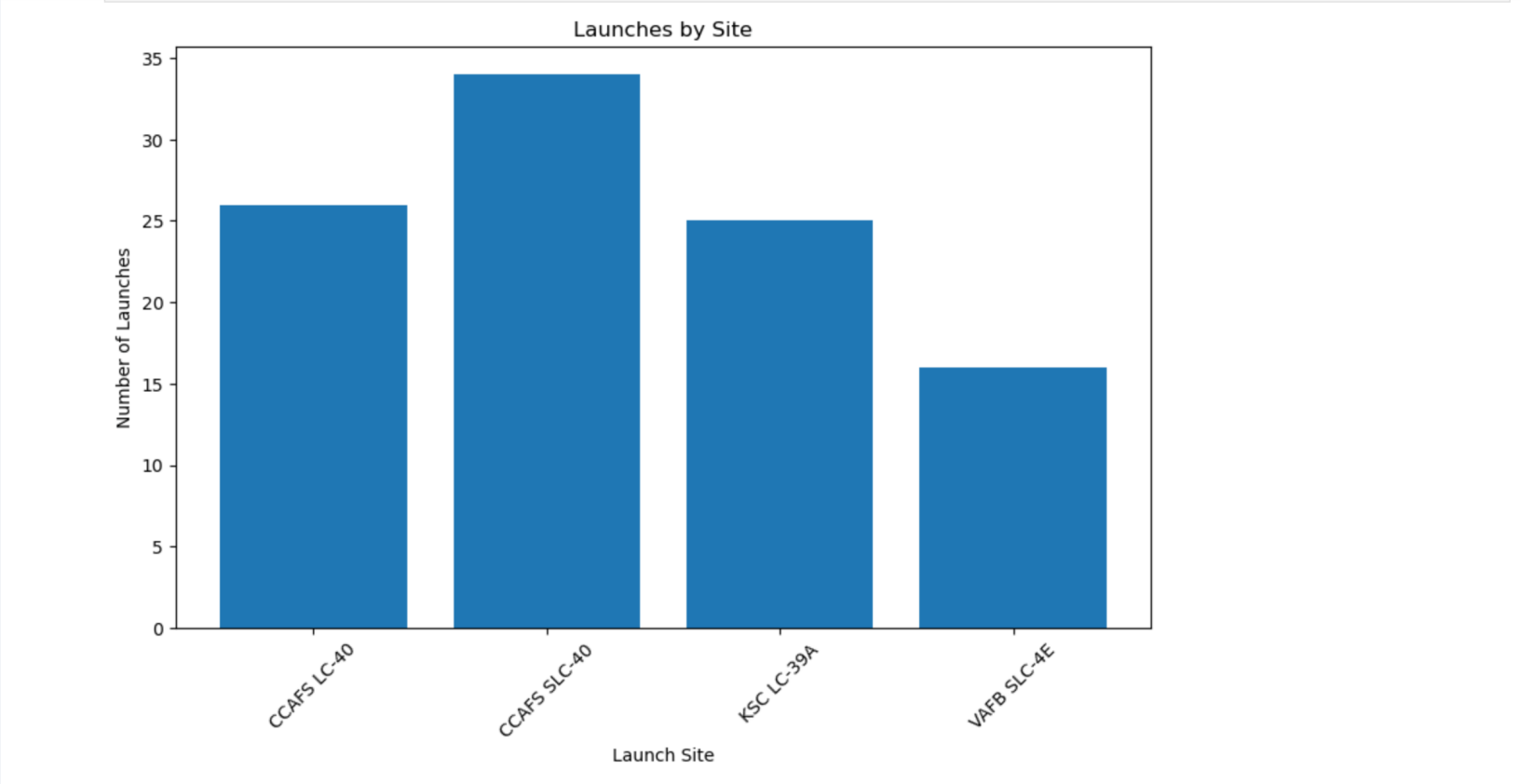
SQL queries were executed to explore the dataset stored in an SQLite database. The database was set up with the cleaned SpaceX dataset, and various queries were used to answer specific questions:

- **Unique Launch Sites**: The number of unique launch sites was identified.
- **Launches Starting with 'CCA'**: Launch sites starting with "CCA" were queried to assess any regional effects on success.
- **Payload Mass for NASA Launches**: A query was run to sum the payload mass for NASA missions.
- **Booster Version Performance**: The average payload mass for each booster version was calculated.
- **Launch Success and Failure Counts**: The number of successful and failed landings was queried to understand the distribution of outcomes.

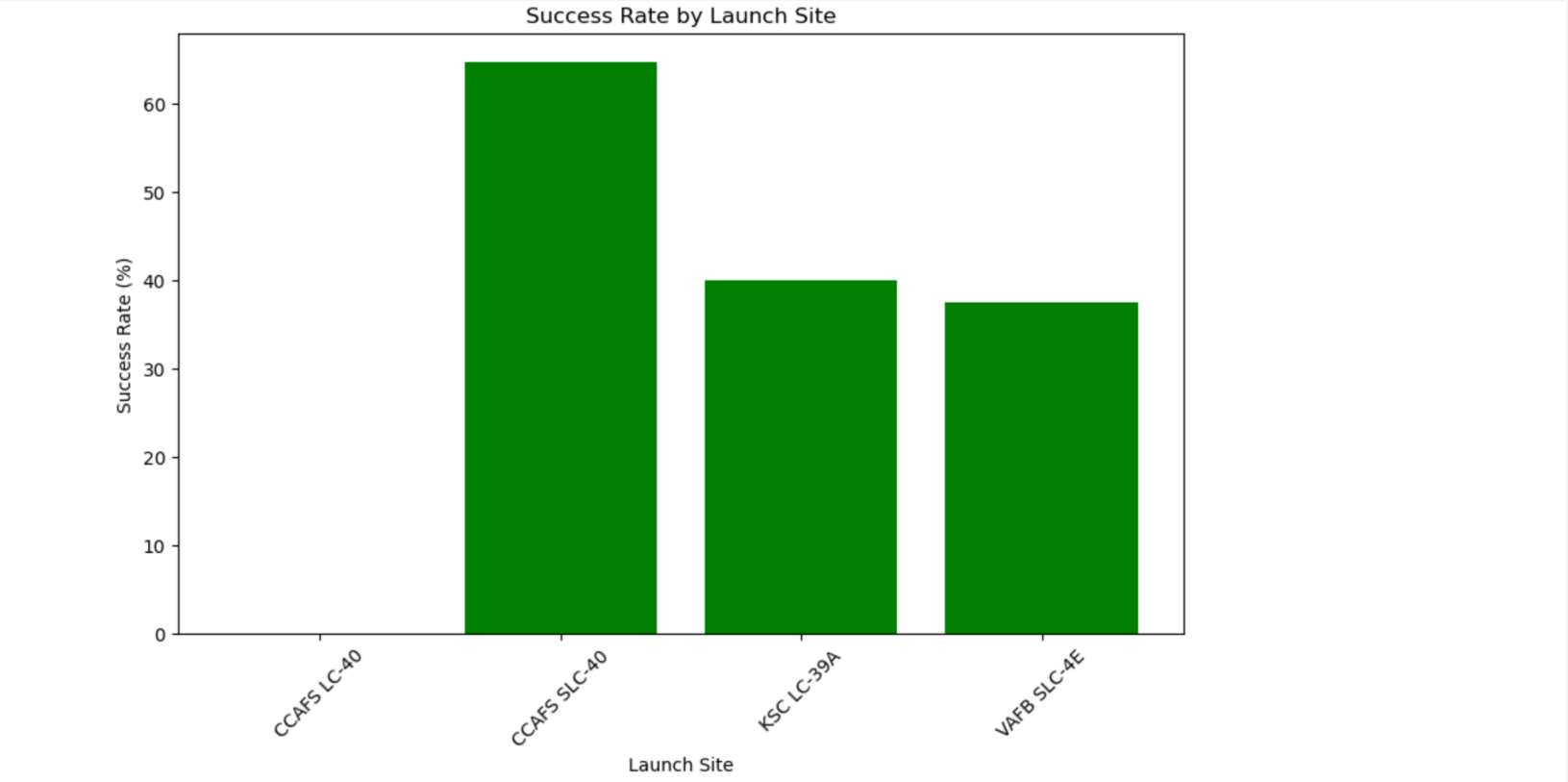
Conclusion

- In conclusion, the project involved collecting and processing data from multiple sources, including APIs and web scraping. After performing exploratory data analysis, feature engineering, and geospatial analysis, a machine learning pipeline was developed to predict the landing success of Falcon 9's first stage.
- Key findings include:
- The **landing success rate** is influenced by factors such as payload mass, orbit type, and launch site.
- **Geospatial patterns** suggest that certain launch sites have higher success rates, possibly due to geographical or logistical advantages.
- **Random Forest** was the best-performing model for predicting landing success.

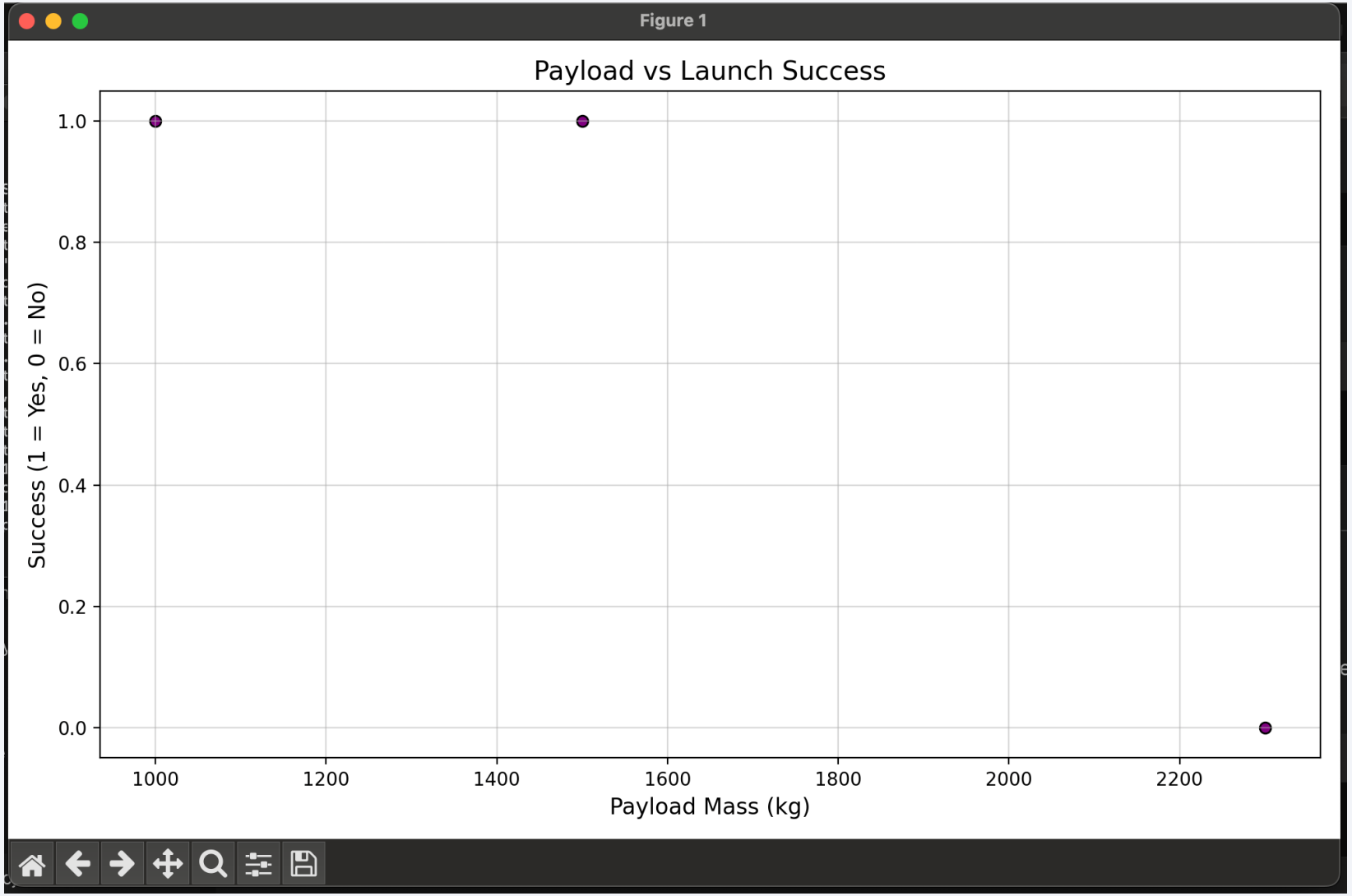
Launches by Site



Build a Dashboard with Plotly Dash



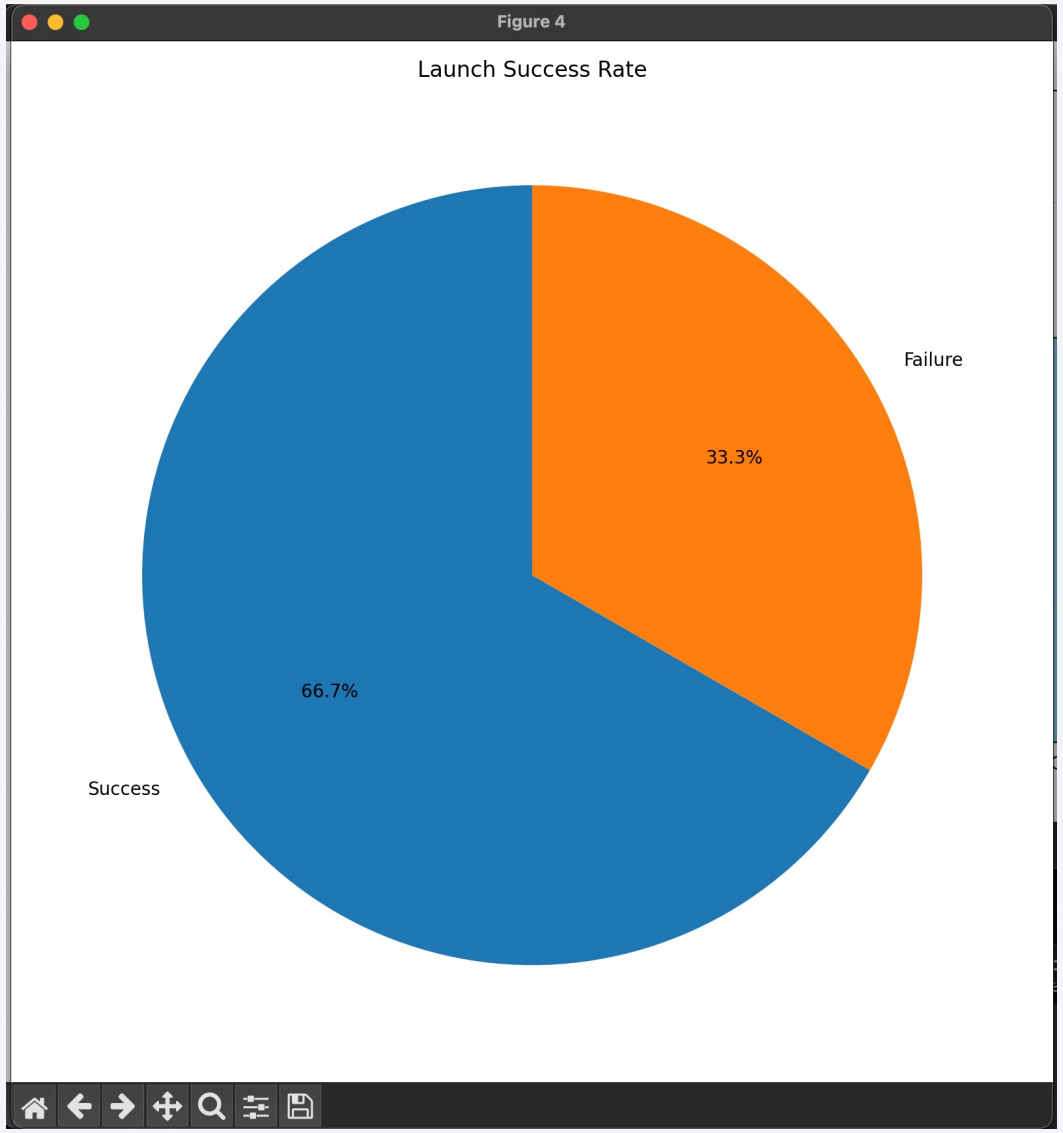
Payload vs Launch Success



Payload vs Launch Success Code

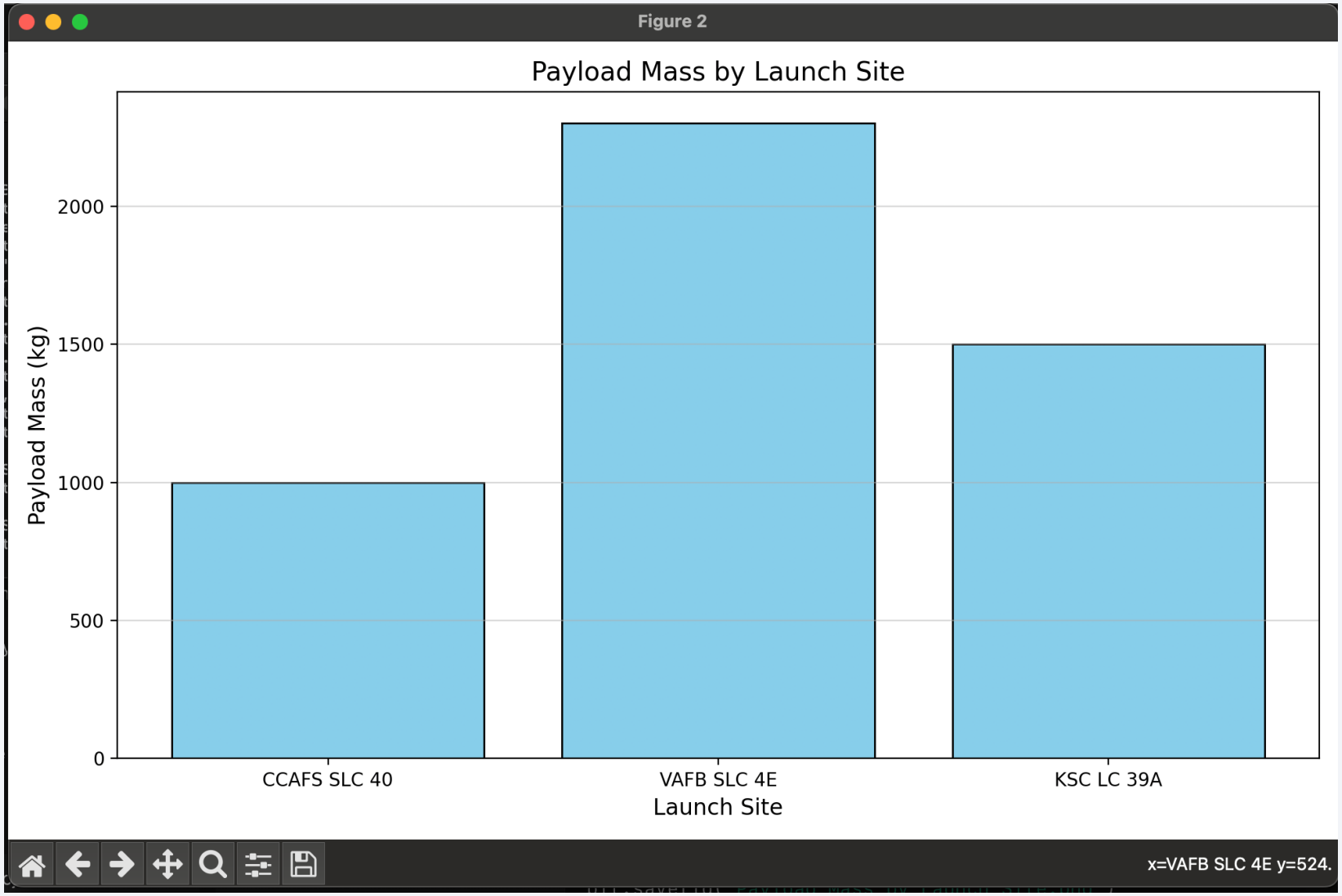
```
# 3. Scatter Plot: Payload vs Success
plt.figure(figsize=(10, 6))
plt.scatter(data['PayloadMass'], data['Success'], color='purple', edgecolor='bla
plt.title('Payload vs Launch Success', fontsize=14)
plt.xlabel('Payload Mass (kg)', fontsize=12)
plt.ylabel('Success (1 = Yes, 0 = No)', fontsize=12)
plt.grid(alpha=0.5)
plt.tight_layout()

# Save the plot
plt.savefig('payload_vs_launch_success.png', dpi=300, bbox_inches='tight')

# Display the plot
plt.show()
```

Launch Success Rate and Code




```
success_counts = [2, 1]  # Replace with real values if necessary
labels = ['Success', 'Failure']
plt.figure(figsize=(8, 8))
plt.pie(success_counts, labels=labels, autopct='%1.1f%%', startangle=90)
plt.title('Launch Success Rate')
plt.tight_layout()
plt.show()
```

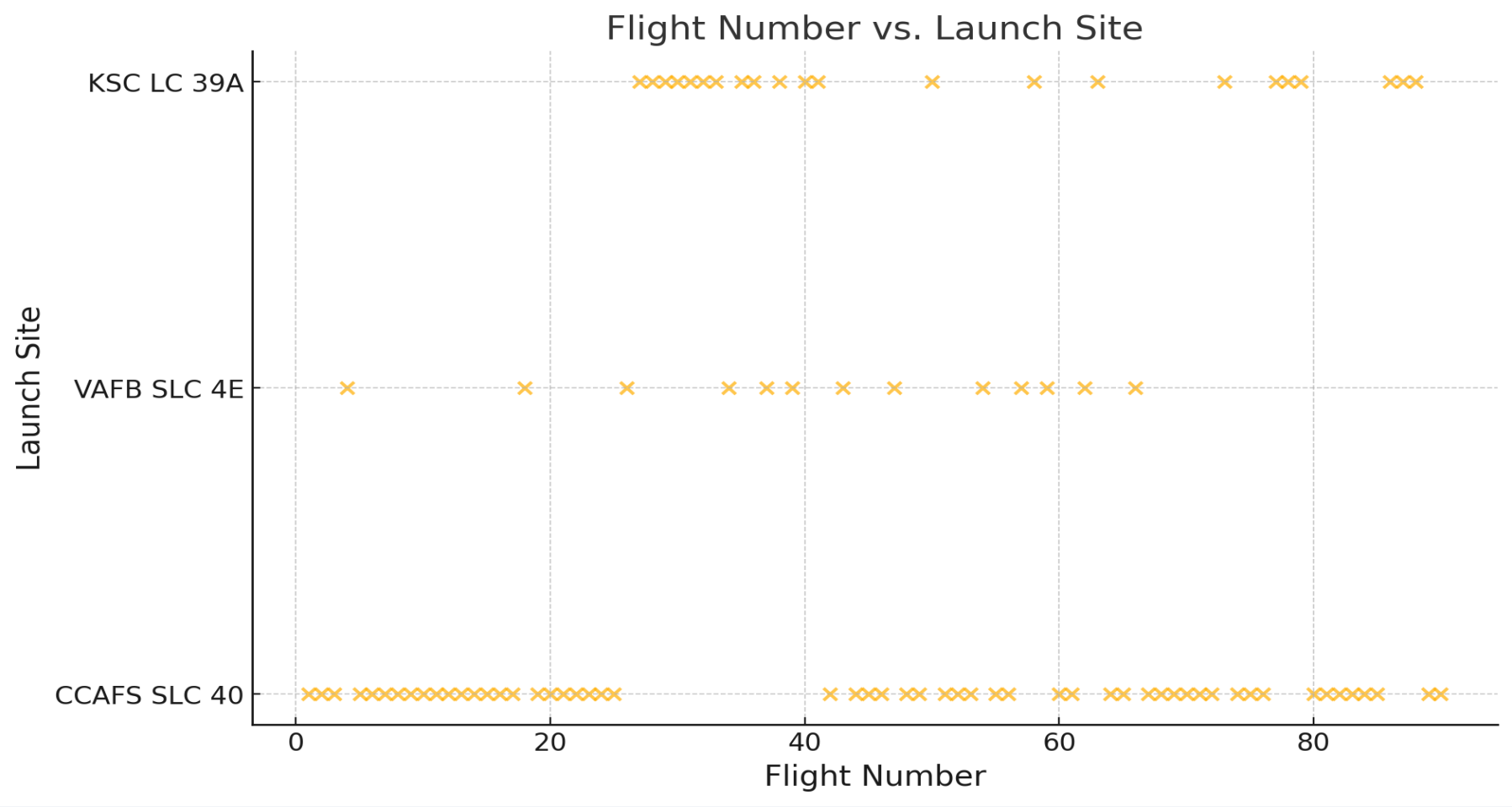
Payload Mass by Launch Site



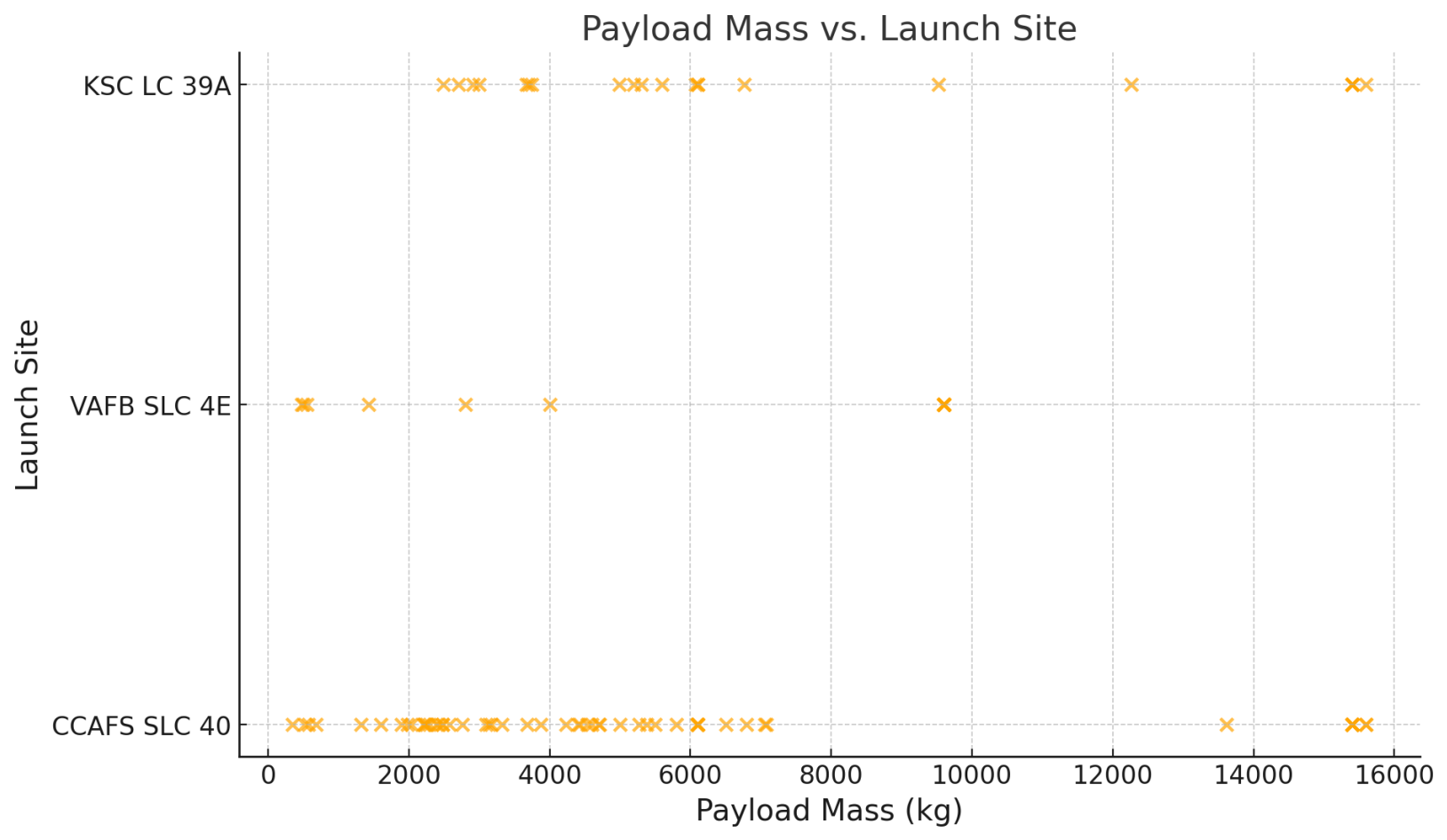
Payload Mass by Launch Site code

```
import matplotlib.pyplot as plt
import pandas as pd

# Example dataset (you can replace this with your actual CSV file or data source
data = pd.DataFrame({
    'LaunchSite': ['CCAFS SLC 40', 'VAFB SLC 4E', 'KSC LC 39A'],
    'PayloadMass': [1000, 2300, 1500]
})

# Bar Plot
plt.figure(figsize=(10, 6))
plt.bar(data['LaunchSite'], data['PayloadMass'], color='skyblue', edgecolor='bla
plt.title('Payload Mass by Launch Site', fontsize=14)
plt.xlabel('Launch Site', fontsize=12)
plt.ylabel('Payload Mass (kg)', fontsize=12)
plt.grid(axis='y', alpha=0.5)
plt.tight_layout()

# Save the plot
plt.savefig("Payload_Mass_by_Launch_Site.png")

# Show the plot
plt.show()
```

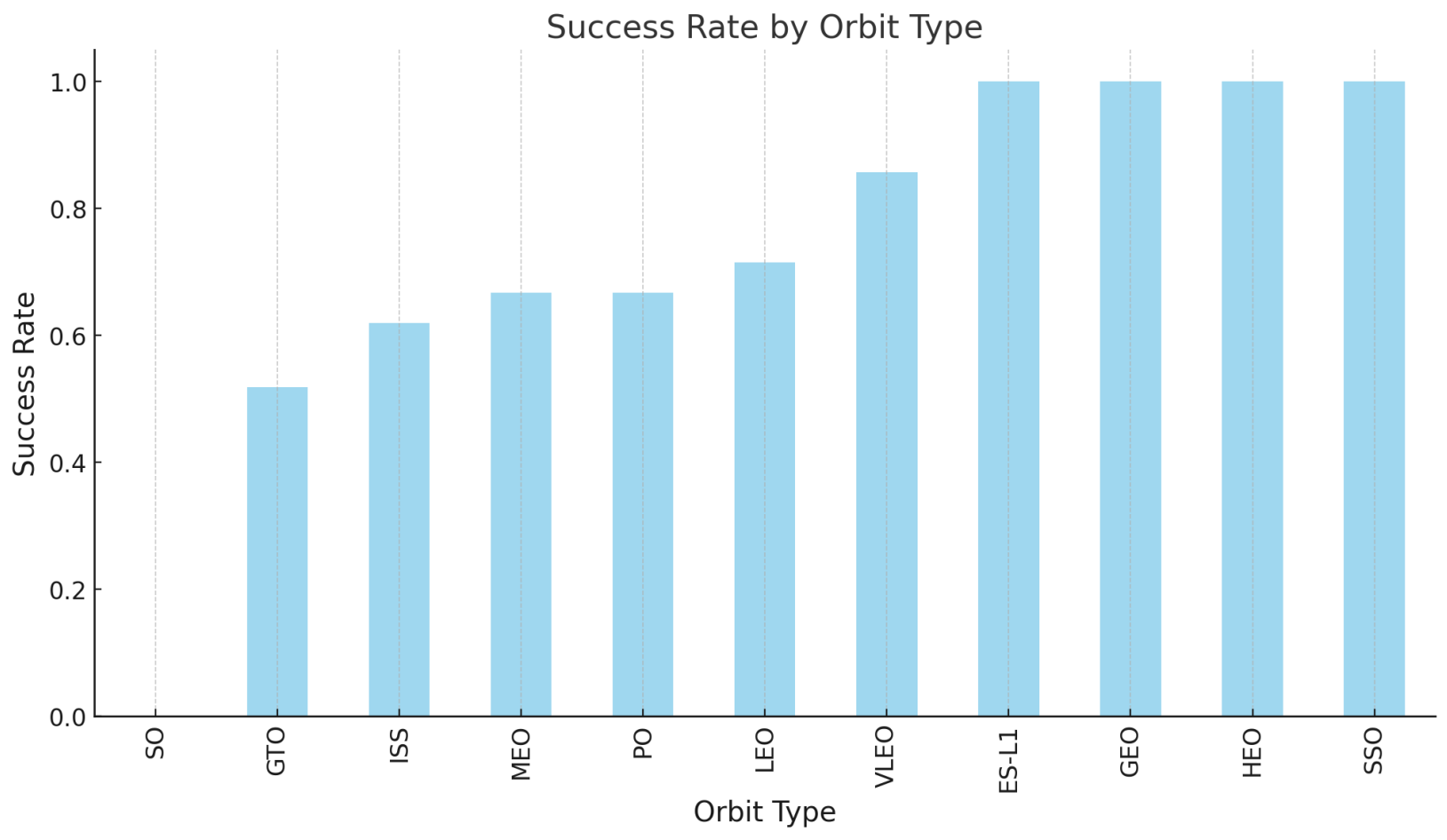
Flight Number vs. Launch Site



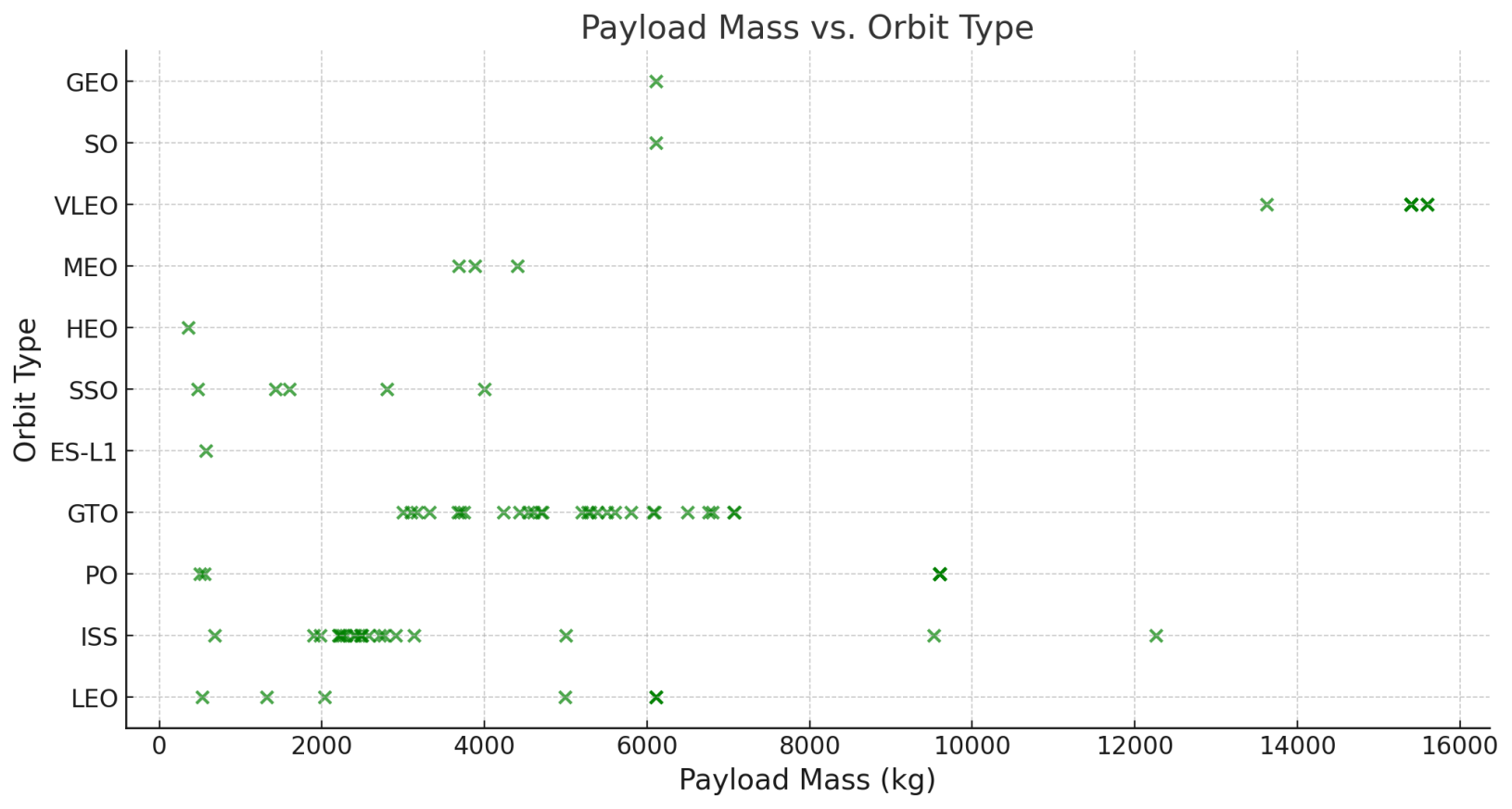
Payload Mass vs. Launch Site



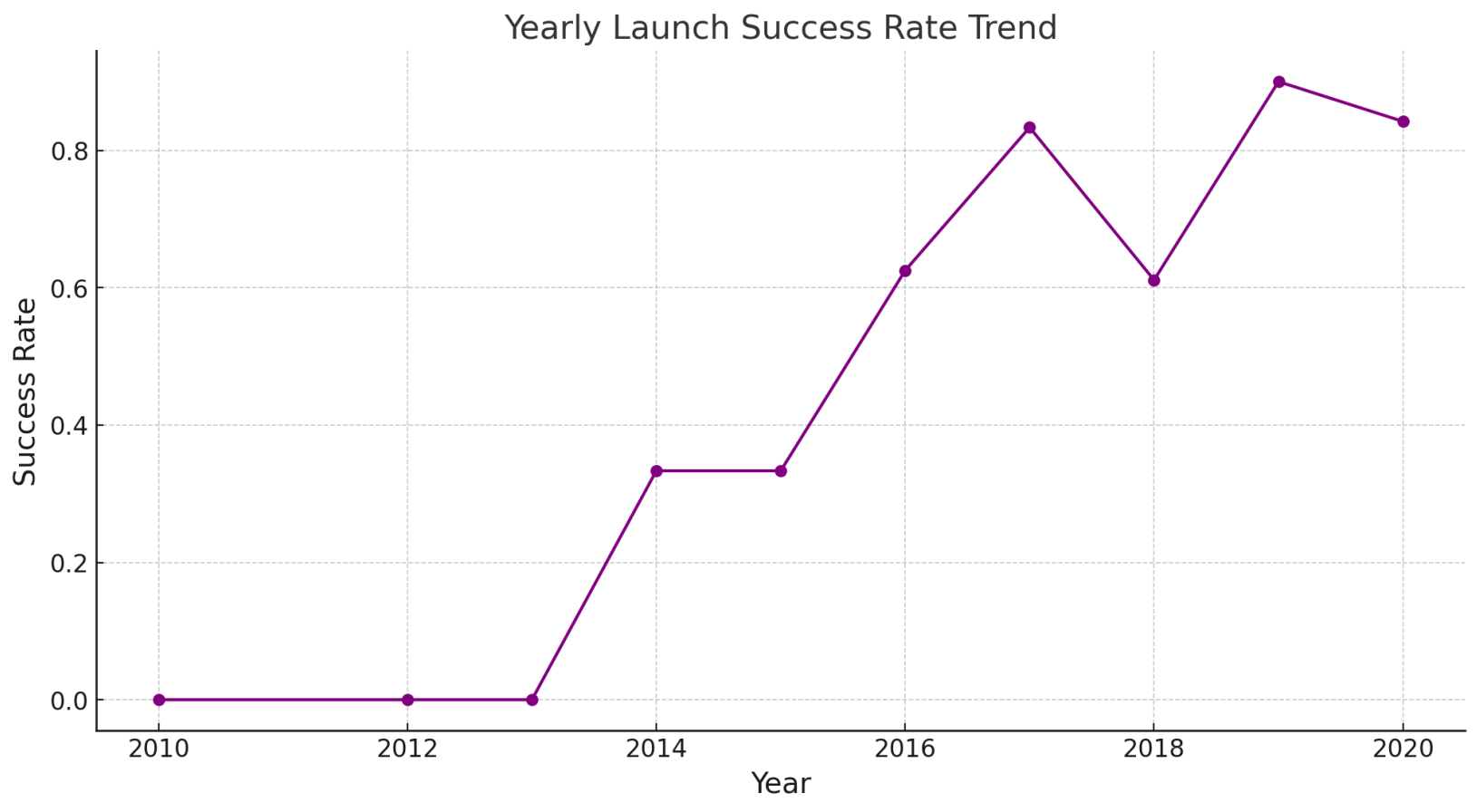
Success Rate by Orbit Type



Payload Mass vs. Orbit Type



Yearly Launch Success Rate Trend



Thank you!